FTF 2016
TECHNOLOGY FORUM

AN INTRODUCTION TO ZIGBEE 3.0 USING THE JN516X WIRELESS MICROCONTROLLER

IAN MORRIS
APPLICATIONS ENGINEER
SESSION FTF-HMB-N1983
19, MAY, 2016

Introductions

Ian Morris

Applications Engineer

NXP Semiconductors (USA)

Shashank Goel

Product Marketing Manager

NXP Semiconductors (USA)

Overview

- **What is the purpose of this session?**
  - We are going to learn about ZigBee 3.0 and will address the following:
    - What is ZigBee 3.0?
    - How do I develop a product that uses ZigBee 3.0 technology?
    - What do I need to know to get this product to market?

- **What is the format of this session?**
  - A mix of theory and exercises using the JN516x Wireless Microcontroller and associated development tools.

- **What is the style of this session?**
  - Relaxed, feel free to ask questions at any time!

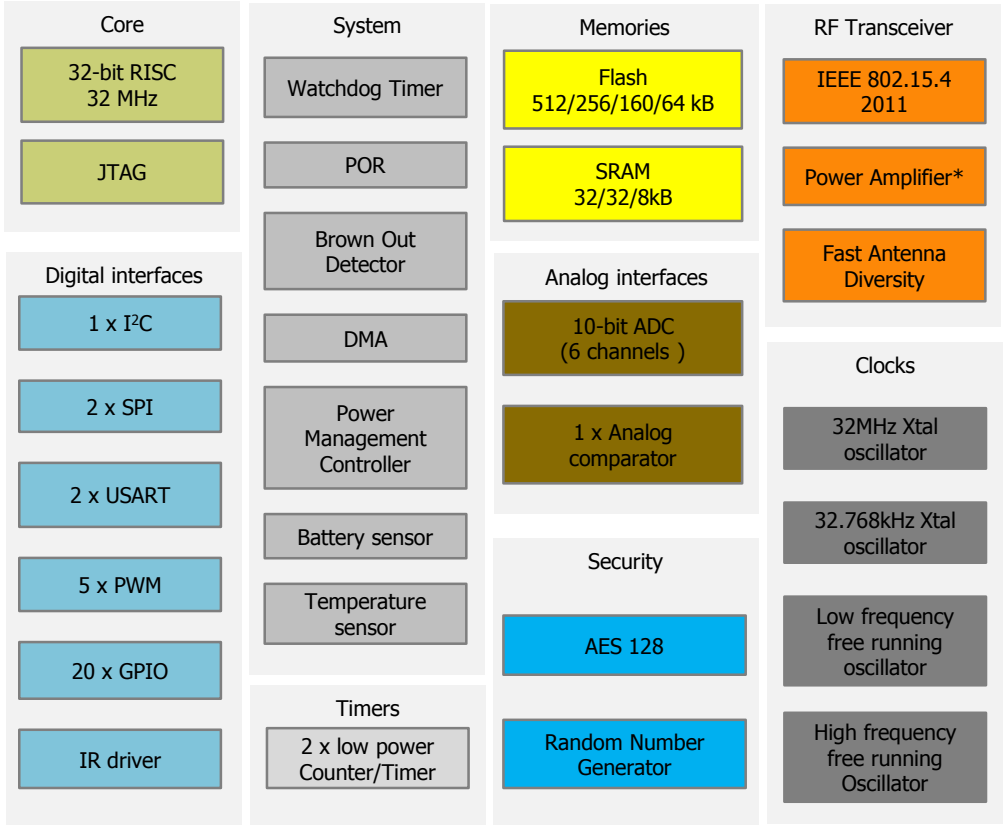
AGENDA

- Theory 1: Silicon - JN516x Wireless Microcontrollers
- Theory 2: Software - ZigBee
- Theory 3: Software - ZigBee 3.0
- Hands On Modules: Preparation
- Hands On Module 1: Joining
- Hands On Module 2: Device & Service Discovery
- Hands On Module 3: Addressing Modes
- Hands On Module 4: Using Reporting
- Q&A

THEORY 1: SILICON
JN516X WIRELESS MICROCONTROLLERS

JN516x Wireless Microcontroller

- **CPU**
  - 32 MHz, 32-bit RISC CPU core
  - 512/256/160/64 kB Flash & 32/8kB RAM & 4KB EEPROM
- **2.4 GHz radio transceiver**
  - IEEE-802.15.4 compliant
  - Antenna diversity
  - Up to +10dBm TX power
  - Up to -96dBm RX sensitivity
  - Peak typical current:
    - 14mA @ +3dBm, 23.3mA @ +10dBm
    - 15mA RX
- **Security**
  - Crypto engine: AES 128, RNG
- **System**
  - USART, SPI, I²C, PWM, IR
  - 10-bit ADC, Analog Comparator
  - Battery operating range: 2.0V to 3.6V,
  - Ambient temperature : -40°C to +125°C
  - HVQFN40 6x6mm



* Only available on the JN5169

JN516x Modules

- **All modules include JN5168 or JN5169 plus supporting components**
  - Surface mountable on motherboards
- **Module Configurations**
  - Standard power modules
    - With integrated printed antenna — 16x30mm — **JN5168-001-M00**
    - With uFl connector — 16x21mm — **JN5168-001-M03**
  - Medium power modules (+10dBm)
    - uFl connector — 16x30mm — **JN5168-001-M05**
    - With integrated printed antenna — 16x30mm — **JN5169-001-M00**
    - uFl connector — 16x21mm — **JN5169-001-M03**
  - High power modules (+22dBm)
    - uFl connector — 16x30mm — **JN5168-001-M06**
    - uFl connector — 16x30mm — **JN5169-001-M06**
- **Module value proposition**
  - Fast time to market
  - Approved to FCC and EU regulations
  - No need for RF design resource for board design and testing
  - Overall lowest cost of implementation up to 20-50ku

JN5168-001-M00

JN5168-001-M03

JN5168-001-M05

JN5169-001-M00

JN5169-001-M03

JN5168-00-M06

JN5169-001-M06

JN516x Development Tools

JN516x-EK004 Evaluation Kit

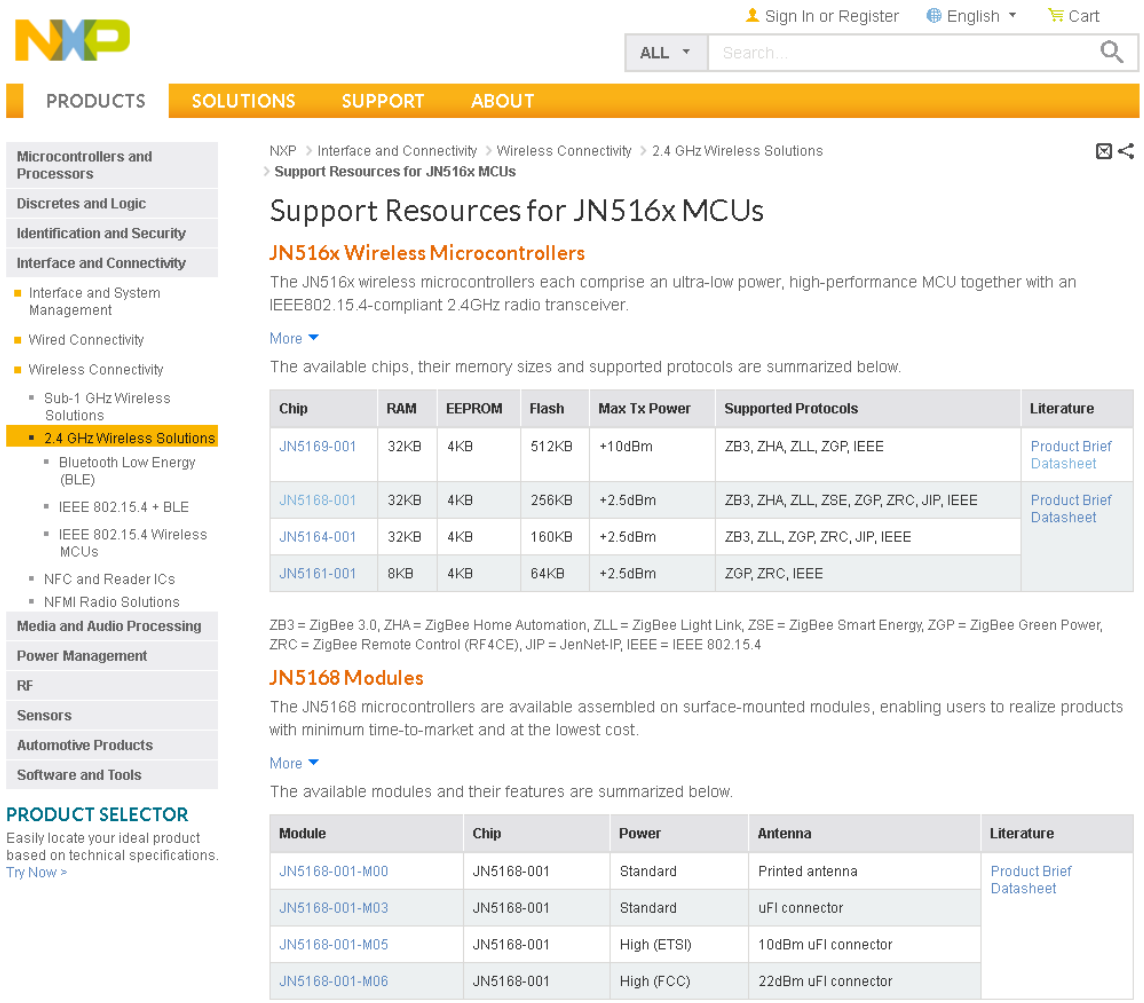
Contains all hardware components to demonstrate, evaluate and develop ZigBee solutions including NFC commissioning.

Eclipse/GCC based SDK

Eclipse based IDE and GCC compiler. Integrated flash programmer and GUI for ZigBee stack/application configuration.

JN516x Support Collateral

- **Comprehensive set of supporting material is available on the NXP Website, including:**
  - User Guides
  - Reference Manuals
  - Application Notes
  - Reference Designs
  - Test Tools
  - Datasheets
  - Development Tools
  - Software Libraries



THEORY 2: SOFTWARE ZIGBEE

What is ZigBee?

“ZigBee is the only open, global wireless standard to provide the foundation for the Internet of Things by enabling simple and smart objects to work together, improving comfort and efficiency in everyday life.”

IoT Gateway
Connects network to the Internet

User Console
Provides interface for user input and output

Meters
Record user consumption data

Smart Devices
Provide autonomous control

What is ZigBee?

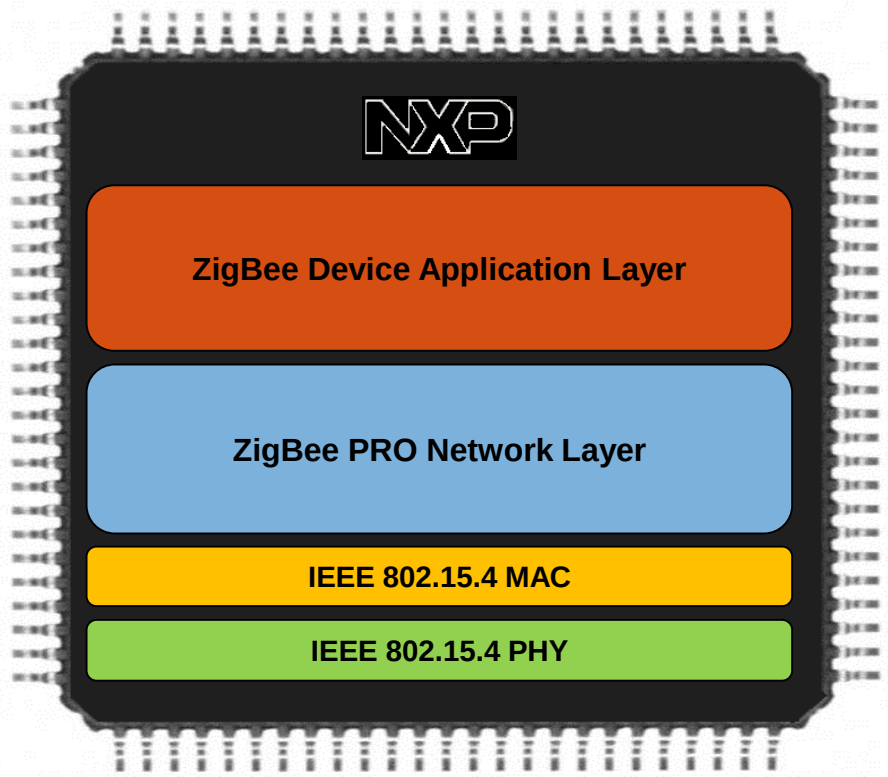
ZigBee provides a full stack on top of the IEEE 802.15.4 MAC.

ZigBee Features:

- Industry-standard protocol defined by ZigBee Alliance
- Facilitates interoperability between products from different vendors
- Sits on and enhances IEEE 802.15.4
- Provides Mesh networking capability

NXP-Specific Features:

- Standard ‘ C’ APIs to simplify ZigBee application coding
- Configuration tools
- Certified stack as part of the released SDK
- Verified on a 250+ node Large Network before public release



ZigBee Features

- Operates in **IEEE 802.15.4 radio bands**: 868 MHz, 915 MHz, 2400 MHz
- Contains the following node types:
  - **Coordinator**: Forms network, accepts children and routes data packets
  - **Router**: Accepts children and routes data packets
  - **End Device**: No child or routing functionality (only sends/receives data)
- Supports **Mesh topology** for flexible routing and route repair
- Standard device definitions for **key market sectors –** for example:
  - Smart Energy
  - Home & Building Automation
  - Retail Services
- **Discovery and binding** mechanisms for easy communication
- **Highly secure** with network-level and application-level security options
- **Certification programme** to support co-existence and interoperability of products

ZigBee Alliance Membership and Fees

Three types of ZigBee Alliance member:

- **Adopter** (*annual fee: $4000*): Access to final specs, use of the ZigBee Member logo, participation in interoperability events, access to standard work/task group documents and development activities
- **Participant** (*annual fee: $9900*): Early access to specs in development, full participation in all committees, work/task groups and member meetings (can earn voting rights)
- **Promoter** (*annual fee: $55000*): Automatic voting rights in all work groups, final approval rights on all standards, a seat on the Board of Directors

NXP is a Promoter and is on the Board of Directors

ZigBee Compliance and Certification

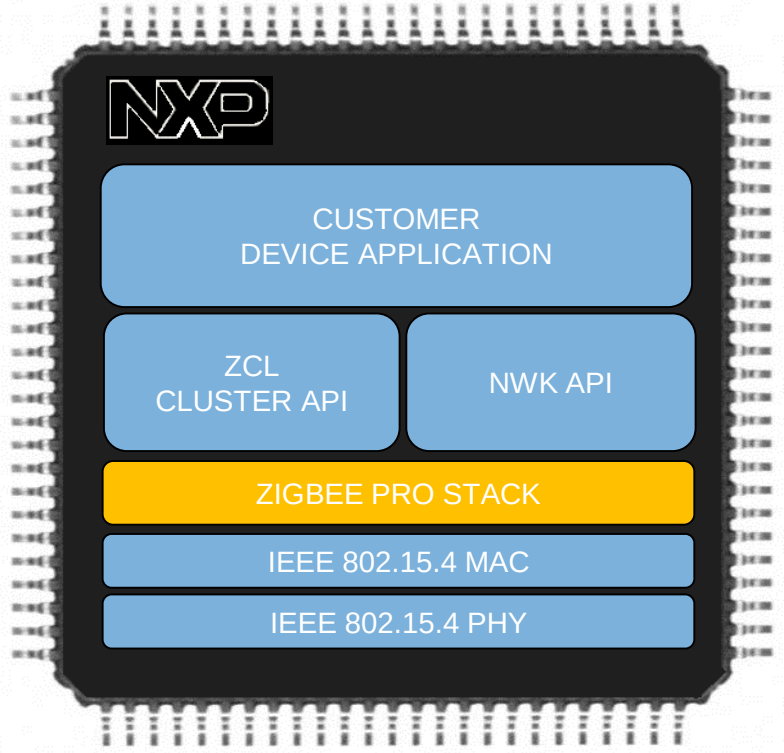
ZigBee Alliance defines two levels of compliance:

- **ZigBee Compliant Platform (ZCP)**
  - Applies to devices intended as building blocks for use in end-products
- **ZigBee Certified Product**
  - Applies to end-products built on ZCPs and that use public ZigBee Alliance application profiles
  - After successful completion of ZigBee Alliance certification programme, the ZigBee Certified Product logo can be applied to the product
  - End-products based on manufacturer-specific profiles can also obtain ZigBee Certified Product status but cannot carry the logo

ZigBee Alliance authorise test service providers to undertake certification.

Cost of certification: *$3000-4500 per device*

Certification Responsibilities



NXP	CUSTOMER
NXP certify example application to reduce the customer risk	Customer makes manufacturer-specific updates and is required to re-certify the device

NXP	CUSTOMER
NXP certifies a ZigBee PRO compliant platform	Customer provides the NXP certification document to the relevant test house

THEORY 3: SOFTWARE
ZIGBEE 3.0

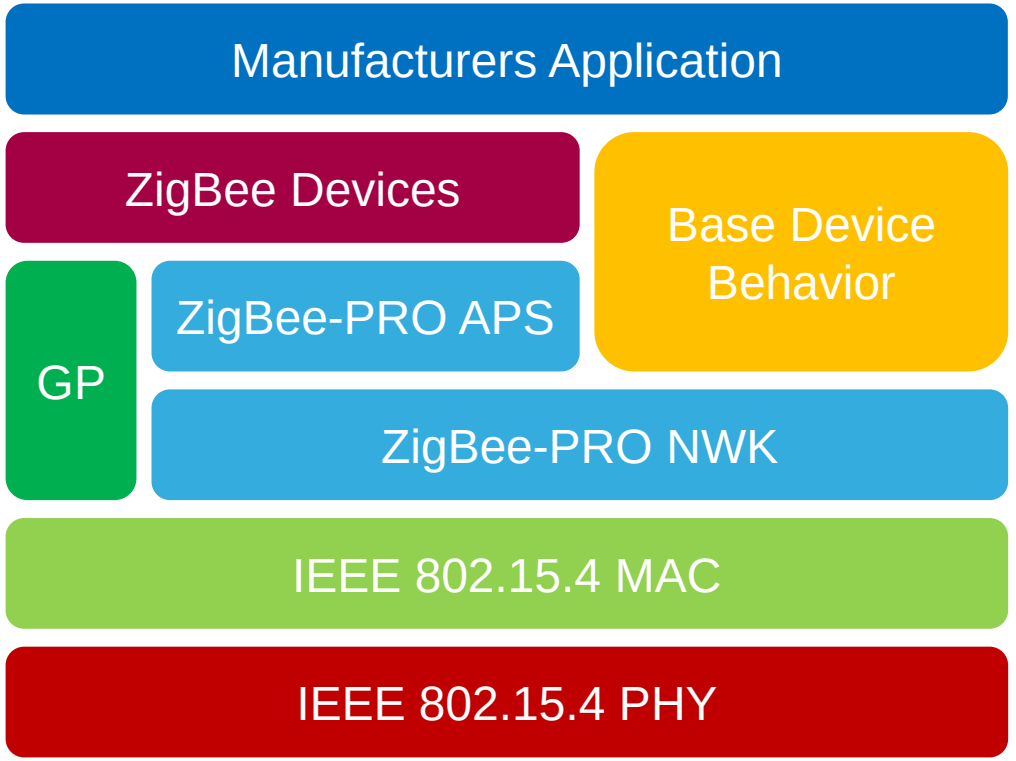
Why is ZigBee 3.0 needed?

Overcomes the following issues with former profile-based ZigBee:

- Lack of interoperability between device manufacturers/profiles due to commissioning/discovery/normal operation
- No defined mechanism for sleepy child maintenance
- Security hole around insecure rejoin and use of well-known keys
- Lack of a universal set of rules to promote cohesive solution development

What does ZigBee 3.0 provide?

- Combines multiple application profiles into a collection of devices
  (e.g. **Lighting & Occupancy** devices)
- Clusters from these profiles incorporated into the **ZCL R6**
- An obligatory '**Base Device**', providing basic networking functionality required by all nodes (BDB stands for Base Device Behavior)
- Alignment on network commissioning and common security models
- Revised **ZigBee PRO stack (R21)**
- **Backwards compatibility** with all previous ZigBee PRO based stacks, so will not alienate devices already deployed in the field

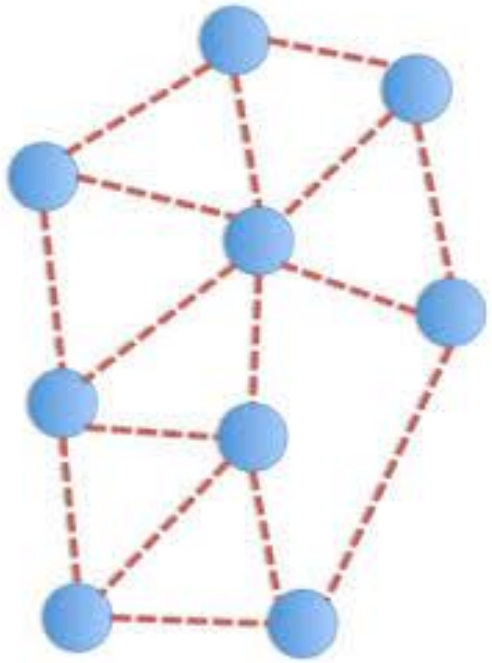


Combined Application Profiles

- *ZigBee 3.0 unifies market-specific application profiles to allow all devices to be wirelessly connected and communicate, irrespective of their market designation and function.*
- ZLL and HA profiles incorporated in Lighting & Occupancy (ZLO) Group of Devices
- Other profiles (e.g. Health Care, Building Automation) will adopt ZigBee 3.0 when the relevant workgroups can find a Technical Author resource
- Smart Energy profile resisting move to ZigBee 3.0 due to security risks
  - ZigBee Smart Energy uses advanced security based on elliptical curve cryptography, specifically implemented for use by electric utilities to enable a highly secured smart grid. This level of security will be integrated as an optional feature of ZigBee 3.0 in the future, allowing Smart Energy to be incorporated into ZigBee 3.0

What does ZigBee PRO R21 provide?

- **Trust Centre Link Key (TCLK)**
  - Request of a new link key between the joining device and Trust Centre
  - This feature uses a new key type and 2 additional APSME messages
- **Aging mechanism for End Devices**
  - New commands added for declaring ‘keep-alive’ time and for parent to convey which keep-alive mechanism it supports
- **Two mechanisms for keep-alive on the parent**
  - Data request (new command sent to keep alive)
  - Poll request
- **Two network types based on security**
  - Centralised network with security managed by single Trust Centre
  - Distributed network with security managed across all Routers



ZigBee 3.0 Status

- ZigBee Alliance will shortly open the certification program for ZigBee 3.0
- Golden Units have been deprecated
- NXP JN51xx is only one of two devices that can be used for the ZigBee 3.0 ZTT
- This tool will be available for members to hire
- NXP ZigBee 3.0 SDK Beta program has opened
  - JN516x ZigBee 3.0 SDK (JN-SW-4170)

ZigBee 3.0 ZTT

(ZigBee Test Tool)

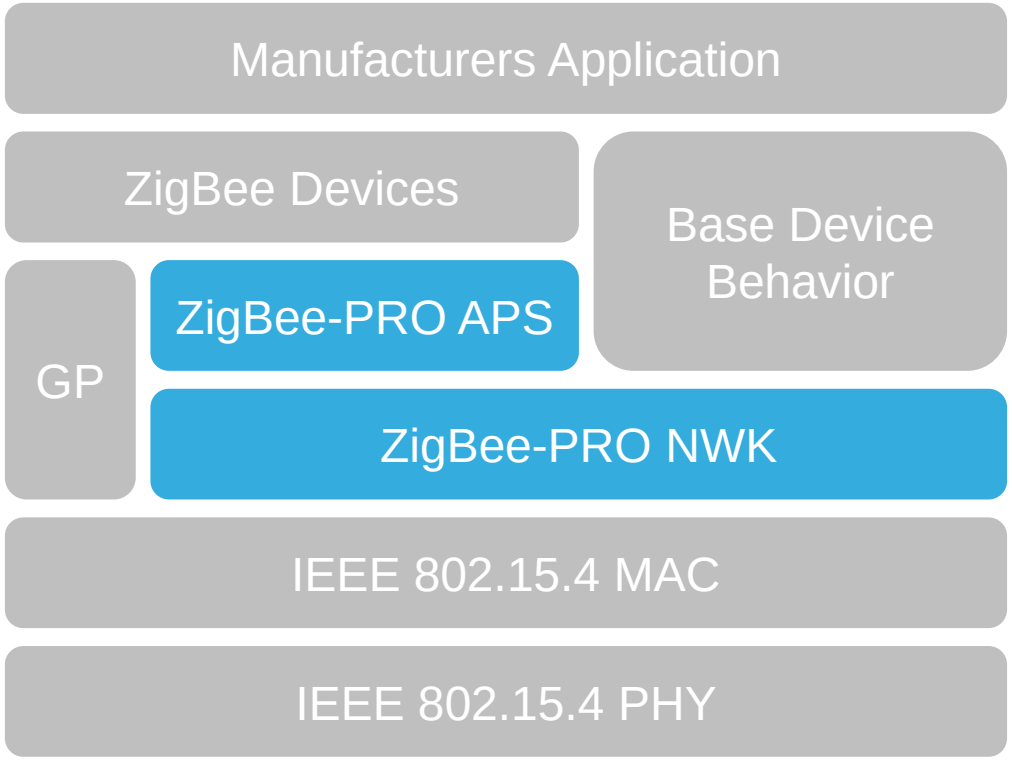
PC Test Tool

Supplier Element

ZigBee 3.0 dongle

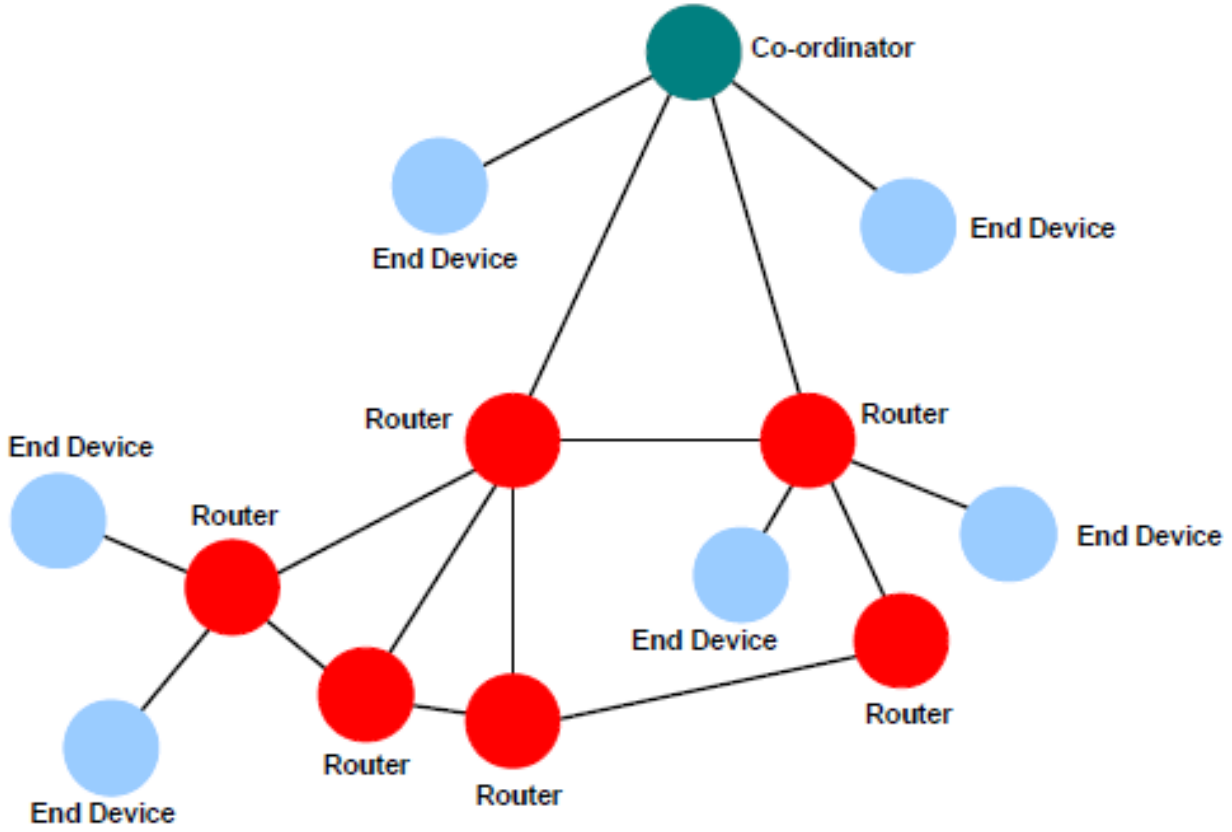
Suppliers NXP/Silabs

ZigBee-PRO NWK & APS Layers



Network Topology

- **Coordinator**
  - This is the first node to be started and is responsible for forming the network.
  - Has a routing role (able to relay messages from one node to another) and also able to send/receive data.
- **Router**
  - Has routing capability, is also able to send/receive data.
  - Also allows other nodes to join the network through it, so plays a role in extending the network.
  - A network may have many Routers
- **End Device**
  - A node which is only capable of sending and receiving data (no routing capability).
  - A network may have many End Devices



Network Identity

Two network identifiers are used:

PAN ID

- 16-bit value (as in IEEE 802.15.4 standard)
- Should be unique in operating neighbourhood
- Randomly assigned by Coordinator at start-up
- Used in messaging between network nodes

Extended PAN ID (EPID)

- 64-bit value
- Can be pre-programmed or obtained from MAC address of Coordinator at start-up
- If pre-programmed, used in network formation to identify network to join

Network Addressing

- **IEEE (MAC) address:** This is a 64-bit address, allocated by the IEEE, which uniquely identifies the device - no two devices in the world can have the same IEEE address. It is often referred to as the MAC address and, in a ZigBee network, is sometimes called the ‘extended’ address.

- **Network address:** This 16-bit address identifies the node in the network and is local to that network (thus, two nodes in separate networks may have the same network address). It is sometimes called the ‘short’ address.

Security Keys

- ZigBee 3.0 uses AES 128 bit security (encryption & MIC) with the following keys:
  - **Pre-Shared Link Key**: Pre-programmed into ALL devices that want to join
  - **Network key**: Shared between ALL devices that have joined the network
  - **Trust Center Link Key**: New to ZigBee 3.0

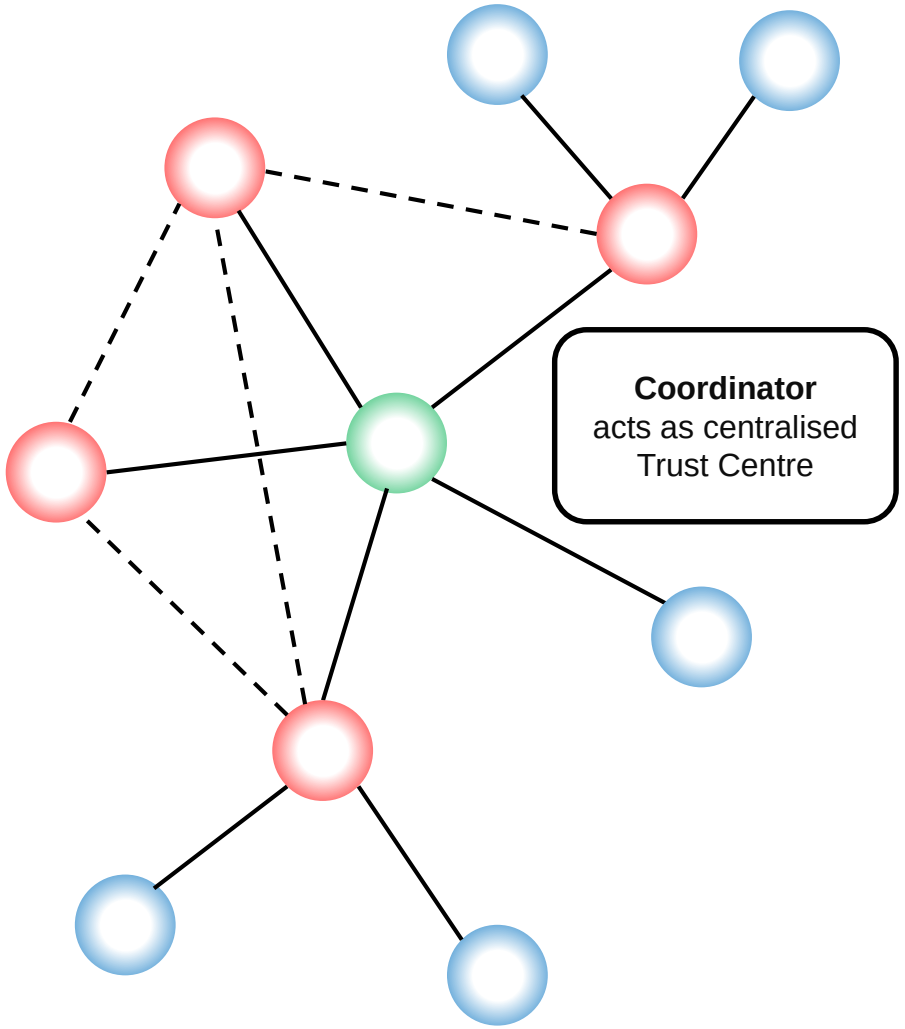
Trust Centre Link Key (TCLK)

- TCLK is a unique link key to encrypt packets between the TC and one other node
- Trust Centre generates a TCLK for each node that joins the network and holds a list of TCLKs for all the nodes in the network
- Trust Centre passes the TCLK to the joining node, encrypted with the Network Key and Pre-configured Link Key (shared by both nodes)
- TCLK replaces Pre-configured Link Key for securing subsequent communications
- TCLK also used by node to do an insecure re-join (without network-level security), particularly relevant after being removed from the network by child ageing:
  - Node will first try several secure re-joins using the network key that it still has
  - While out of the network, the network key may have been ‘rolled’ (switched)
  - If all the secure re-joins fail, the node will try an insecure re-join with the TCLK
  - If insecure re-join is successful, the Trust Centre will pass the latest network key to the node

(Pre-ZigBee 3.0, the node could do an insecure re-join with the default key, even with ‘Permit Join’ off)

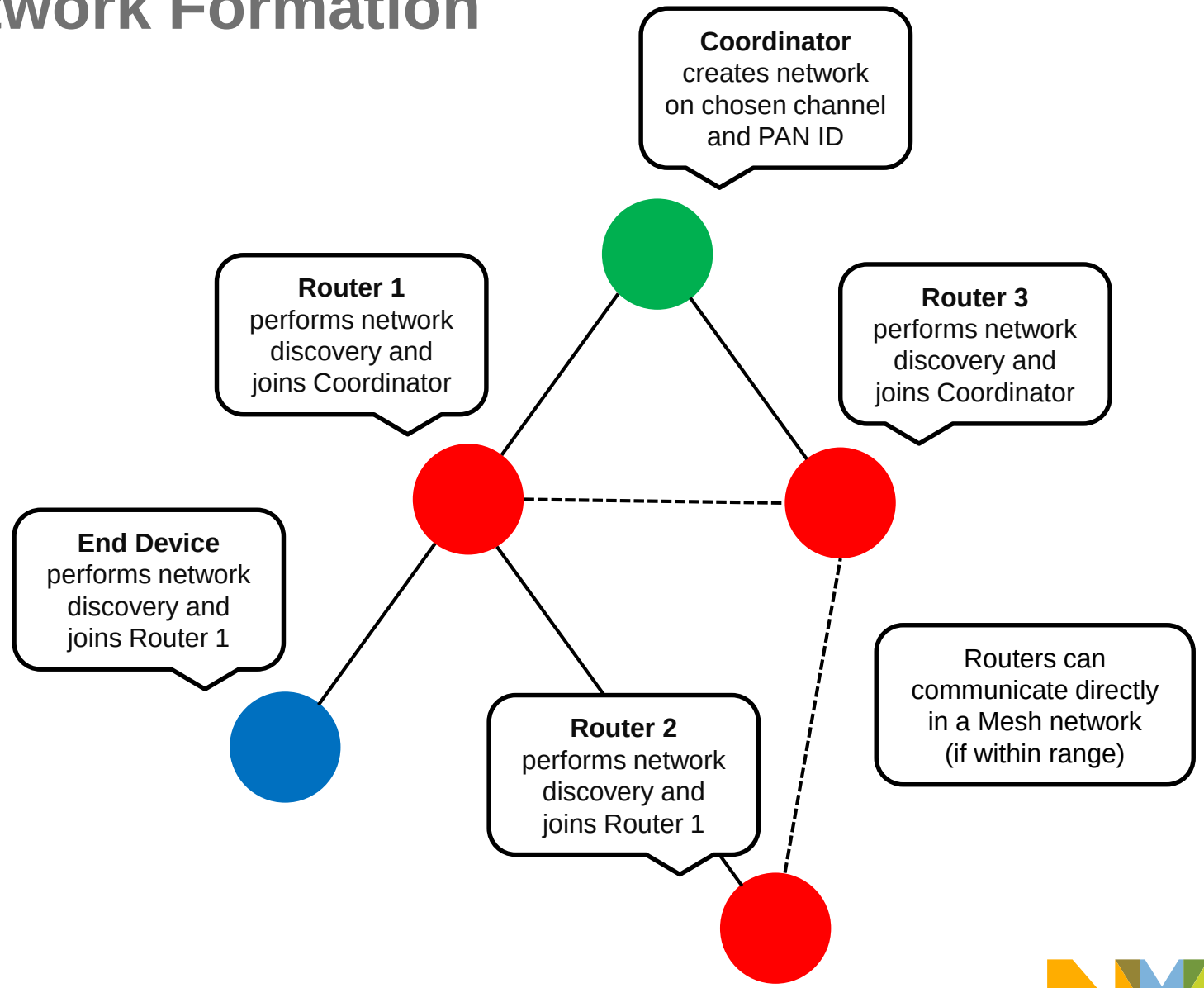
Centralized Security Network

- Always has a Coordinator
  - Needed for joining new nodes to network
  - Not needed for node re-joins
- Only one Trust Centre (usually the Coordinator)
- Trust Centre Link Key (TCLK) is supported
- Trust Centre supports unique keys, permissions and other TC policies
- Trust Centre address is unique MAC address of the Coordinator

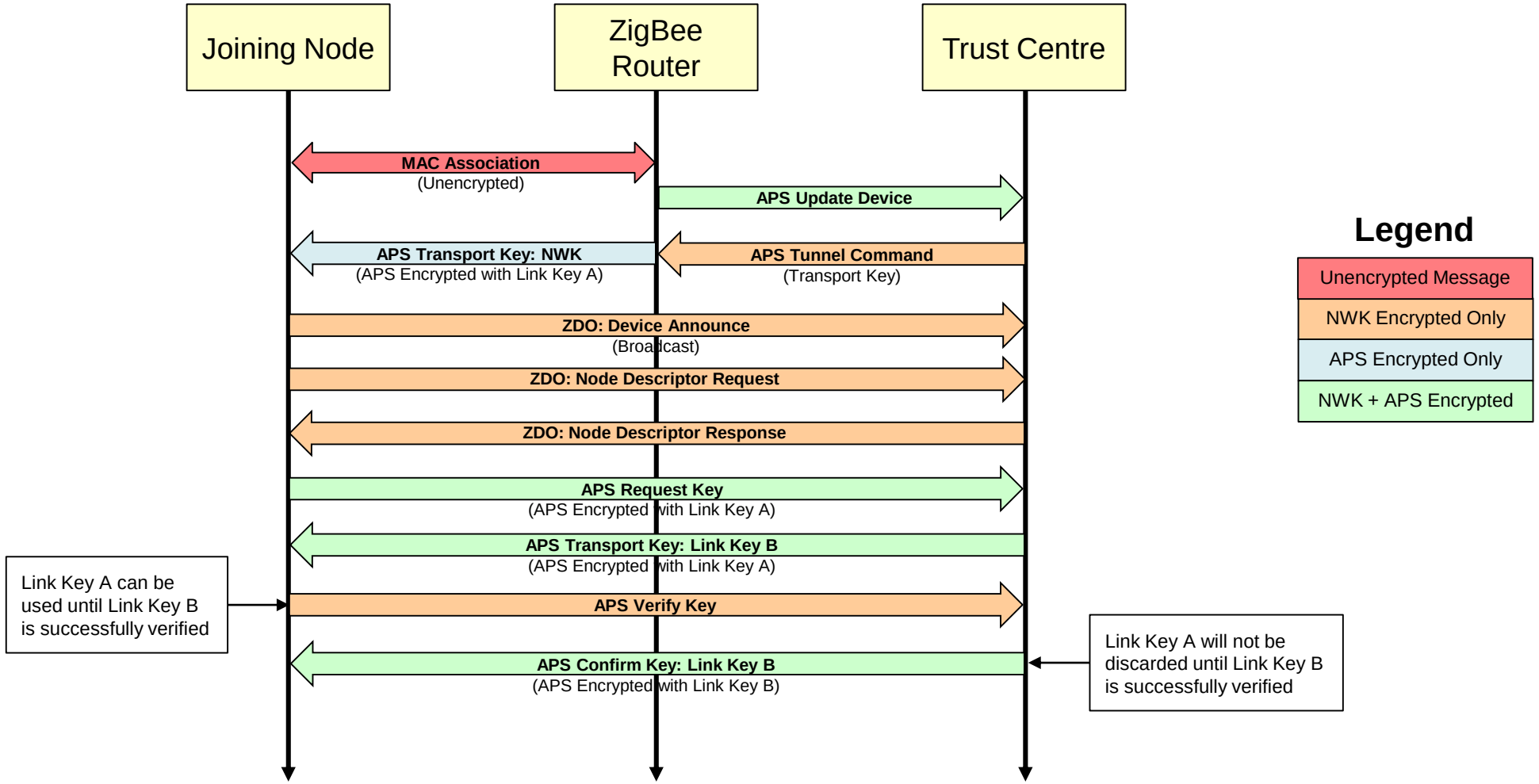


Centralized Security Network Formation

1) Coordinator started first
   - Creates network
   - Chooses radio channel
   - Assigns PAN ID and Extended PAN ID (EPID)
   - Initialises as Trust Centre
2) Next node joins the Coordinator
3) Further nodes join either the Coordinator or a Router (if present)

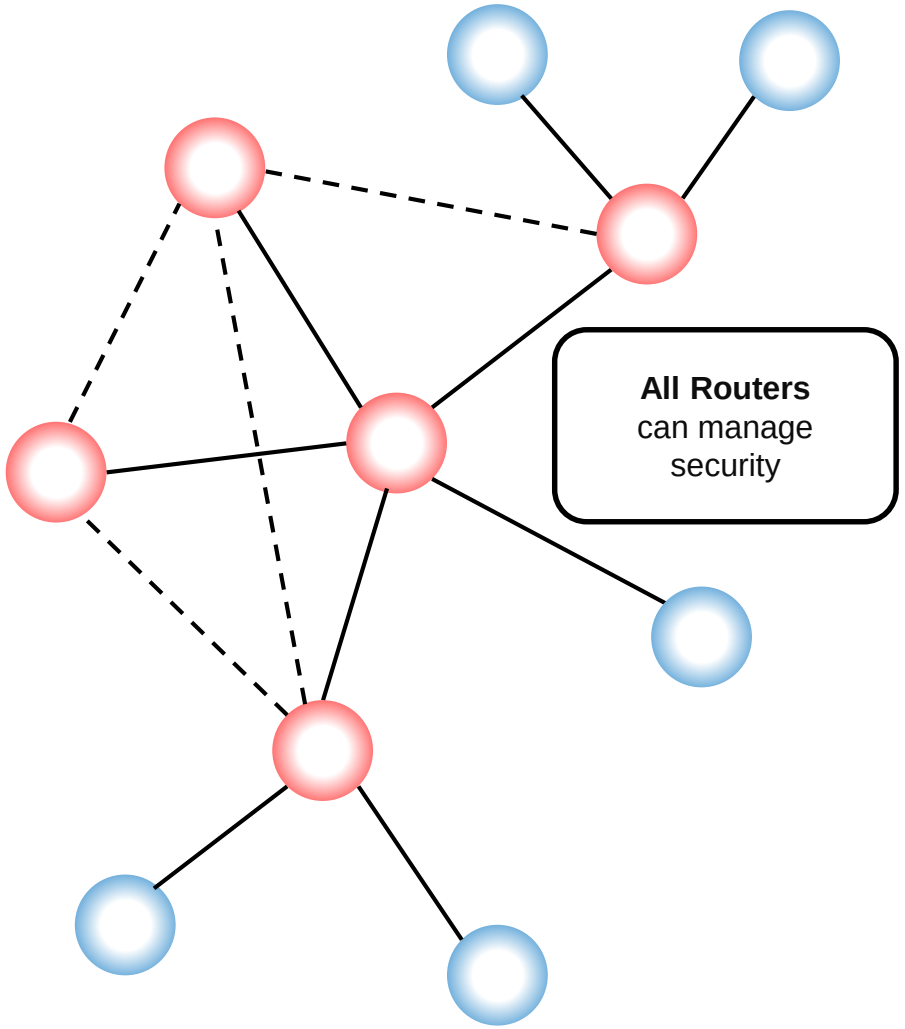


TCLK Negotiation

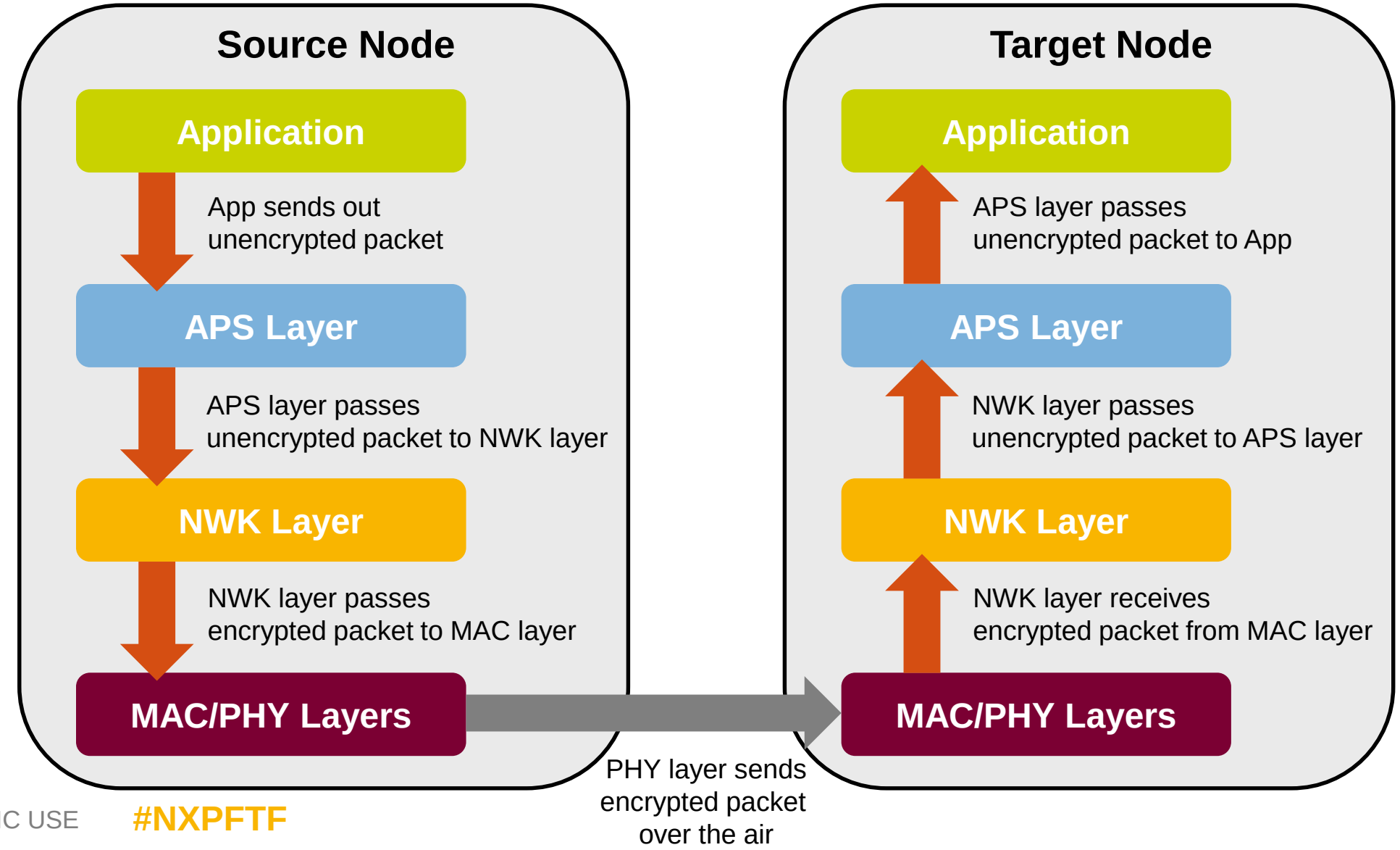


Distributed Security Network

- No Coordinator in the network
- No single Trust Centre
- Each Router offers some Trust Centre functionality
- Trust Centre Link Key (TCLK) is not supported
- Supports following link keys:
  - Distributed Security Global Link Key
  - Touchlink Pre-configured Link Key

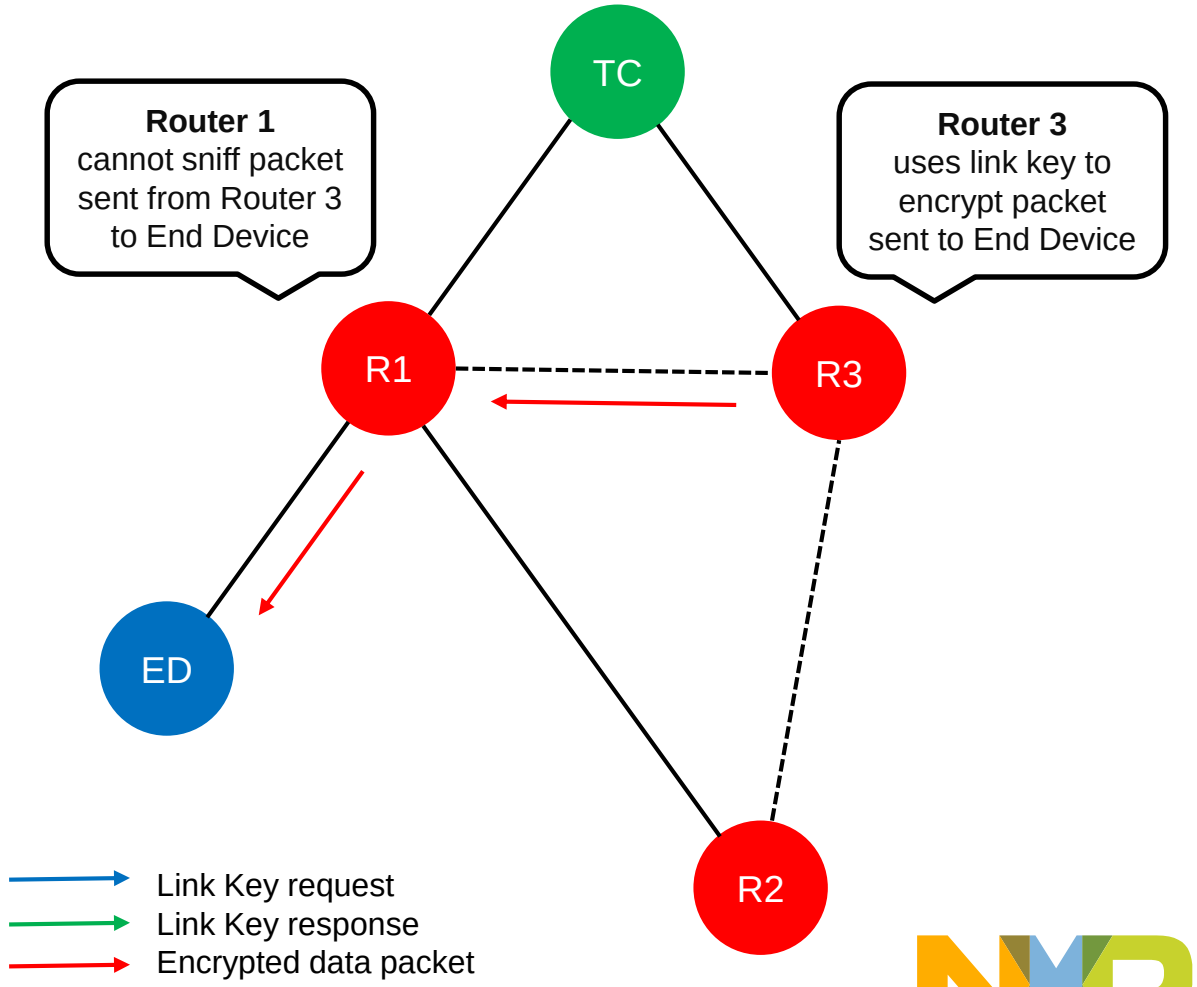


Data Encryption/Decryption in Stack



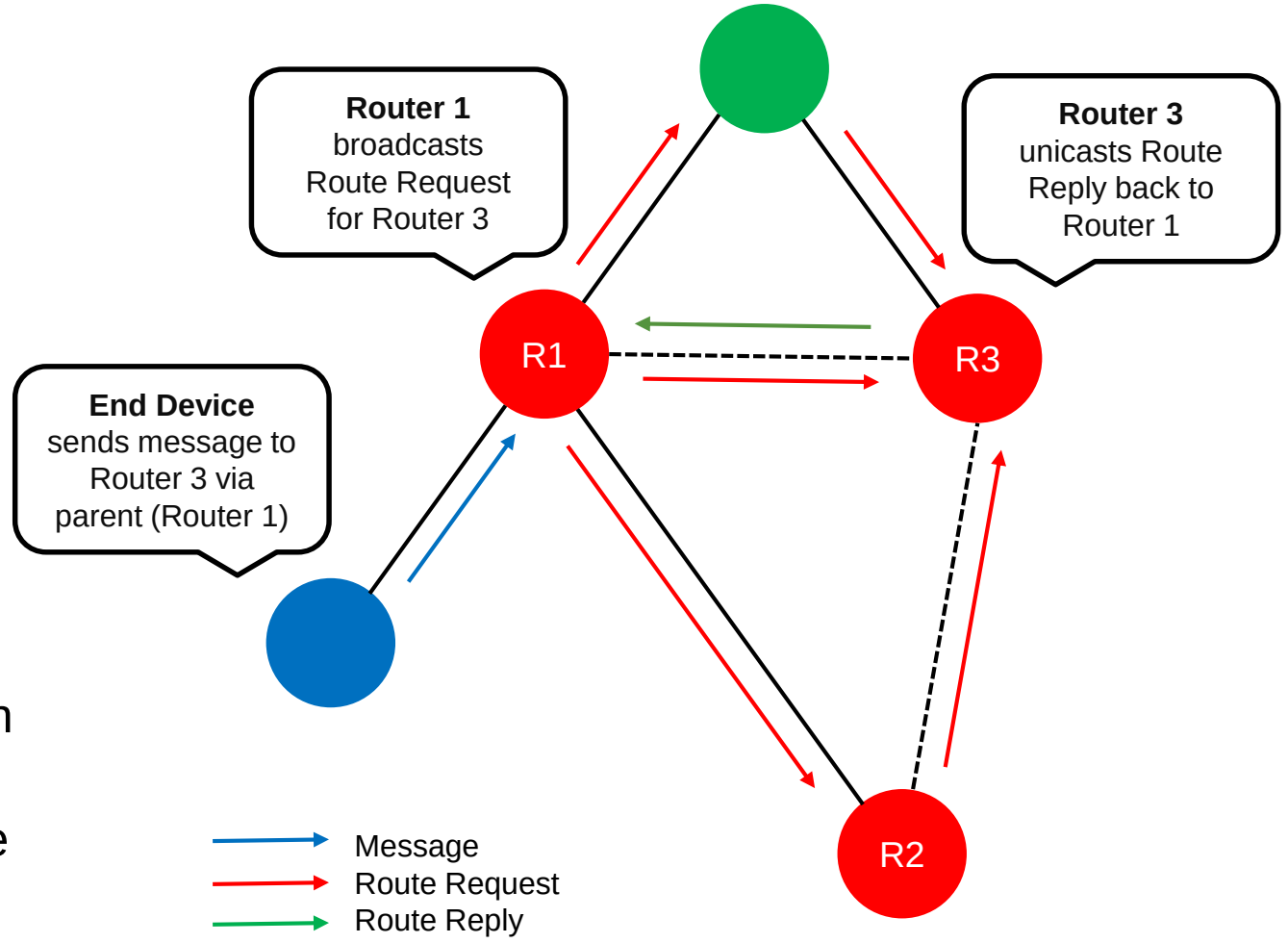
Application-level Security

- Can be set up for a pair of nodes
- Allows private communications between the two nodes, encrypted/decrypted using an application link key
- Link key is unique, exclusive to a pair of nodes
- Link key is issued by the Trust Centre:
  - One node of requests link key from TC
  - TC sends the key to both nodes
  - Transport of the key is encrypted using Network Key and, if applicable, a link key between TC and target node



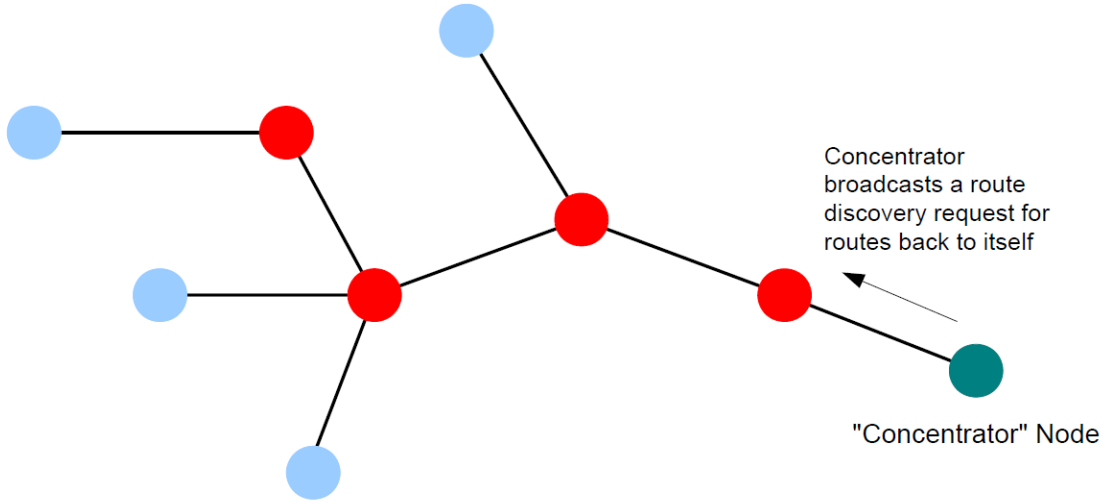
Mesh Network Routing

- Routers (and Coordinator) within radio range can communicate directly
- Routing information held in Routing table on a Router/Coordinator - a table entry contains:
  - Address of destination node
  - Address of ‘next hop’ node to destination
- When sending/forwarding a message, if no entry for destination node in Routing table, Route Discovery is performed:
  - Route Request is broadcast
  - Route Reply is unicast back to originating node
- More than one route from source to destination node is often possible
- Alternative routes may be available in the case of link failure



Many-to-One Routing

- Concentrator broadcasts a route discovery request, the routing tables are updated as the broadcast propagates through the network.
- No responses are generated, the temporary storage of route discovery information is not required and network traffic congestion is minimized.
- Source routing is used - the outward route taken by a message to the concentrator is remembered by the concentrator and embedded in the response message.

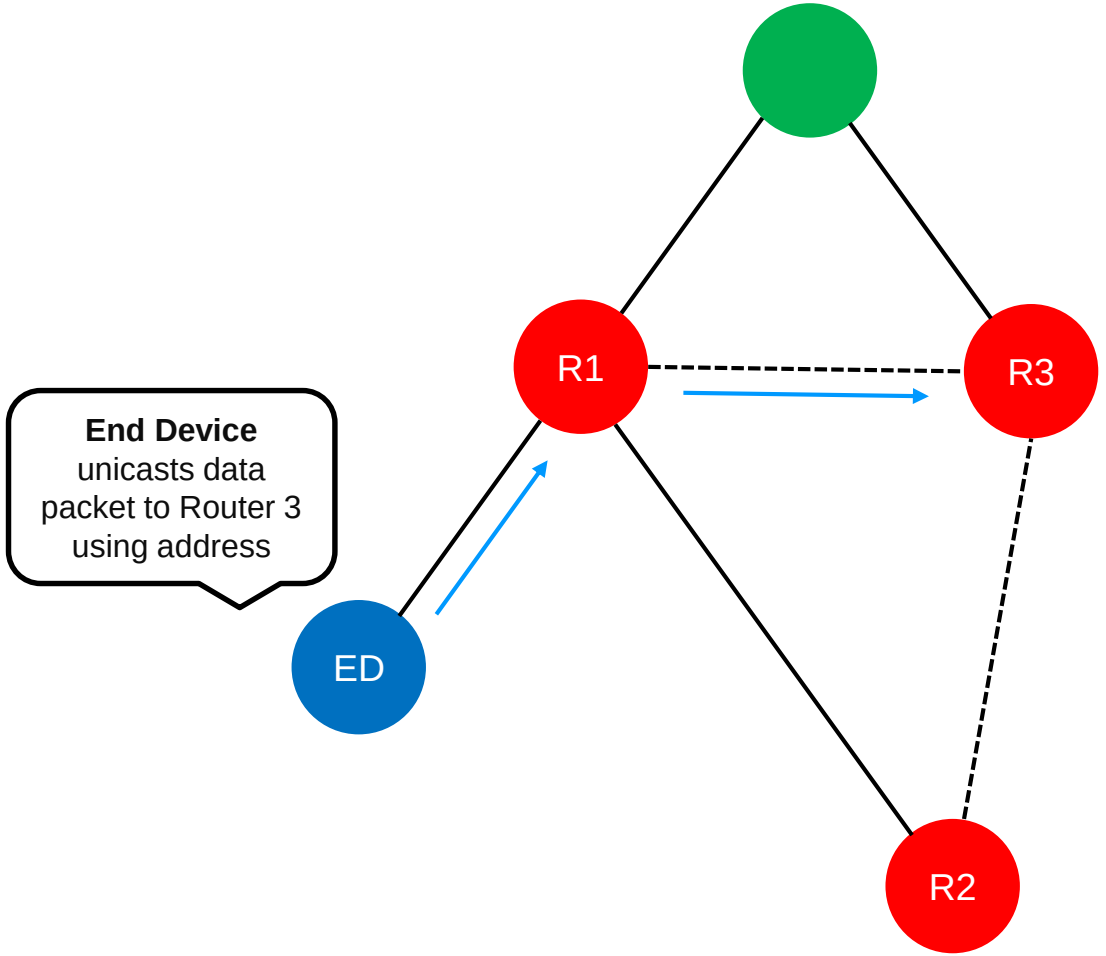


Direct Addressing, Grouping and Binding

Node can communicate with compatible nodes by Direct Addressing, Grouping or Binding:

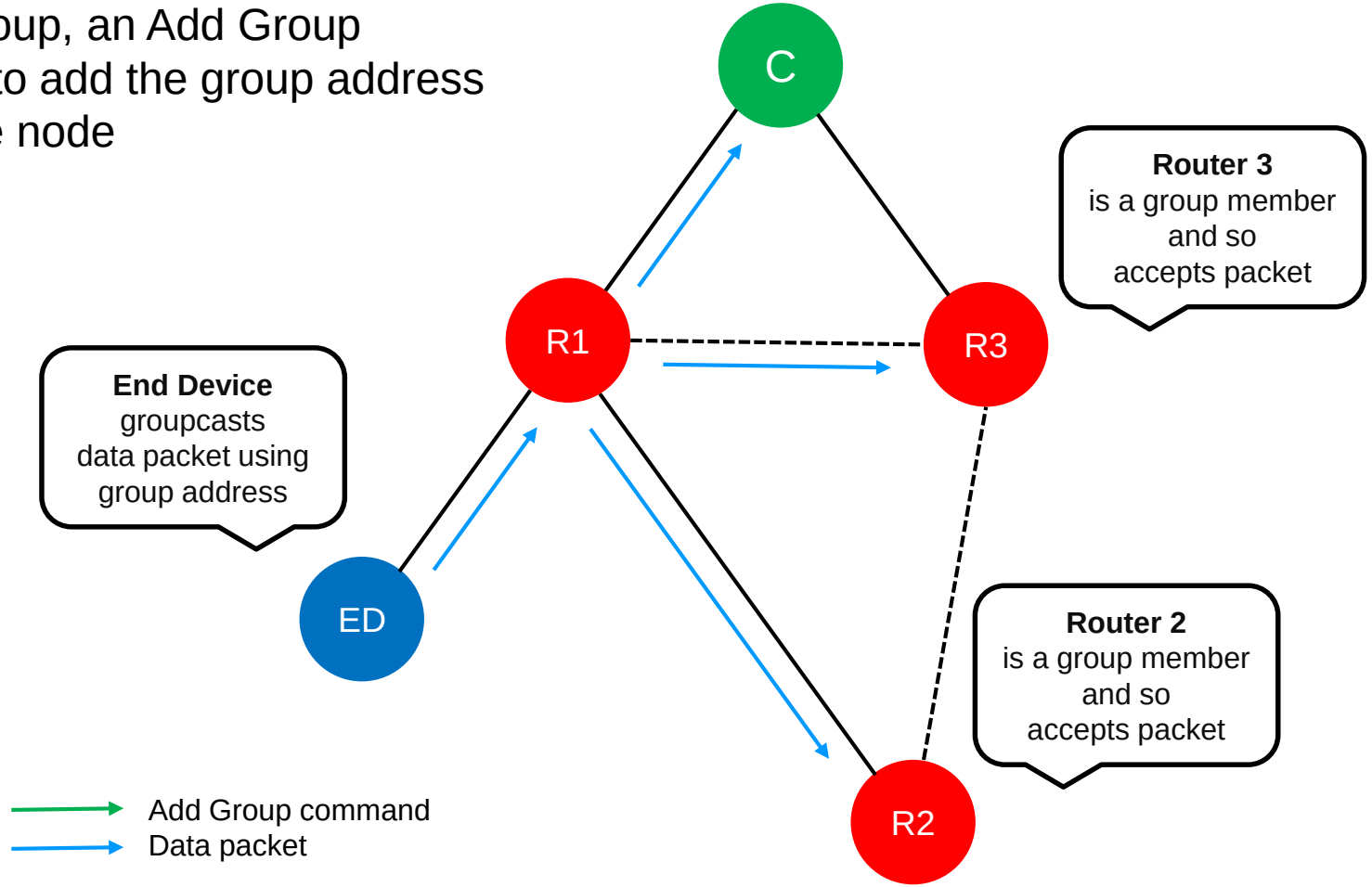
- **Direct Addressing** involves sending a message to individual nodes using their addresses
- **Grouping** involves creating a group of nodes with an associated 16-bit group address:
  - A Group Table is held on each remote (target) node
  - This table contains the addresses of the groups to which the node belongs
  - A message for the group is sent to the appropriate group address
  - When a message for a group is received, it is accepted only if the group address is in the table
- **Binding** involves creating a set of pre-defined links or bindings to remote nodes:
  - Binding links the local endpoint (application) with an endpoint (application) on the remote node
  - A Binding Table containing these links is held on the local (source) node
  - All subsequent communications from the local endpoint are automatically sent to the bound remote endpoint(s)
  - Bindings can be created locally or remotely

Sending Data Packets by Direct Addressing



Sending Data Packets by Group Addressing

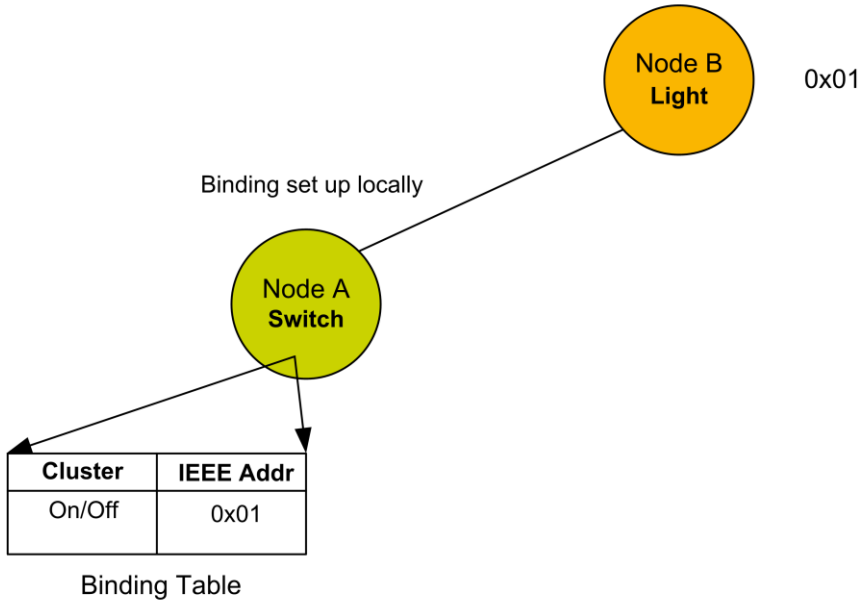
To include a node in a group, an Add Group command must be used to add the group address to the Group Table on the node



Local and Remote Binding

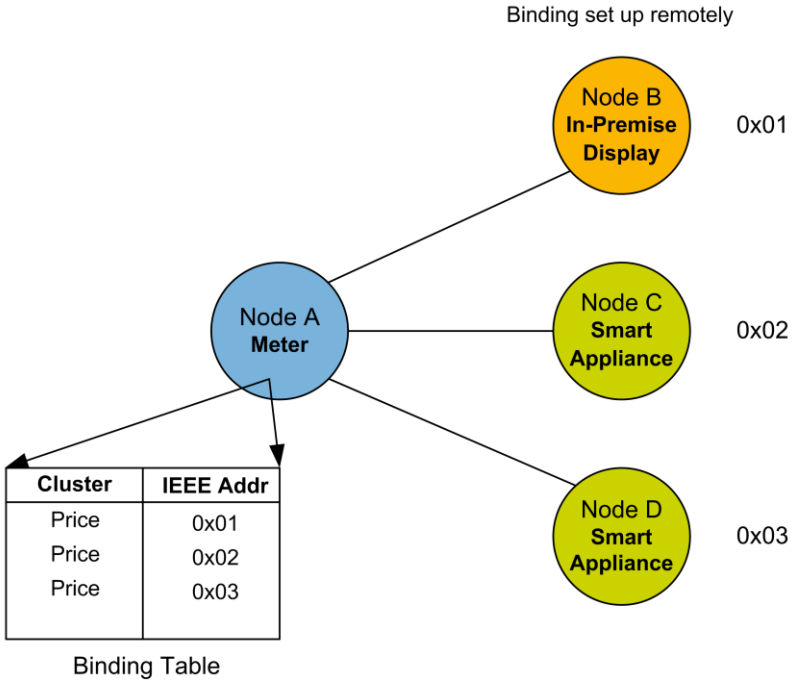
Local Binding

Local device writes details of bound endpoint into its Binding Table.

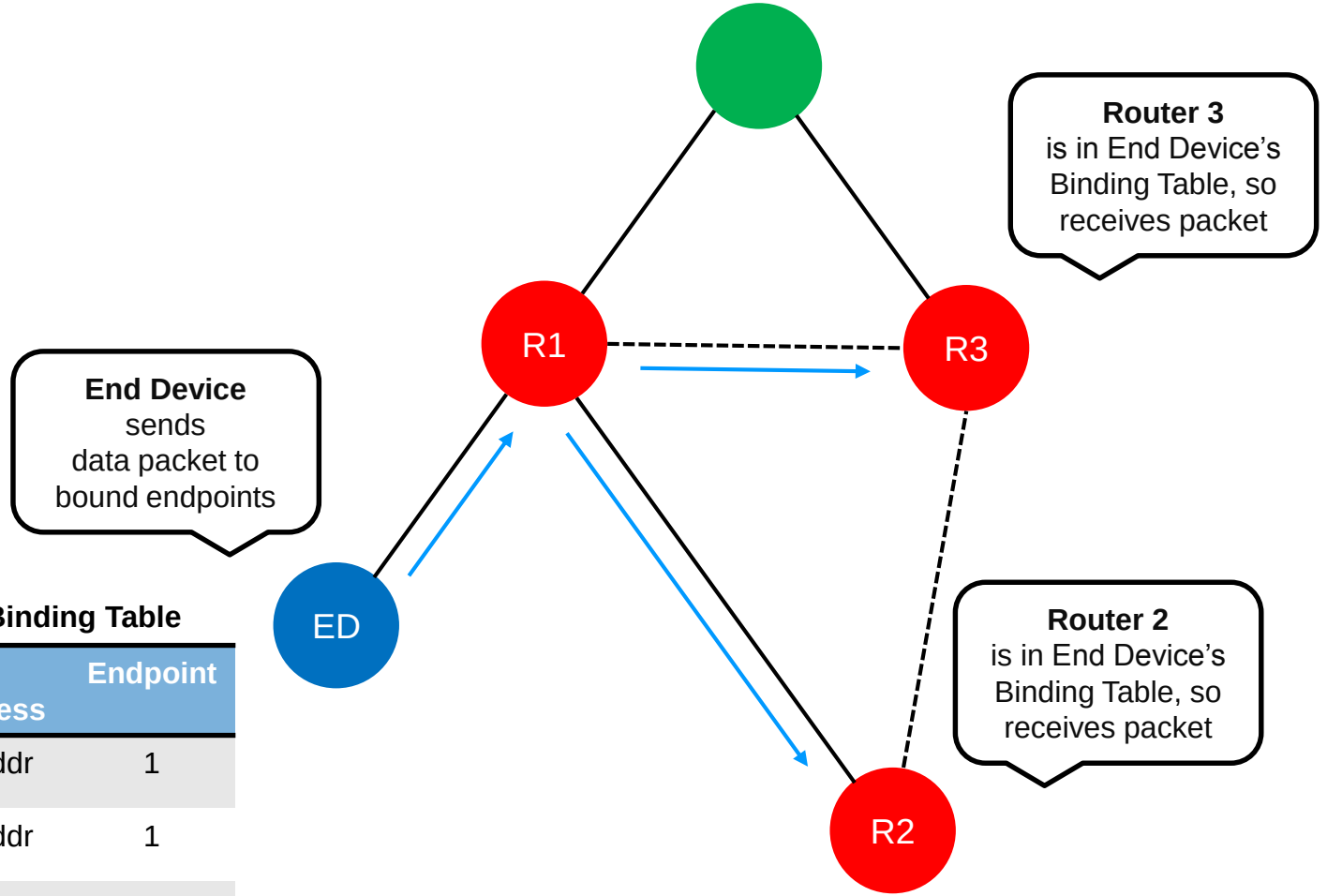


Remote Binding

Local device requests remote devices to write its details into their Binding Tables.



Sending Data Packets to Bound Endpoints



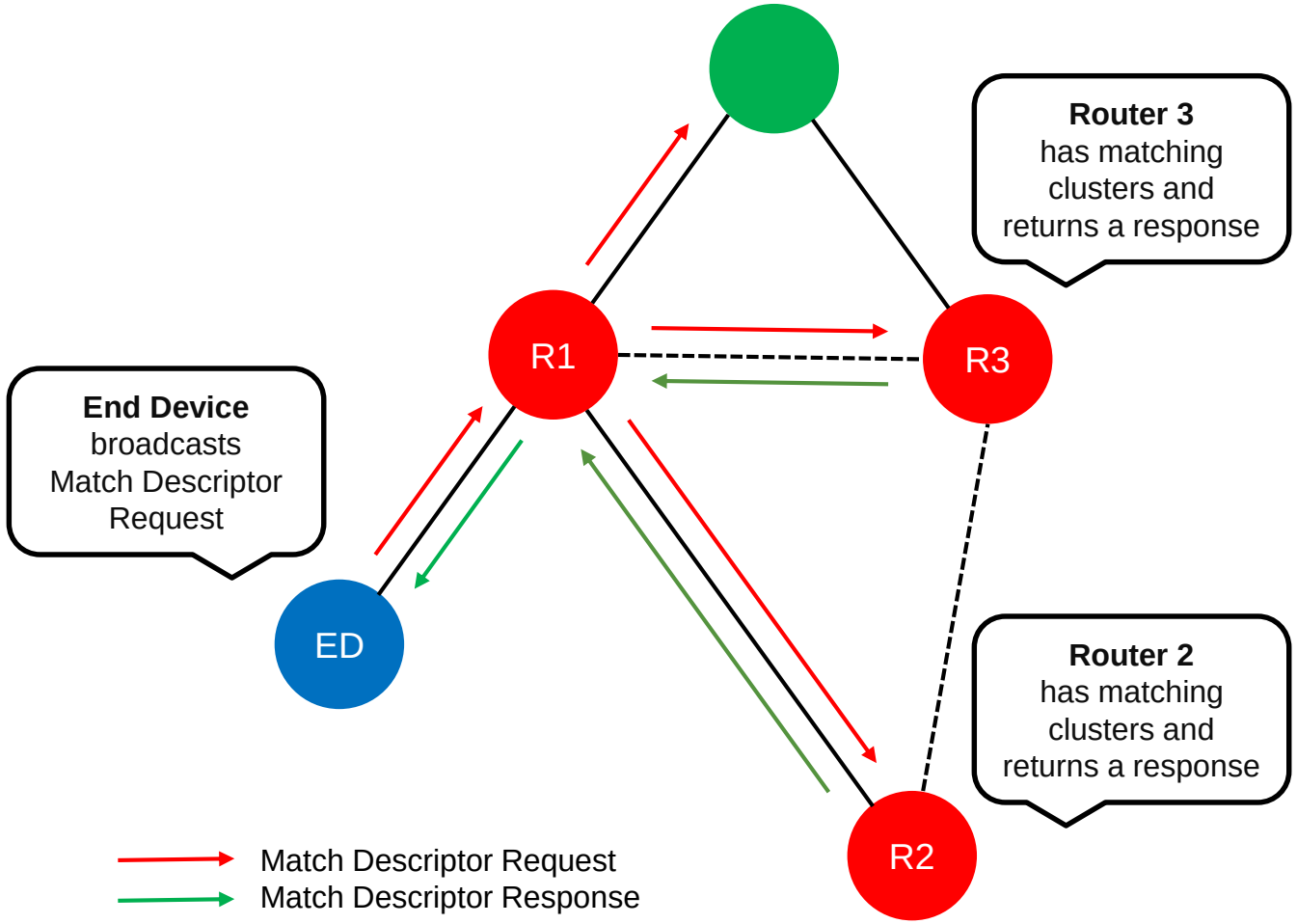
Binding Table

MAC Address	Endpoint
R2 addr	1
R3 addr	1

Service Discovery [1]

- Allows a joining device to find compatible nodes with which to operate (e.g. Dimmer Switch may need to find Dimmable Lights)
- Information about a node is held on the node in 'descriptors'
  - Matching is done in terms of compatible clusters
  - Cluster information is held in the Simple Descriptor on a node
- Joining node broadcasts a Match Descriptor Request listing the clusters of interest
- A node with the relevant clusters unicasts a Match Descriptor Response containing address and endpoint information
  - The joining node may receive multiple responses
  - The application on the joining node must determine which of the responding nodes it will operate with

Service Discovery [2]

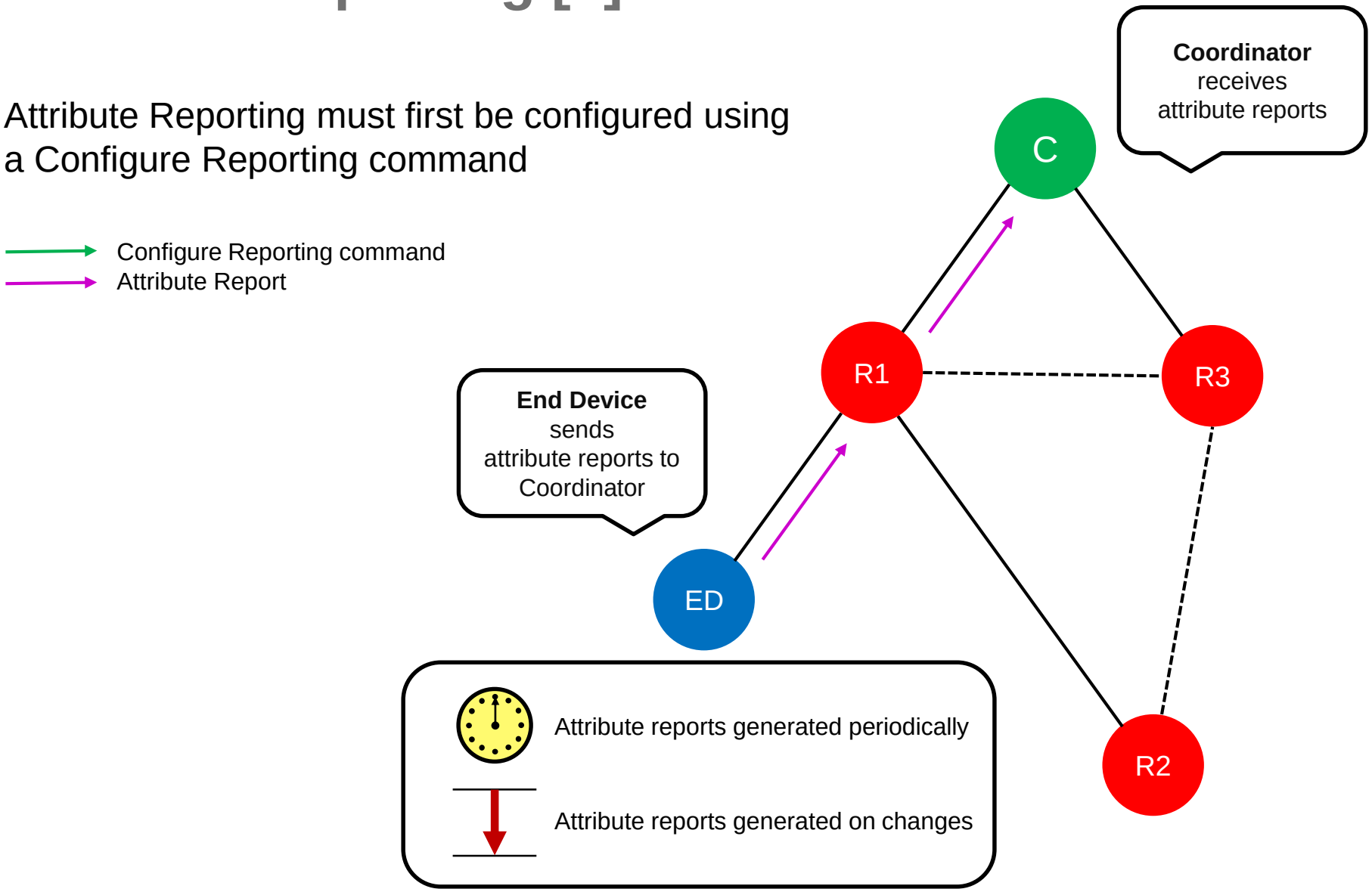


Attribute Reporting [1]

- A node may need to send regular/periodic reports to another node
  e.g. an On/Off Light reporting its state to a Control Bridge (used in IoT)
- Attribute Reporting involves sending attribute values unsolicited from cluster server to client (instead of client polling the server):
  - Reduces network traffic
  - Allows a sleeping server to report attribute values while awake
- An ‘attribute report’ can be issued either:
  - by a function call in the user application (on the server device)
  - automatically by the ZCL
- Automatic attribute reporting involves two mechanisms:
  - Reports are triggered by changes in the attribute values
  - Reports are issued for the attributes periodically
  - These mechanisms can operate at the same time

Attribute Reporting [2]

Attribute Reporting must first be configured using a Configure Reporting command



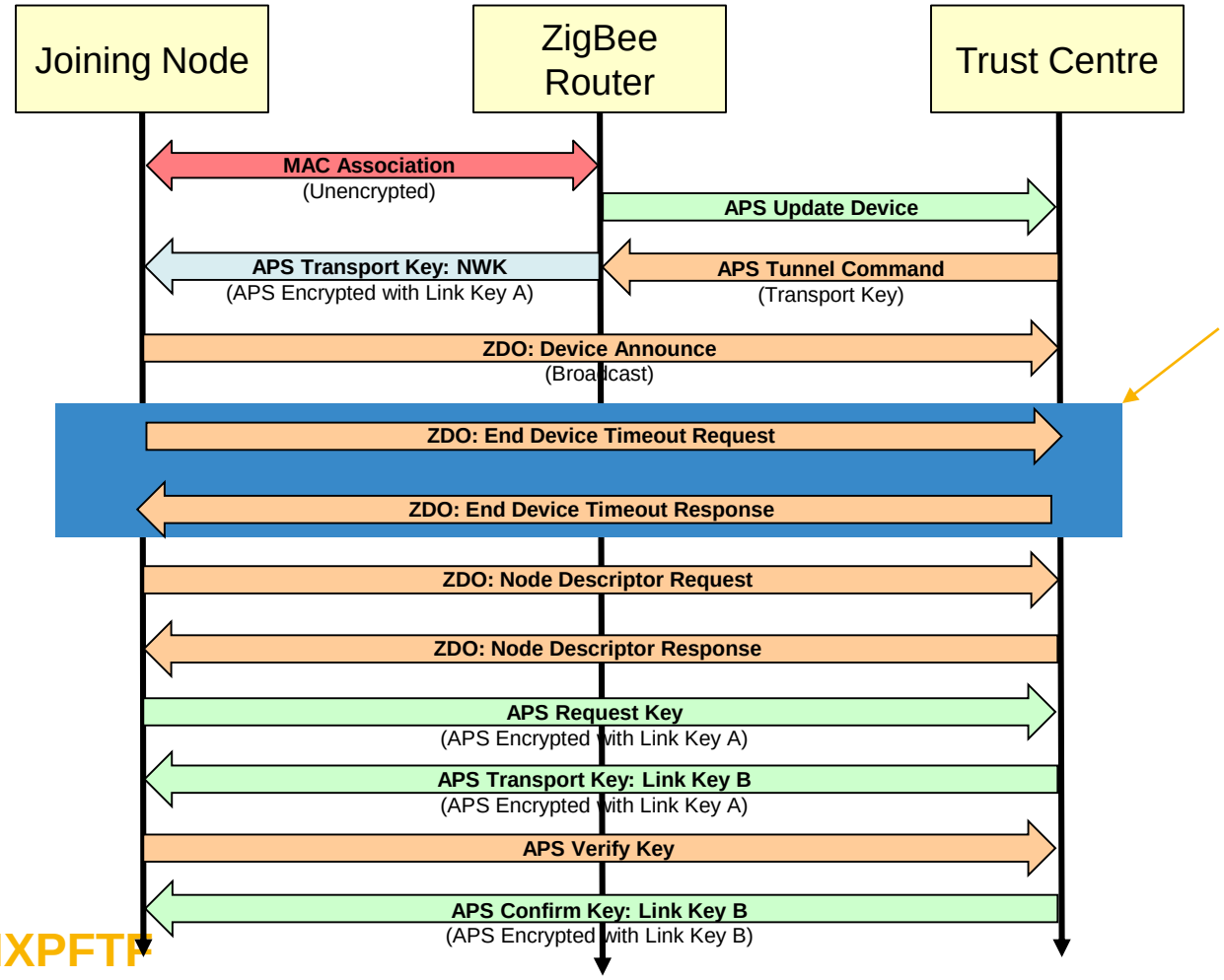
End Device Aging

This is a method of removing inactive End Device children from the parent's Neighbour Table (and therefore removing them from the network).

- **In ZigBee PRO R20:**
  - Child ageing is not handled in ZigBee PRO R20 - it is application/vendor-specific
  - Stack tables are only cleared when a device leaves - nothing happens at the stack layers if a device is just removed
- **In ZigBee PRO R21:**
  - Upon joining, the End Device provides the parent with its timeout period
  - If the End Device does not contact the parent within 3 timeout periods, the device will be expelled from the network.
  - Trust Centre Link Key (TCLK) has added as a secure way of getting these nodes back into the network – switch key

End Device Ageing: Overview

When a ZigBee 3.0 End Device joins the network, it sends additional information to it's parent to indicate how often it will poll.



End Device Timeout Request command is sent from end devices to inform the parent when it should age the device out.

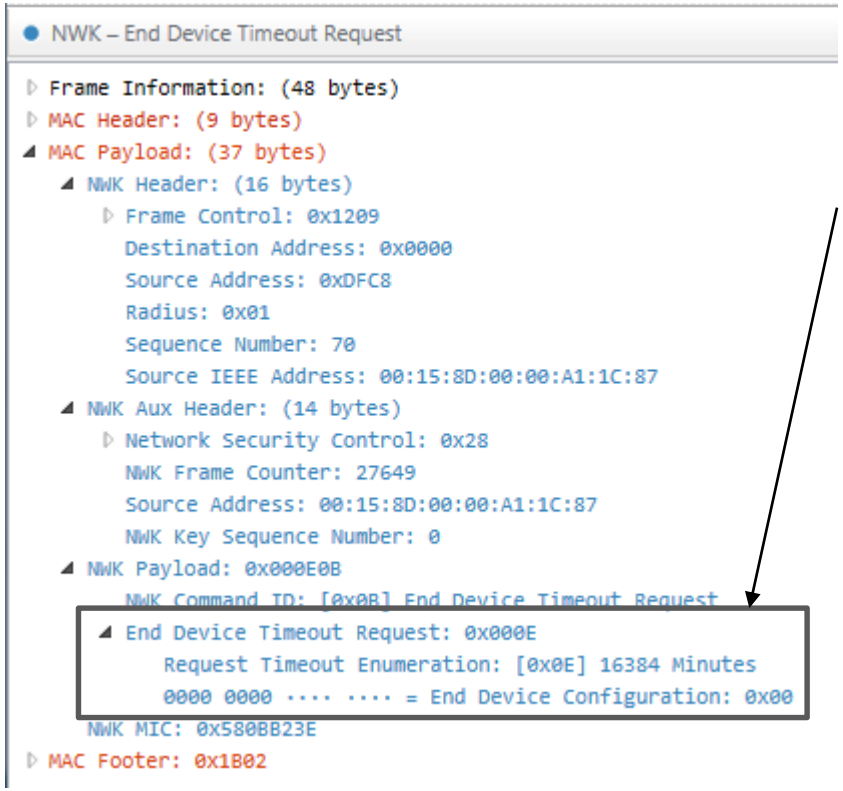
End Device Ageing: Overview

The below sniffer trace shows the flow of joining which has the End Device Timeout Request. This shows the previous diagram in practice up to the end device timeout response.

28	14:33:10.610570	0.027808	20	ZigBee	NWK	Beacon	0x0000		64	
21	14:33:10.732362	0.121792	20	ZigBee	MAC	Association Request	00:15:…	0x0000	165	
5	14:33:10.733418	0.001056	20	ZigBee	MAC	Acknowledgement			165	
50	14:33:10.984634	0.251216	20	ZigBee	NWK	Command	0xD6AF	0xFFFF	133	
18	14:33:11.234218	0.249584	20	ZigBee	MAC	Data Request	00:15:…	0x0000	166	
5	14:33:11.235178	0.000960	20	ZigBee	MAC	Acknowledgement			166	
27	14:33:11.245866	0.010688	20	ZigBee	MAC	Association Response	00:15:…	00:15:8…	58	
5	14:33:11.247114	0.001248	20	ZigBee	MAC	Acknowledgement			58	
74	14:33:11.448298	0.201184	20	ZigBee	NWK	Command	0x0003	0xFFFF	129	
12	14:33:11.469754	0.021456	20	ZigBee	MAC	Data Request	0xDFC8	0x0000	167	
5	14:33:11.470522	0.000768	20	ZigBee	MAC	Acknowledgement			167	
73	14:33:11.481210	0.010688	20	ZigBee	APS	Transport Key	0x0000	0xDFC8	59	
5	14:33:11.483930	0.002720	20	ZigBee	MAC	Acknowledgement			59	
57	14:33:11.536570	0.052640	20	ZigBee	ZDP	Device Announce	0xDFC8	0x0000	168	
5	14:33:11.538778	0.002208	20	ZigBee	MAC	Acknowledgement			168	
48	14:33:11.542570	0.003792	20	ZigBee	NWK	End Device Timeout Request	0xDFC8	0x0000	169	
5	14:33:11.544490	0.001920	20	ZigBee	MAC	Acknowledgement			169	
51	14:33:11.567258	0.022768	20	ZigBee	NWK	Route Request	0x0000	0xFFFF	60	
57	14:33:11.611754	0.044496	20	ZigBee	ZDP	Device Announce	0x0000	0xFFFF	61	
12	14:33:11.658138	0.046384	20	ZigBee	MAC	Data Request	0xDFC8	0x0000	170	
5	14:33:11.658906	0.000768	20	ZigBee	MAC	Acknowledgement			170	
48	14:33:11.669594	0.010688	20	ZigBee	NWK	End Device Timeout Response	0x0000	0xDFC8	62	

End Device Ageing: Overview

The payload of the End Device Timeout Request is shown to the left and the Request Timeout Enumeration table is on the right.



End device is indicating it should be timed out every 16384 minutes (~11 days). This means it will be aged out if no keepalive message is received within this time period.

The end device configuration bits are reserved for future modifications and are currently not used.

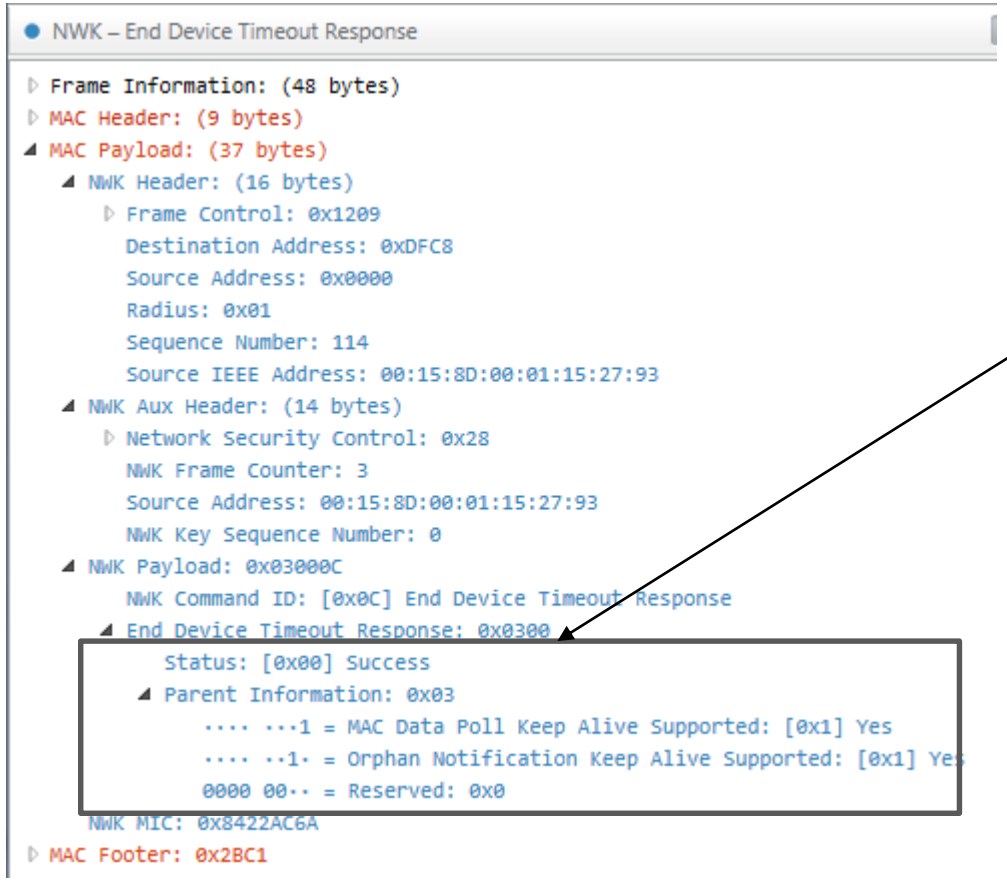
Table 3.44 Requested Timeout Enumerated Values

Requested Timeout Enumeration Value	Actual Timeout Value
0	10 seconds
1	2 minutes
2	4 minutes
3	8 minutes
4	16 minutes
5	32 minutes
6	64 minutes
7	128 minutes
8	256 minutes
9	512 minutes
10	1024 minutes
11	2048 minutes
12	4096 minutes
13	8192 minutes
14	16384 minutes

The Timeout is defined as an enumeration meaning it cannot be configured to a custom value. A map table is listed to show which enumeration value relates to timeout.

End Device Ageing: Overview

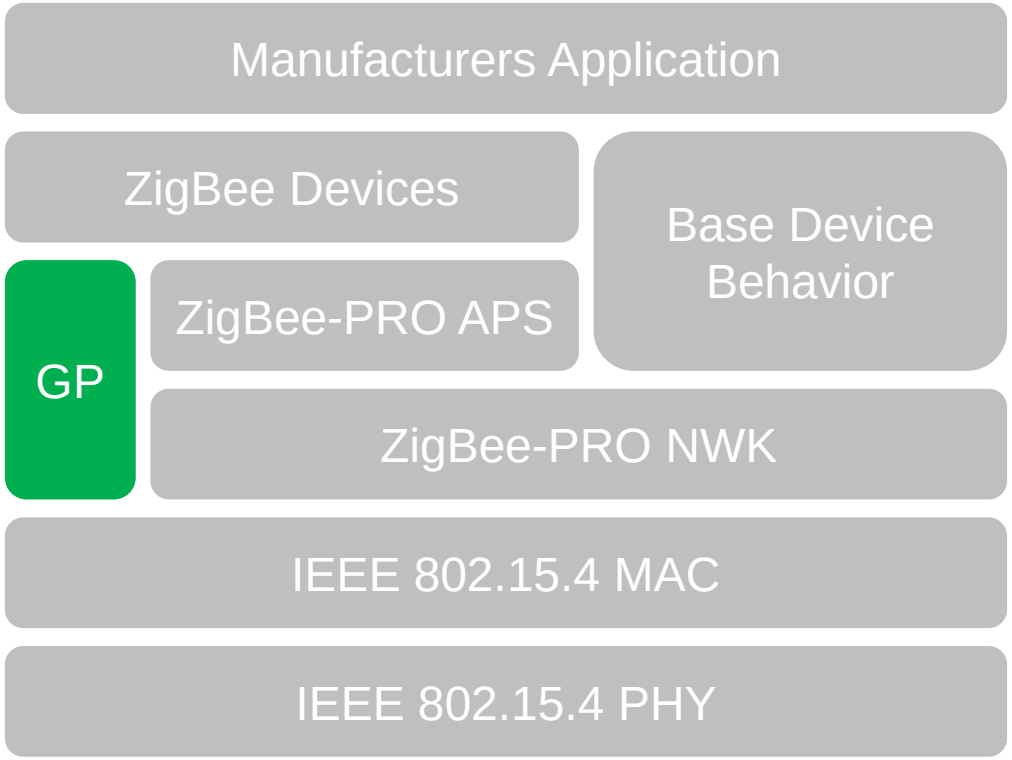
The payload of the End Device Timeout Response is shown to the left.



The End Devices parent responds with information on what keepalive commands it supports. Our default stack support both keepalive mechanisms via polls or keepalive notifications.

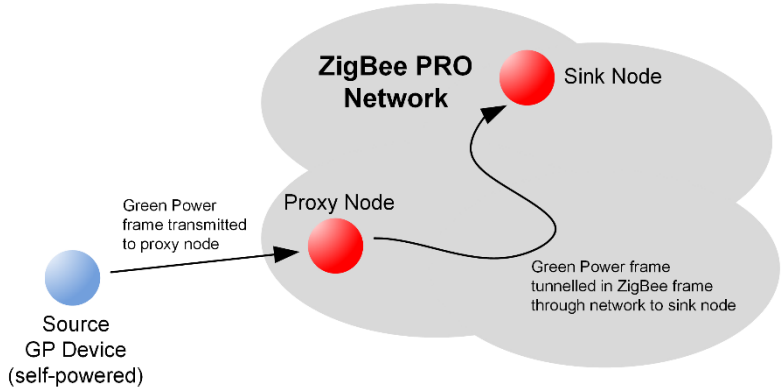
Additionally, there are also some reserve bits for forward compatibility.

ZigBee Green Power



ZigBee Green Power [1]

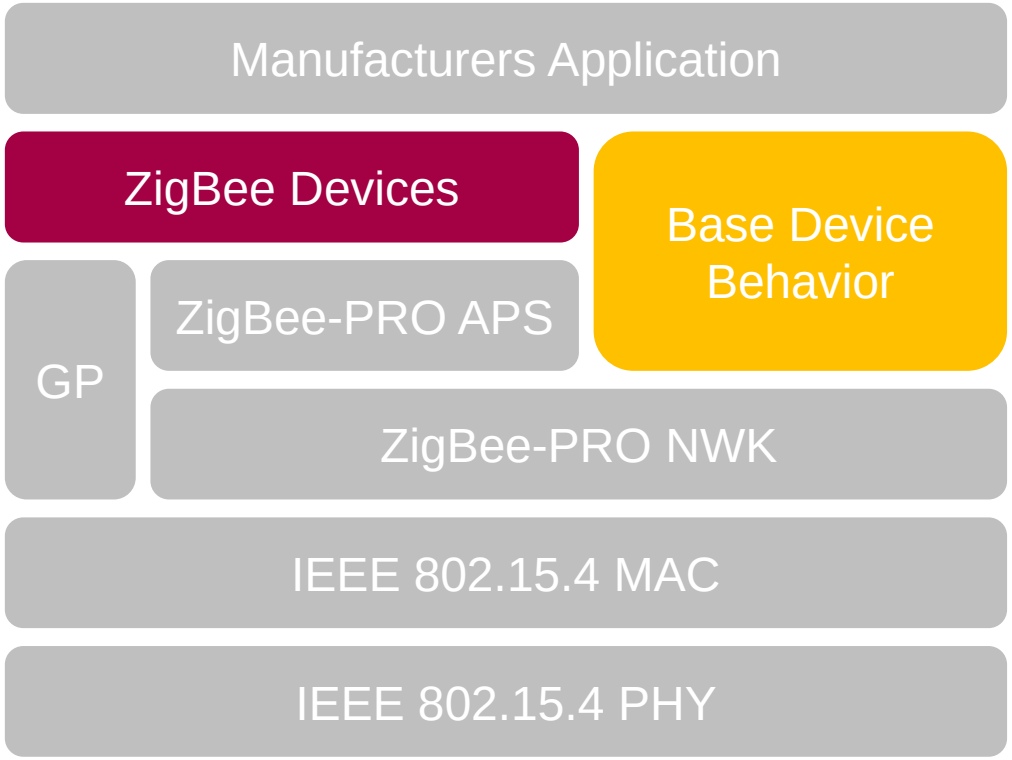
- ZigBee PRO R21 stack includes support for the ZigBee Green Power (GP) cluster, allowing a ZigBee 3.0 node to act as a:
  - **GP Proxy node** which can receive GP frames from a source GP device (e.g. Energy Harvesting switch) and forward a GP frame inside a normal ZigBee frame into the ZigBee network
  - **GP Sink node** which receives and processes a GP frame (embedded in a normal ZigBee frame)



ZigBee Green Power [2]

- A ZigBee 3.0 node may support the GP ‘Combo’ device, which allows it to act as both a proxy node and a sink node.
- GP frames are short IEEE802.15.4 frames that minimise transmission time and the required power
- The GP device that sends GP frames is not a full member of the ZigBee network and runs a reduced GP stack
- The GP stack may operate over an adapted IEEE802.15.4 MAC layer

ZigBee Devices



Base Device Features

Every ZigBee 3.0 node must employ a standard device type called the ZigBee Base Device, which handles fundamental operations such as commissioning

Feature	ZC	ZR	ZED
Network Formation Centralised	✓	✗	✗
Network Formation Distributed	✗	✓	✗
Network Steering (Find Open & Opening Networks)	✓	✓	✓
Join by Installation Codes	✓	✗	✗
Touchlink Joining	✓	✓	✓
Finding and Binding (Commissioning)	✓	✓	✓
Basic and Identify Clusters	✓	✓	✓

Network Forming / Joining

Network Commissioning

Key	
✓	Mandatory
✓	Optional
✗	Not Supported

ZigBee Lighting & Occupancy (ZLO)

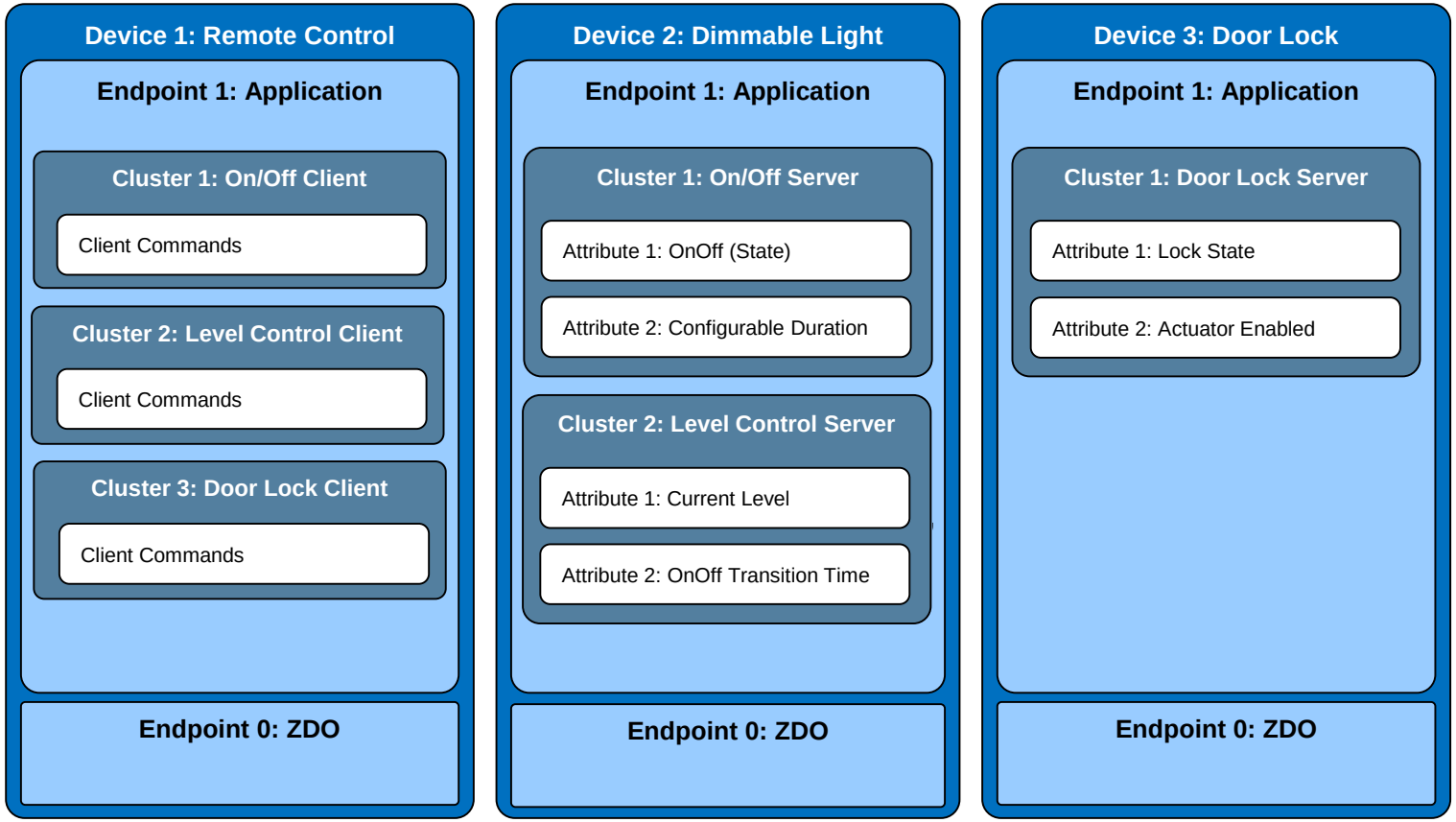
- First set of devices to be part of ZigBee 3.0
- Incorporates devices from ZLL and HA profiles
- No longer have ZLL and HA profiles, but HA Profile Id (0x0104) used
- Application Version Id now 1 (was 2)
- Some devices have new Device Ids – for example, Extended Colour Light is now 0x010D
- Legacy Controller will need updating to recognise the new Device Ids
- No longer a separate Touchlink endpoint - all clusters on a single Application endpoint
- ZLL/HA clusters included in ZCL R6 and modified for interoperability

Device Types

- ZigBee 3.0 defines a number of device types
- A device type is:
  - a software entity corresponding to a functional device (e.g. Dimmable Light)
  - implemented as an application with a corresponding endpoint on the node
- A device type is supported by a number of functional ‘clusters’:
  - Some clusters are mandatory (e.g. Basic cluster) and some are optional
  - Some clusters are profile-specific
  - Some clusters are shared by profiles and come from the ZigBee Cluster Library (ZCL)
- The required clusters are enabled in the device structure
- More than one device type (or multiple instances of the same device type) can be implemented on the same node (in different applications on different endpoints)

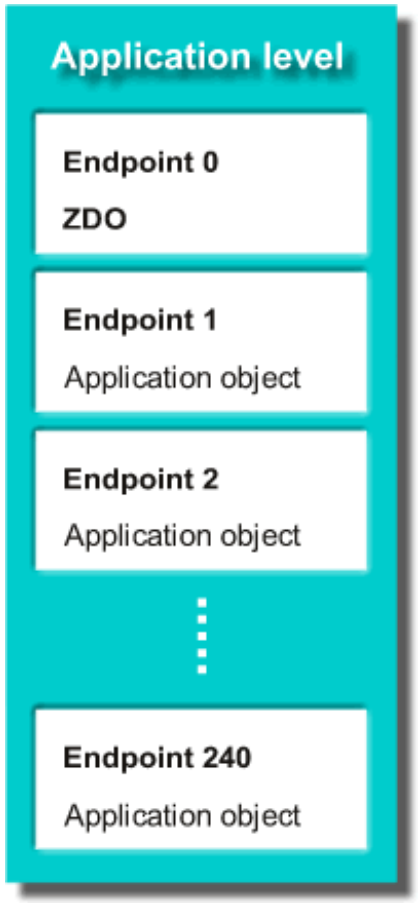
Device Type	Device ID
On/Off Light	0x0100
Dimmable Light	0x0101
Colour Dimmable Light	0x0102
On/Off Light Switch	0x0103
Dimmer Switch	0x0104
Colour Dimmer Switch	0x0105
Light Sensor	0x0106
Occupancy Sensor	0x0107
On/Off Ballast	0x0108
Dimmable Ballast	0x0109
On/Off Plug-in Unit	0x010A
Dimmable Plug-in Unit	0x010B
Colour Temperature Light	0x010C
Extended Colour Light	0x010D
Light Level Sensor	0x010E
Colour Controller	0x0800
Colour Scene Controller	0x0810
Non-Colour Controller	0x0820
Non-Colour Scene Controller	0x0830
Control Bridge	0x0840
On/Off Sensor	0x0850

ZigBee Device Architecture



Endpoints

- A device may run multiple applications or multiple instances of the same application
- An endpoint is a software entity providing a communication port for an application instance
- Endpoint number is used to identify the destination application when sending a message to a node
- 256 endpoints, numbered 0 to 255:
  - 0 is reserved for ZDO (ZigBee Device Objects)
  - 1 to 240 are available for application instances
  - 241 to 254 are reserved, with 242 reserved for ZigBee Green Power
  - 255 is used to broadcast to all application instances



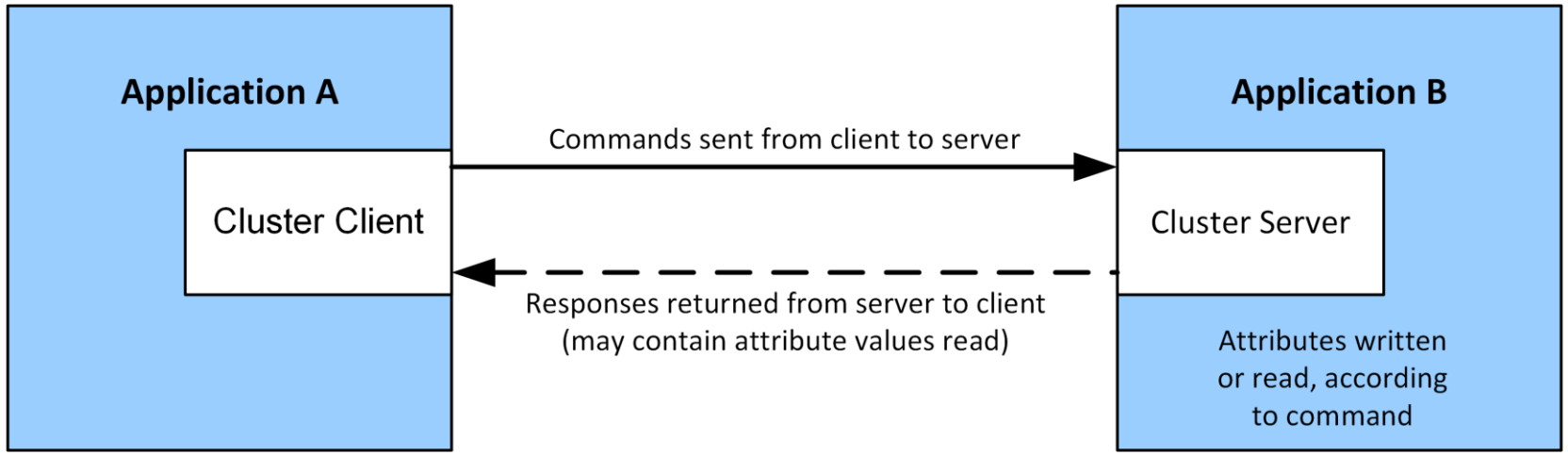
Clusters and Attributes

- A **cluster** is the smallest functional building block, corresponding to specific functionality (e.g. On/Off Switch, Dimmable Light, Thermostat)
- A cluster comprises attributes and the commands used to interact with them
- An **attribute** is a data entity (e.g. temperature) which is held on the device and may be communicated between devices
- For example, the Thermostat cluster contains attributes relating to:
  - Current temperature
  - Minimum temperature
  - Maximum temperature
- ZigBee Alliance have defined standard clusters:
  - Set of common clusters collected together in the ZigBee Cluster Library (ZCL)
  - Specific clusters as part of a public application profile, e.g. Smart Energy
- A cluster has two forms – client and server – that exist on the two communicating devices

Cluster Servers and Clients

A cluster has two forms, server and client:

- **Cluster Server:**
  - Used to store attributes and receive commands to manipulate them
- **Cluster Client:**
  - Used to manipulate attributes in the corresponding cluster server by sending commands (normally ‘write’ commands to set attribute values and ‘read’ commands to obtain attribute values)



BACKWARDS COMPATIBILTY

ZigBee 3.0 Key Descriptions

- The naming of the security keys differs between HA/ZLL profiles & ZigBee 3.0

ZigBee 3.0	HA/ZLL
Default Global Trust Centre Key	HA Default Key
Distributed Trust Centre Key	ZLL Master Key
Touch Link Key	ZLL Master Key

Legacy Devices Joining ZigBee 3.0 Networks

HA Device

- Can only classically join a ZigBee 3.0 network using the: -
  - Default Global Trust Centre Link Key

ZLL Device

- Can classically join the ZigBee 3.0 network using either the: -
  - Default Global Trust Centre Link Key ‘or’
  - Distributed Trust Centre Key
- Can join the ZigBee 3.0 network via Touch Link using: -
  - Distributed Trust Centre Key

Note: Both devices will be allocated the default ‘End Device Timeout Period’, as they will not provide one post join

ZigBee 3.0 Devices Joining Legacy Networks

HA Network

- Can only classically join a HA network using the: -
  - Default Global Trust Centre Link Key

ZLL Network

- Can classically join the ZLL network using either the: -
  - Default Global Trust Centre Link Key ‘or’
  - Distributed Trust Centre Key
- Can join the ZLL network via Touch Link using:
  - Distributed Trust Centre Key

Note: “After joining the network, a ZigBee 3.0 device will determine the type of network it has joined by querying the TC or its parents ‘node’ and ‘simple’ descriptors”

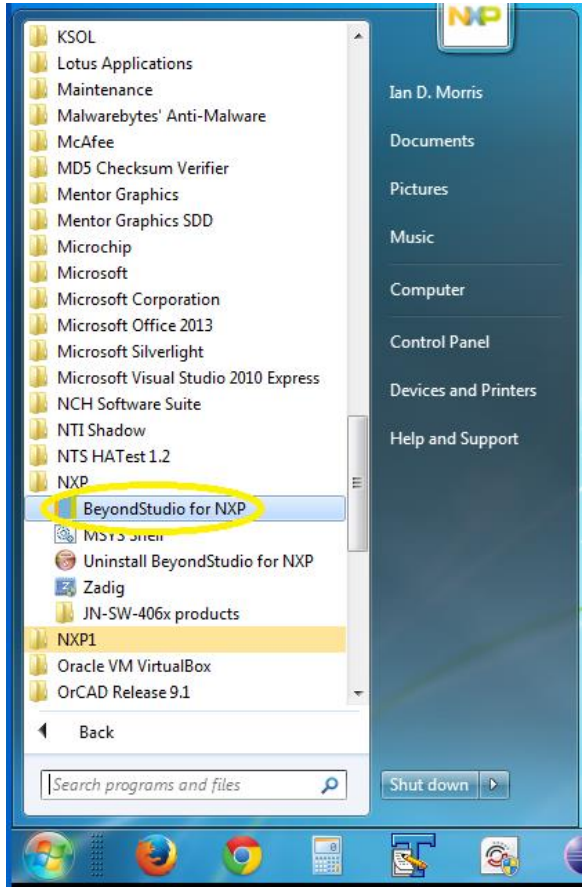
Future ZigBee 3.0 Developments

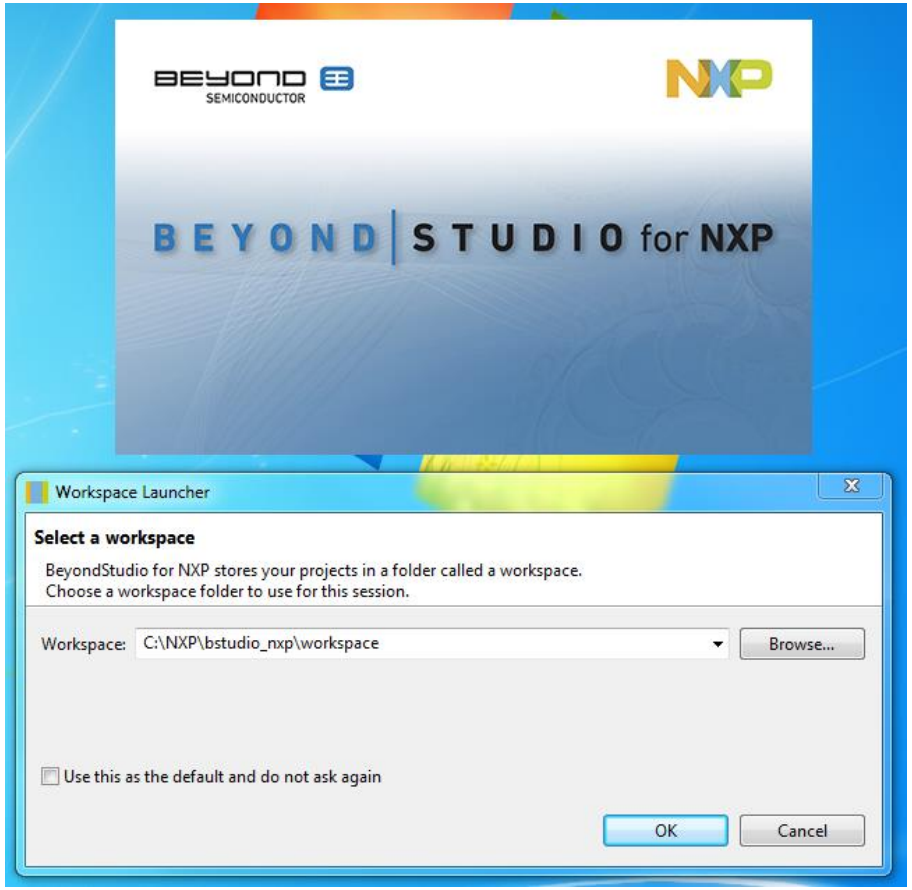
- Other profiles (e.g. Health Care, Building Automation) will adopt ZigBee 3.0 when the relevant workgroups have the necessary resources
- Smart Energy profile resisting move to ZigBee 3.0 due to security risks:
  - ZigBee Smart Energy uses advanced security based on elliptical curve cryptography, specifically implemented for use by electric utilities to enable a highly secured smart grid
  - This level of security will be integrated as an optional feature of ZigBee 3.0 in the future, allowing Smart Energy to be incorporated into ZigBee 3.0

HANDS ON MODULES: PREPARATION

Starting the Eclipse IDE

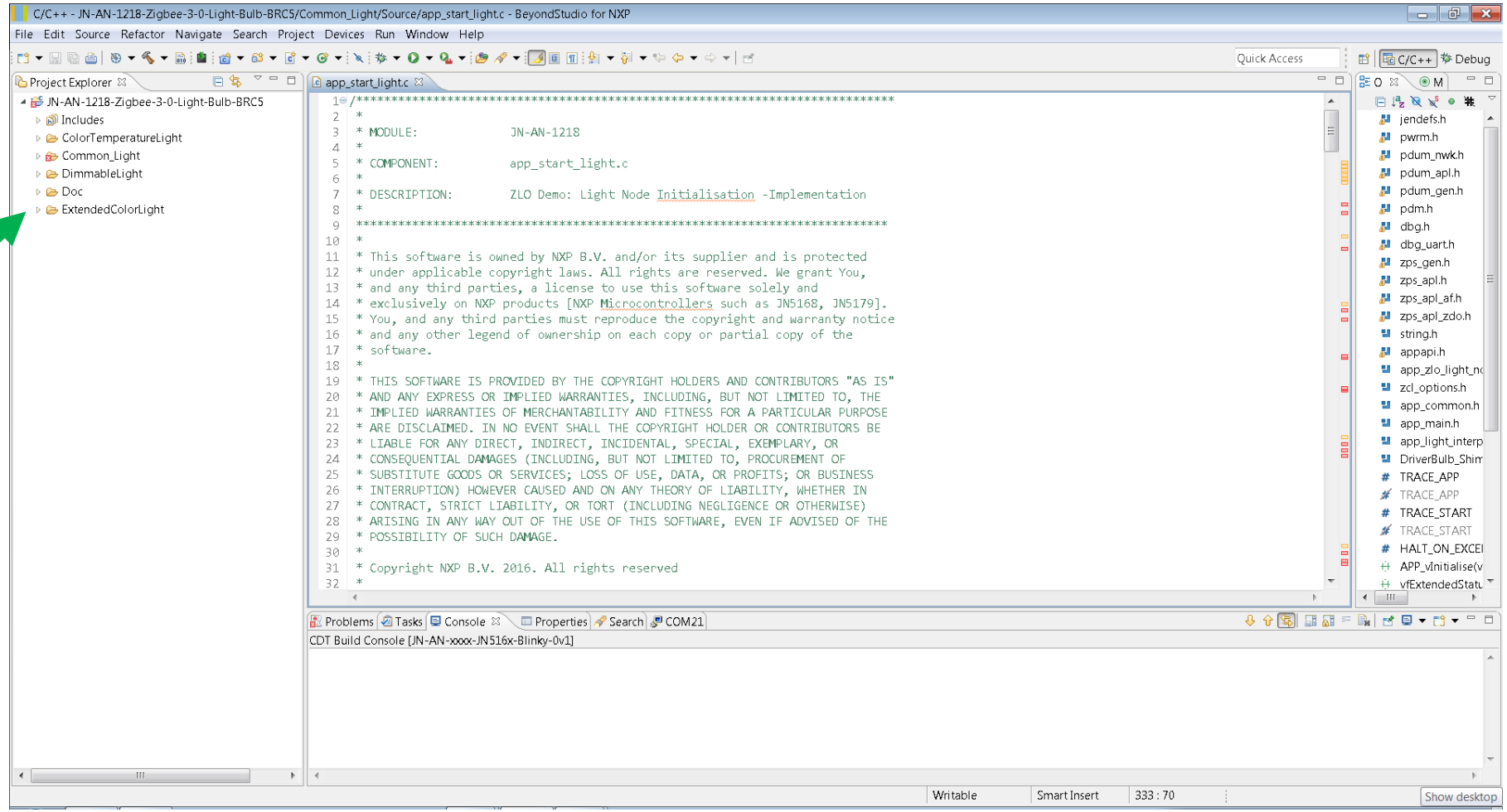
Select “BeyondStudio for NXP” using the Windows Start Menu





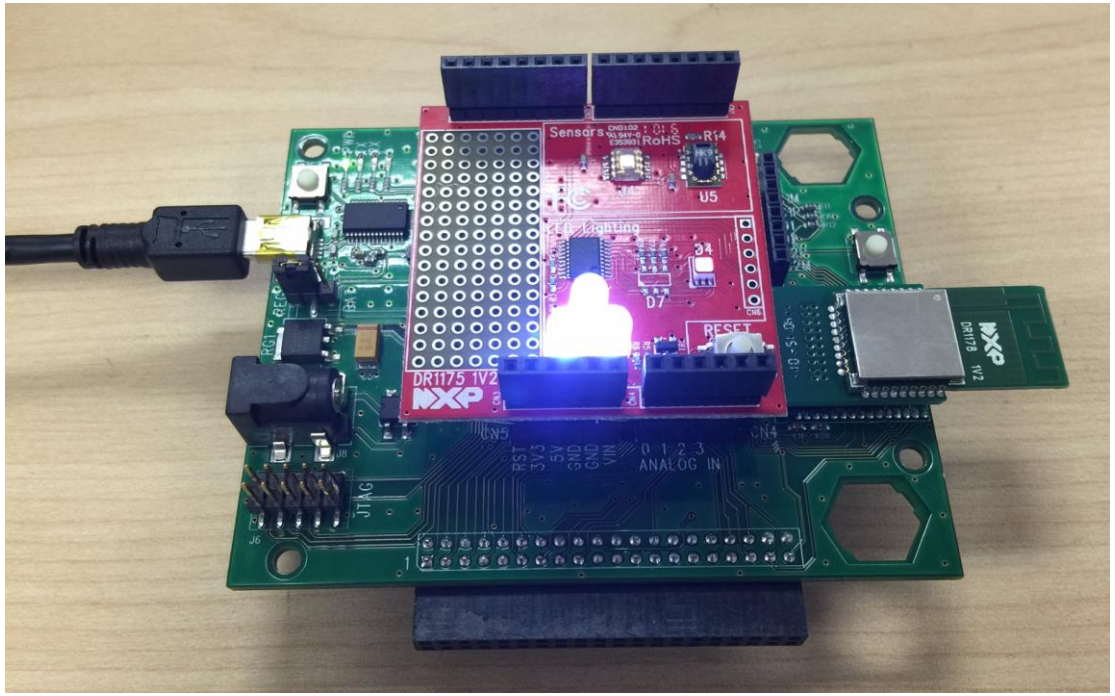
ZigBee Application Note

ZigBee Application Note will be present in the Project Explorer window



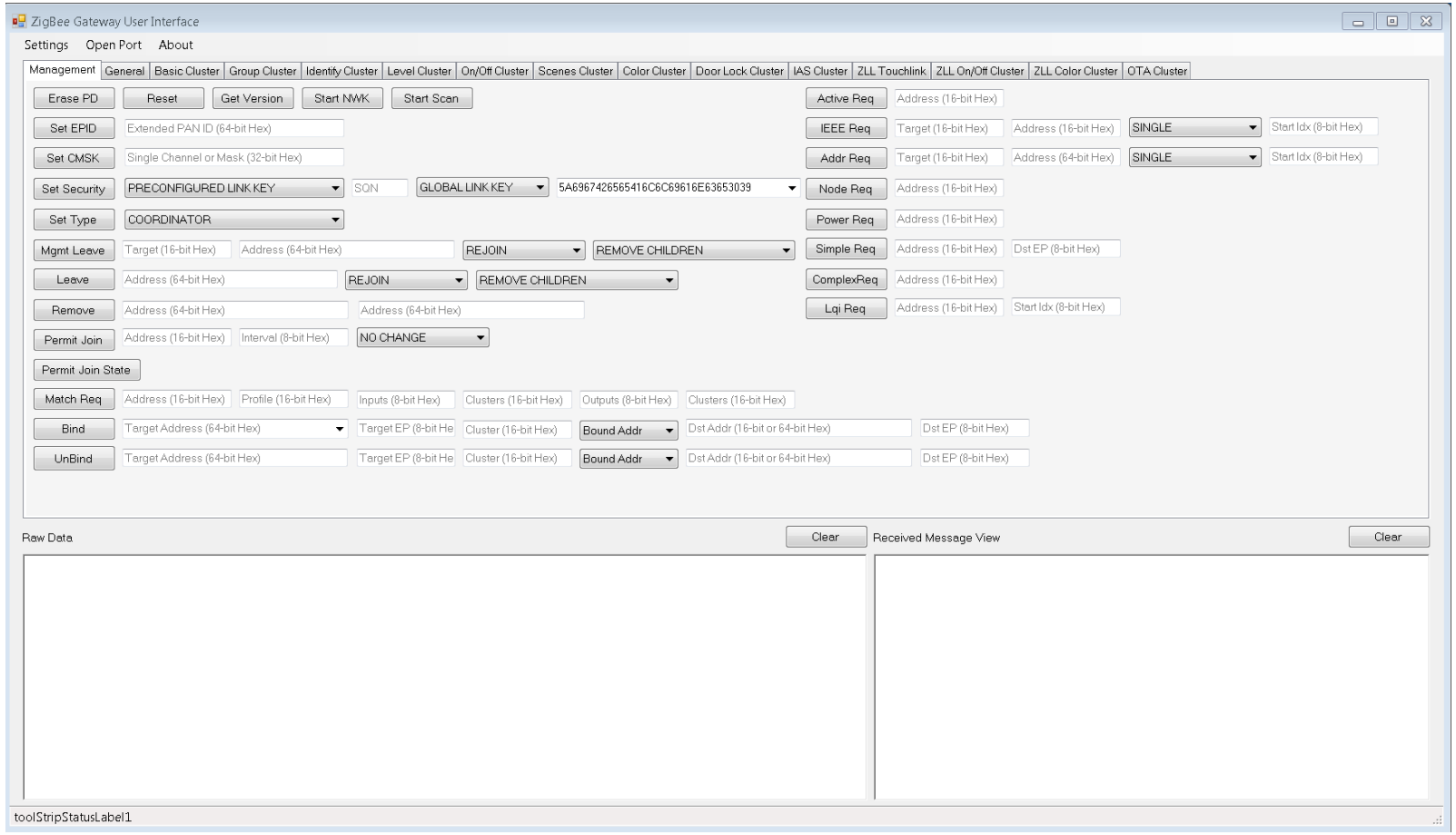
Hardware

- Your hardware should be connected as follows:



ZigBee Gateway User Interface (ZGWUI) Tool

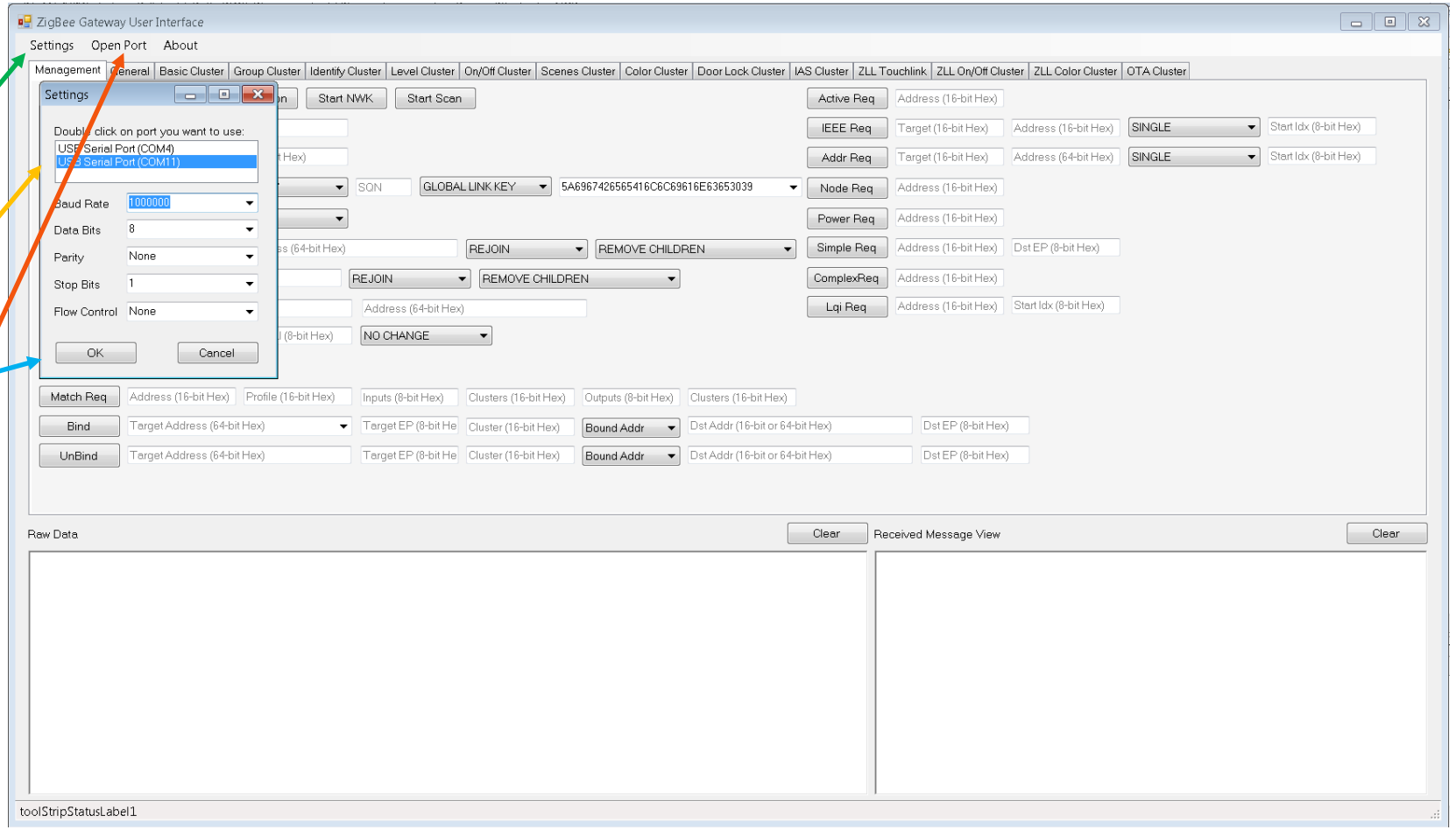
C:\NXP\bstudio_nxp\workspace\JN-AN-1216-Zigbee-3-0-IoT-ControlBridge-BRC5\Tools\TestTool



ZigBee Gateway User Interface (ZGWUI) Tool

Open Port:

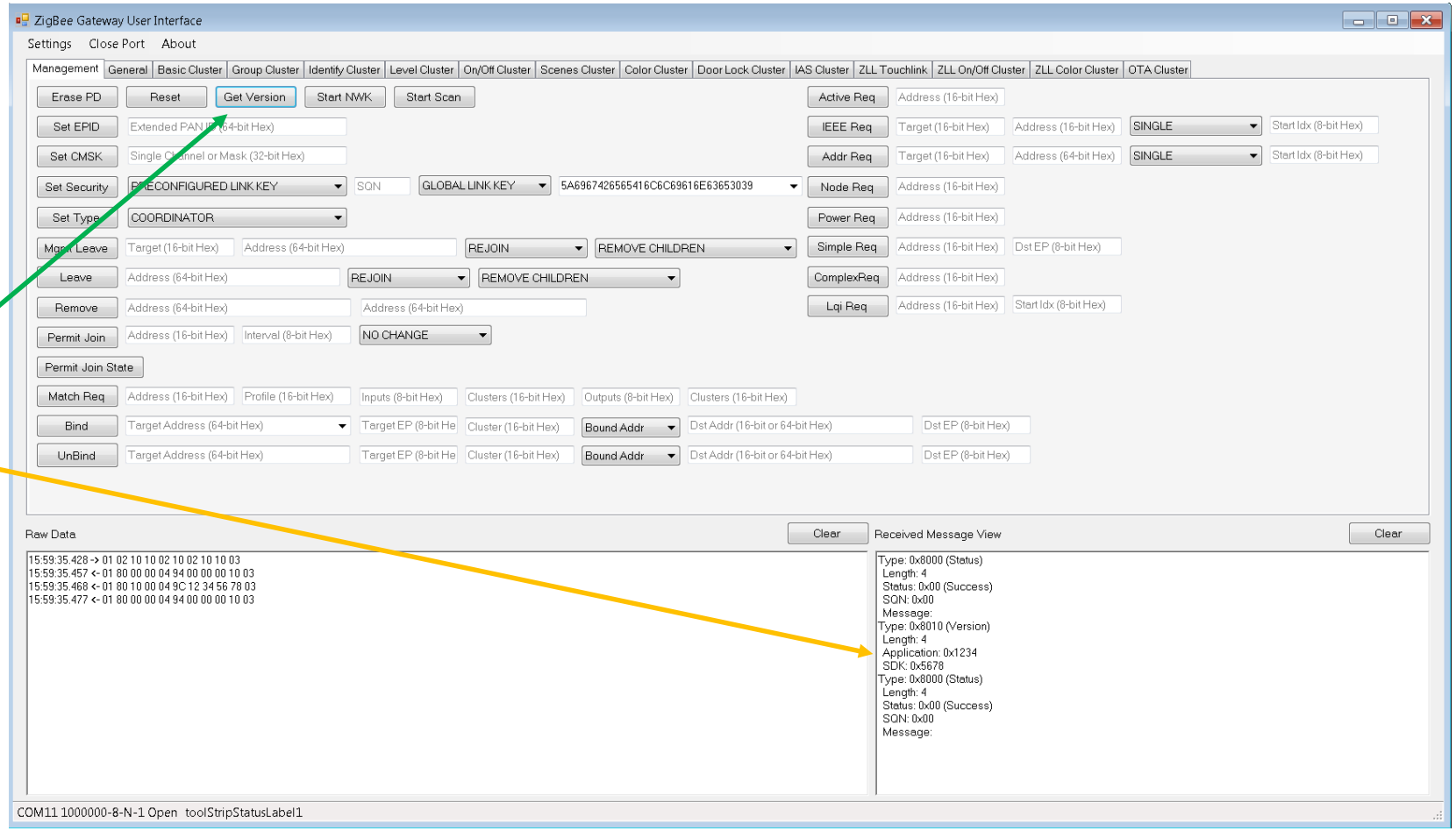
1. Select “Settings”
2. Select COM Port
3. Press “OK”
4. Press “Open Port”



ZigBee Gateway User Interface (ZGWUI) Tool

Read USB Dongle Version:

1. Press “Get Version”
2. Check return



Channel Allocation

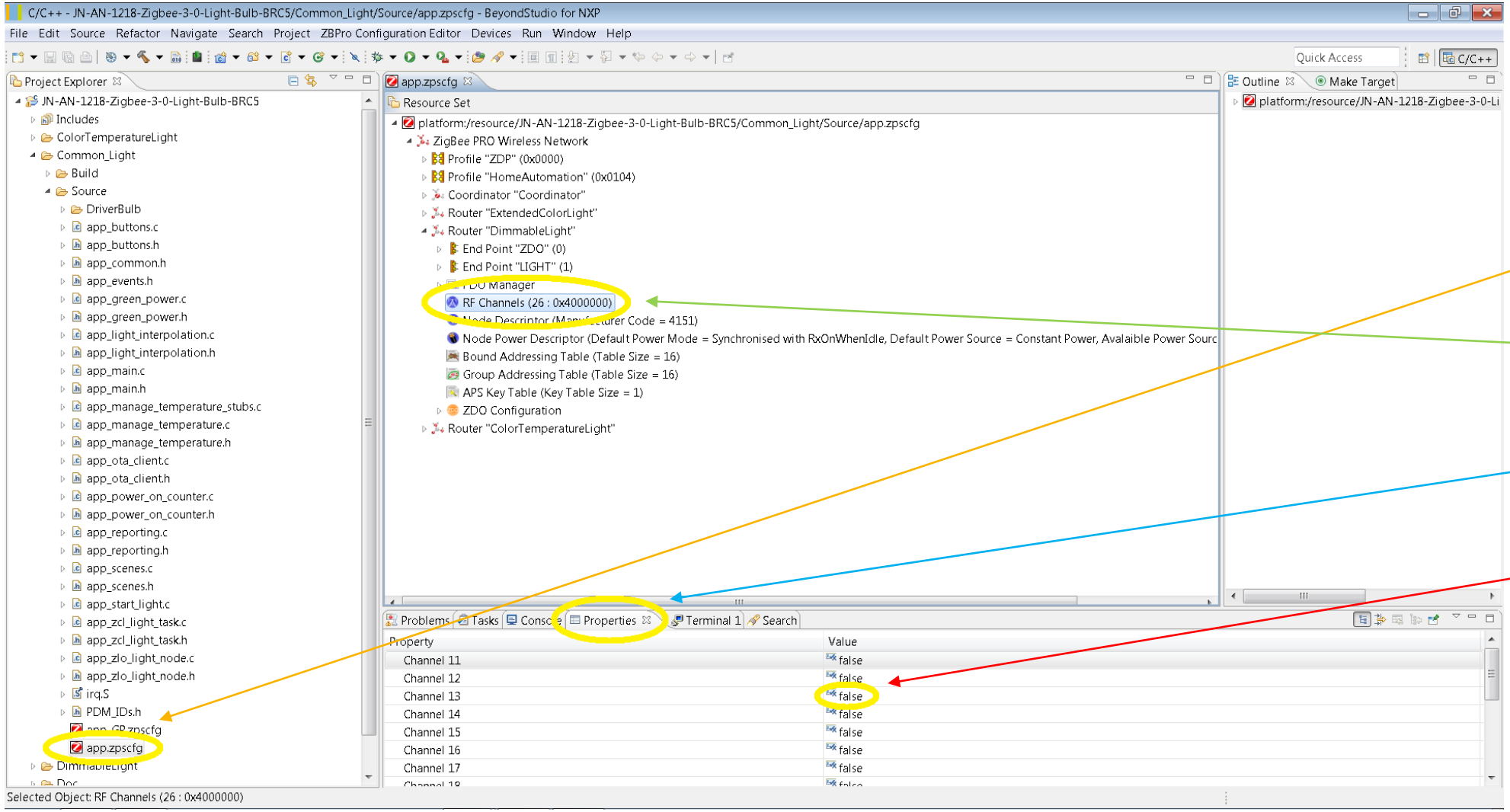
Each group has a radio channel assigned to them, please use your channel for the following hands-on modules.

HANDS ON MODULE 1: JOINING

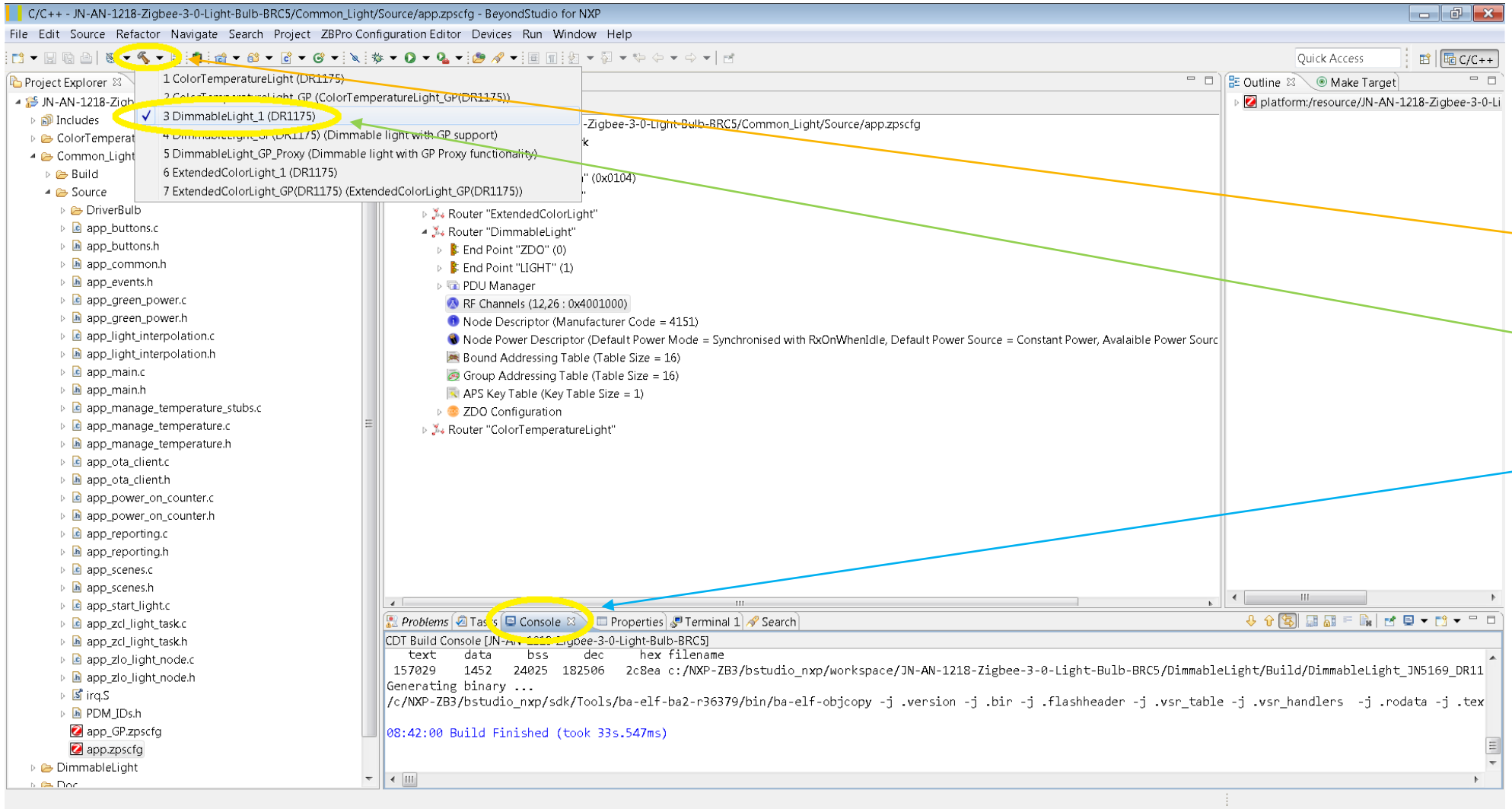
Overview

- **Module 1: Joining (30 mins)**
  - Modify the Dimmable Light Bulb example (router) to use assigned channel
  - Build the Dimmable Light Bulb example with serial debug enabled
  - Download binary file into target hardware
  - Start network coordinator using ZGWUI tool
  - Join Light Bulb to network
  - Examine debug output using serial console
  - Observe on air packets with sniffer and walk through the joining process

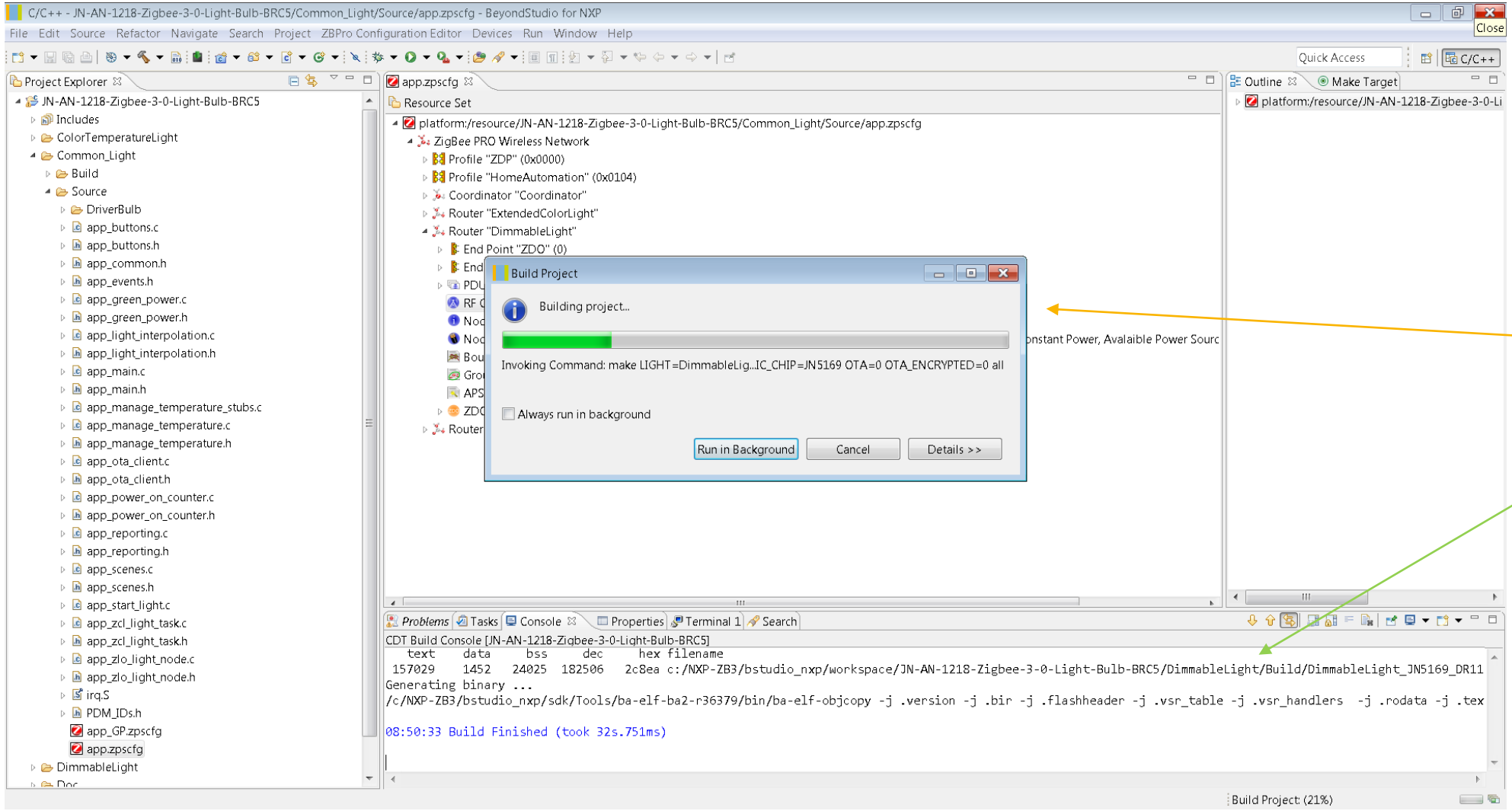
Select the Light Bulb Operating Channel



Rebuild the Light Bulb Binary File [1]



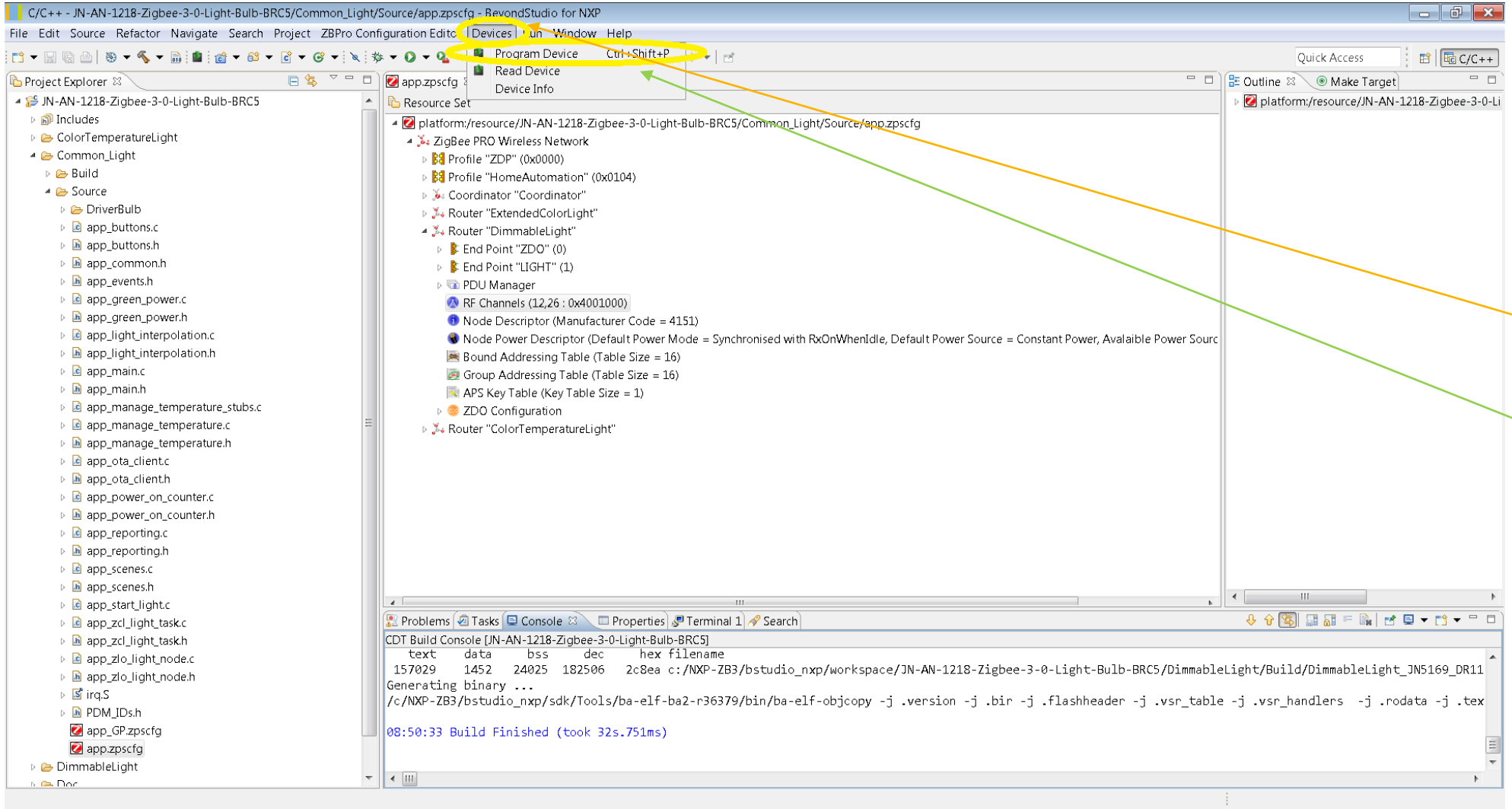
Rebuild the Light Bulb Binary File [2]



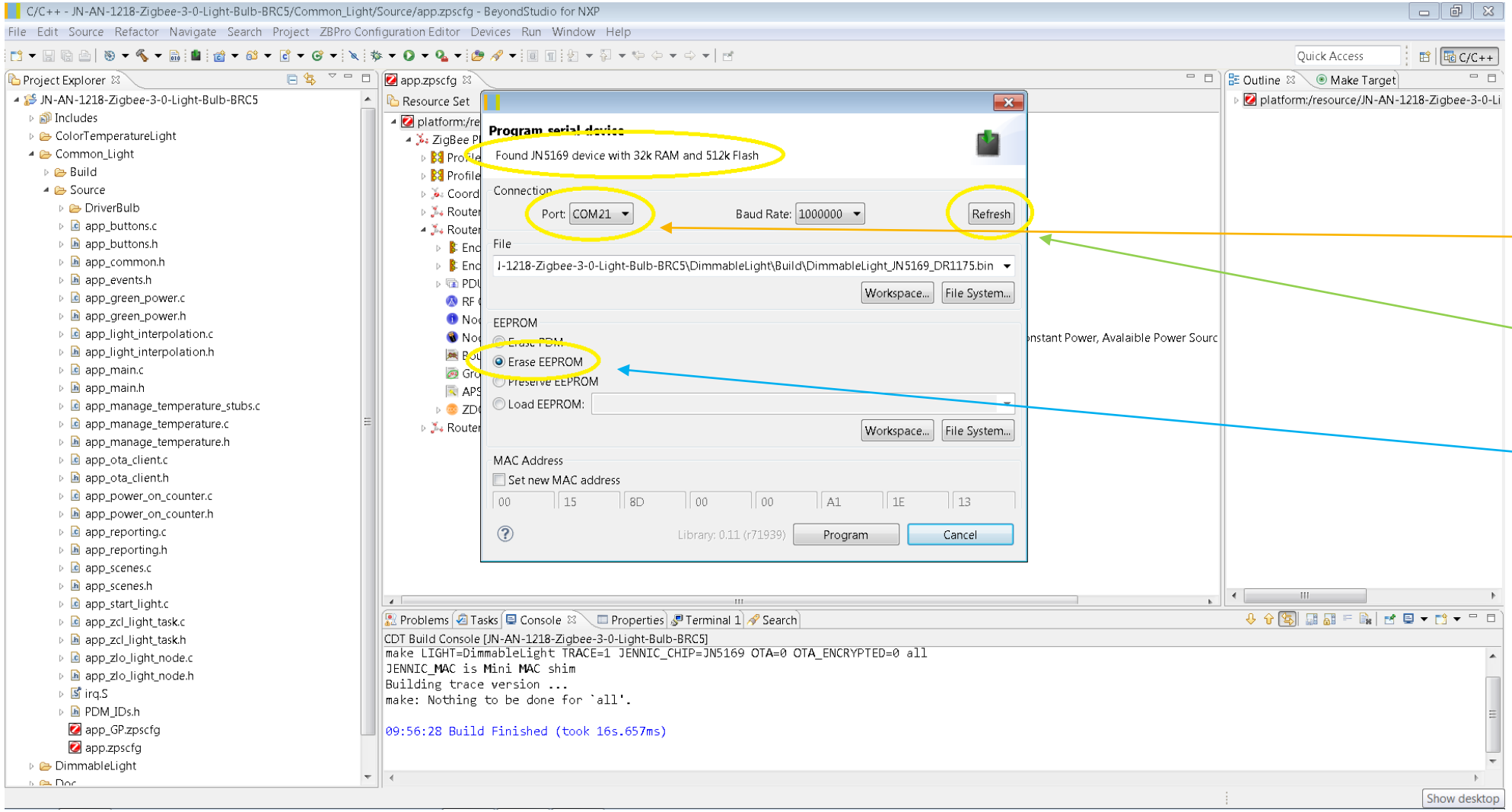
Step 8

Step 9

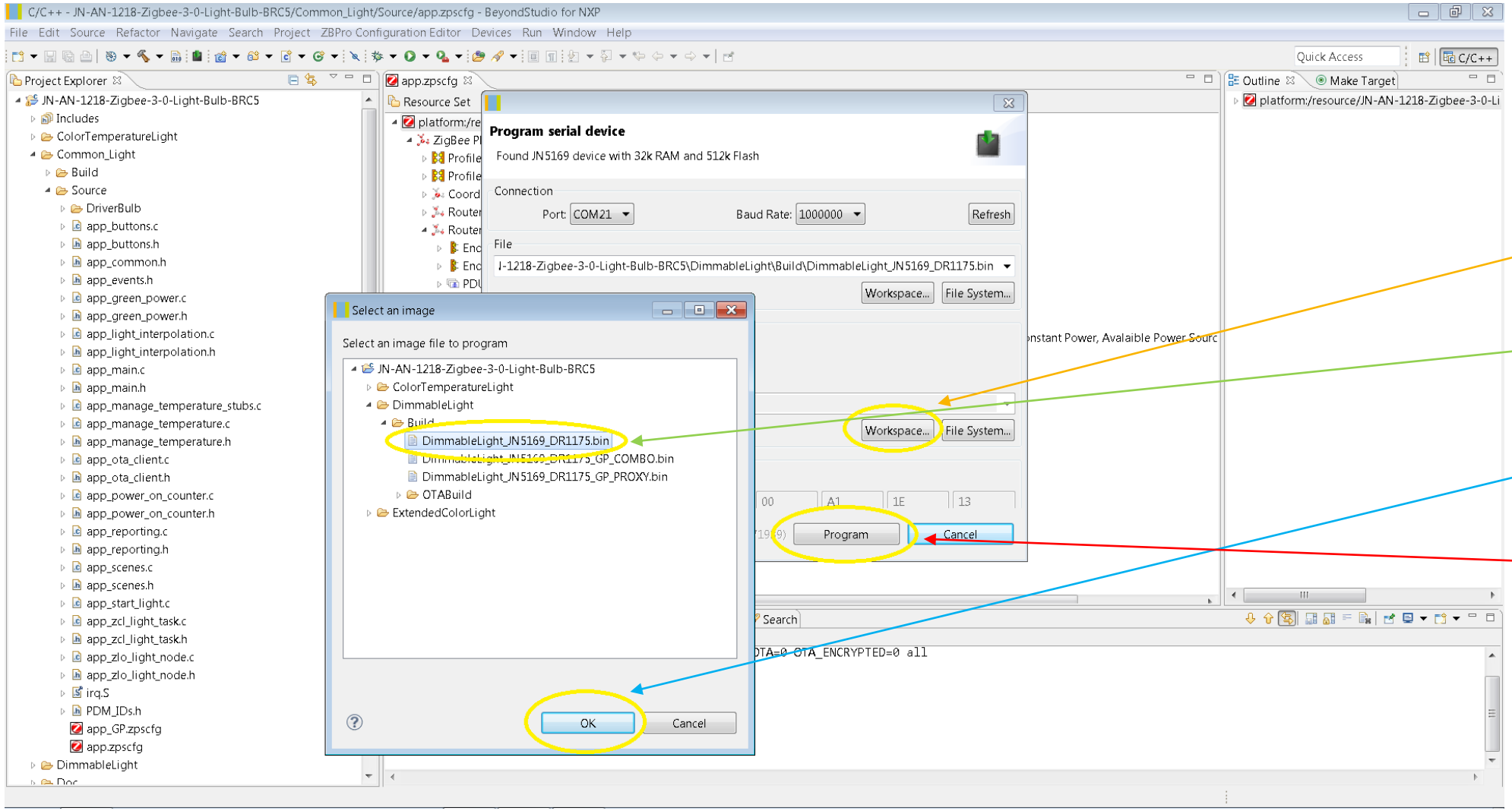
Program the Light Bulb Binary into the Target Hardware [1]



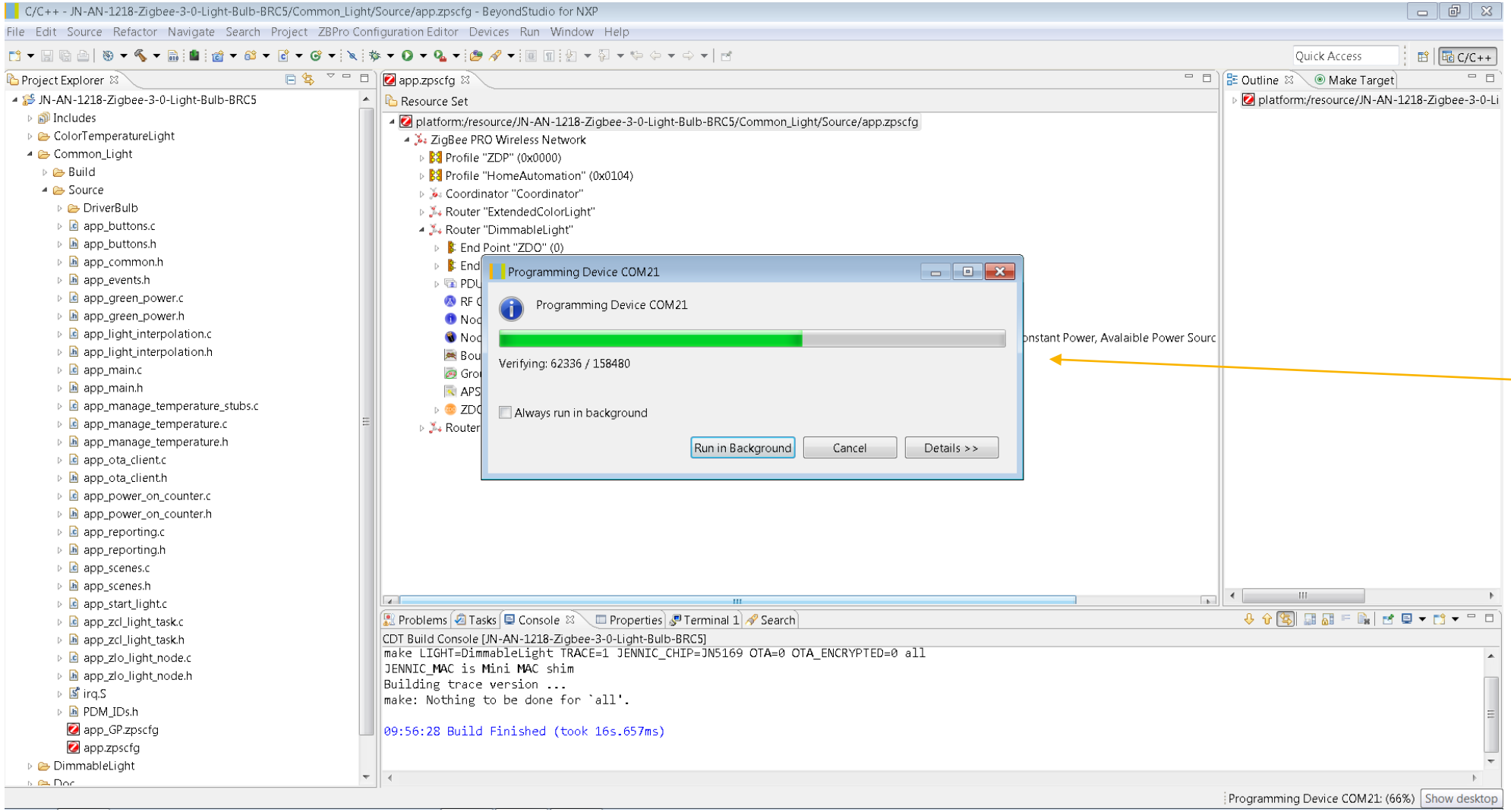
Program the Light Bulb Binary into the Target Hardware [2]



Program the Light Bulb Binary into the Target Hardware [3]

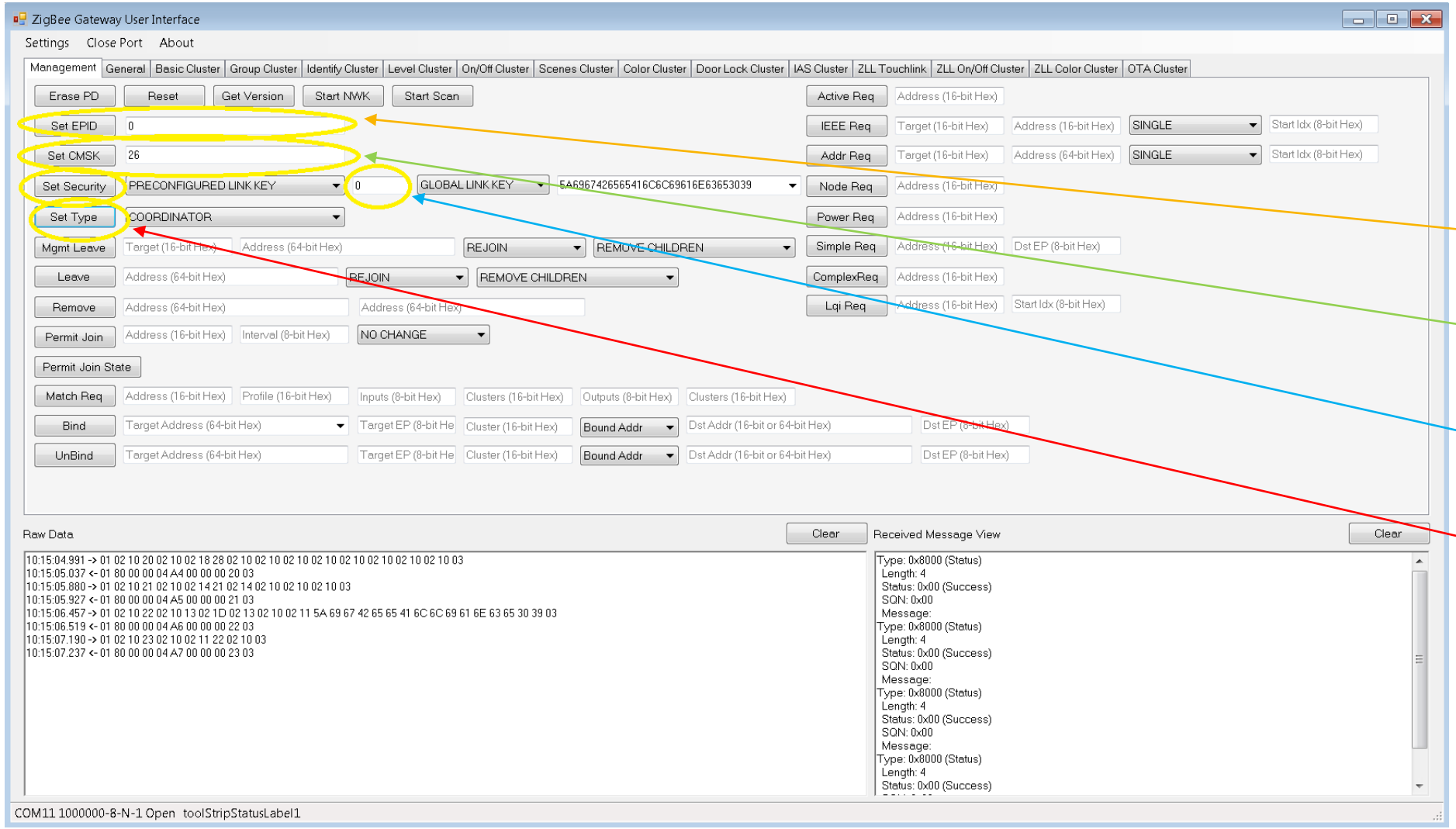


Program the Light Bulb Binary into the Target Hardware [4]



Step 10

Start Network Coordinator using ZGWUI Tool [1]



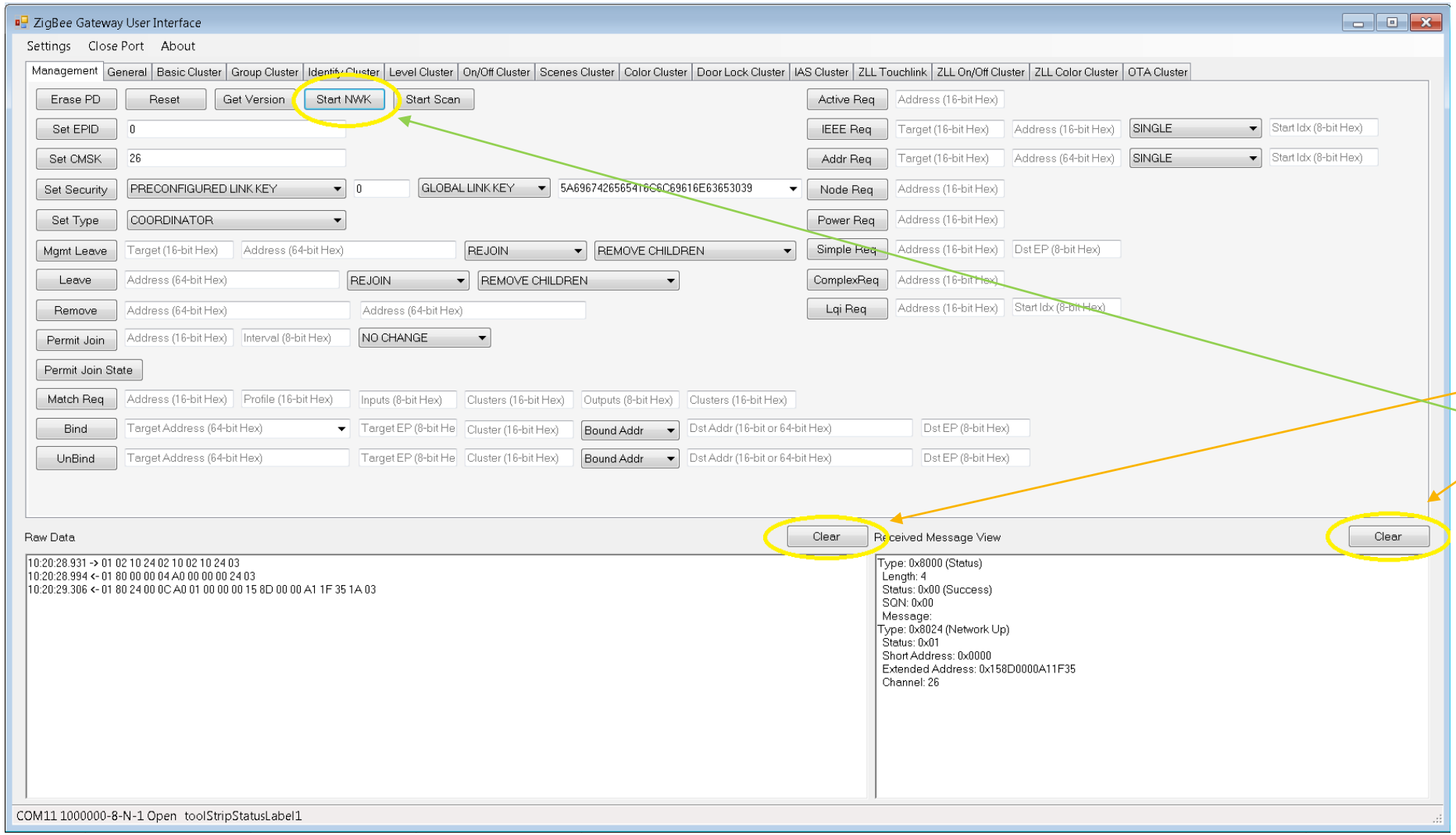
Step 1

Step 2

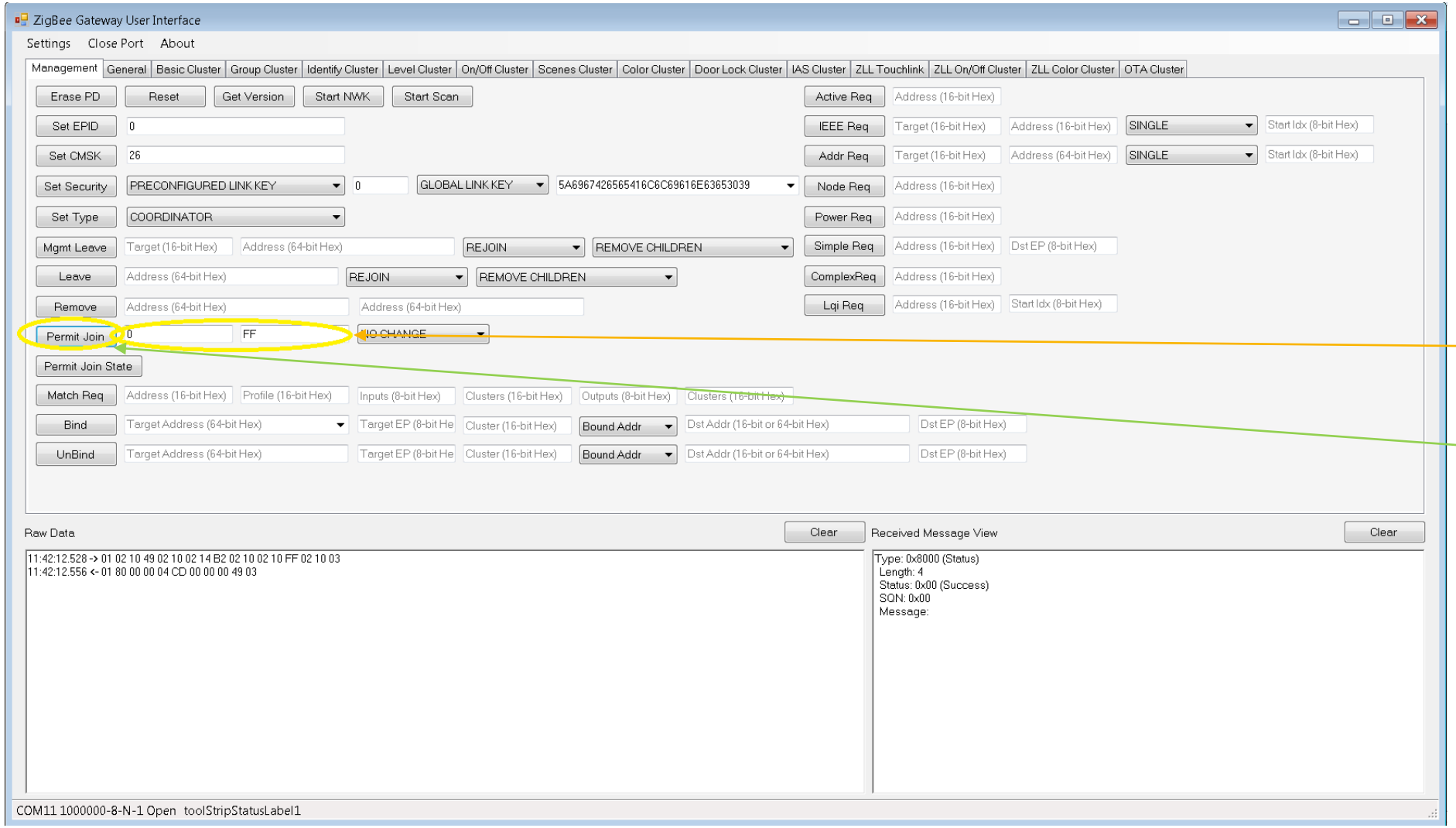
Step 3

Step 4

Start Network Coordinator using ZGWUI Tool [2]



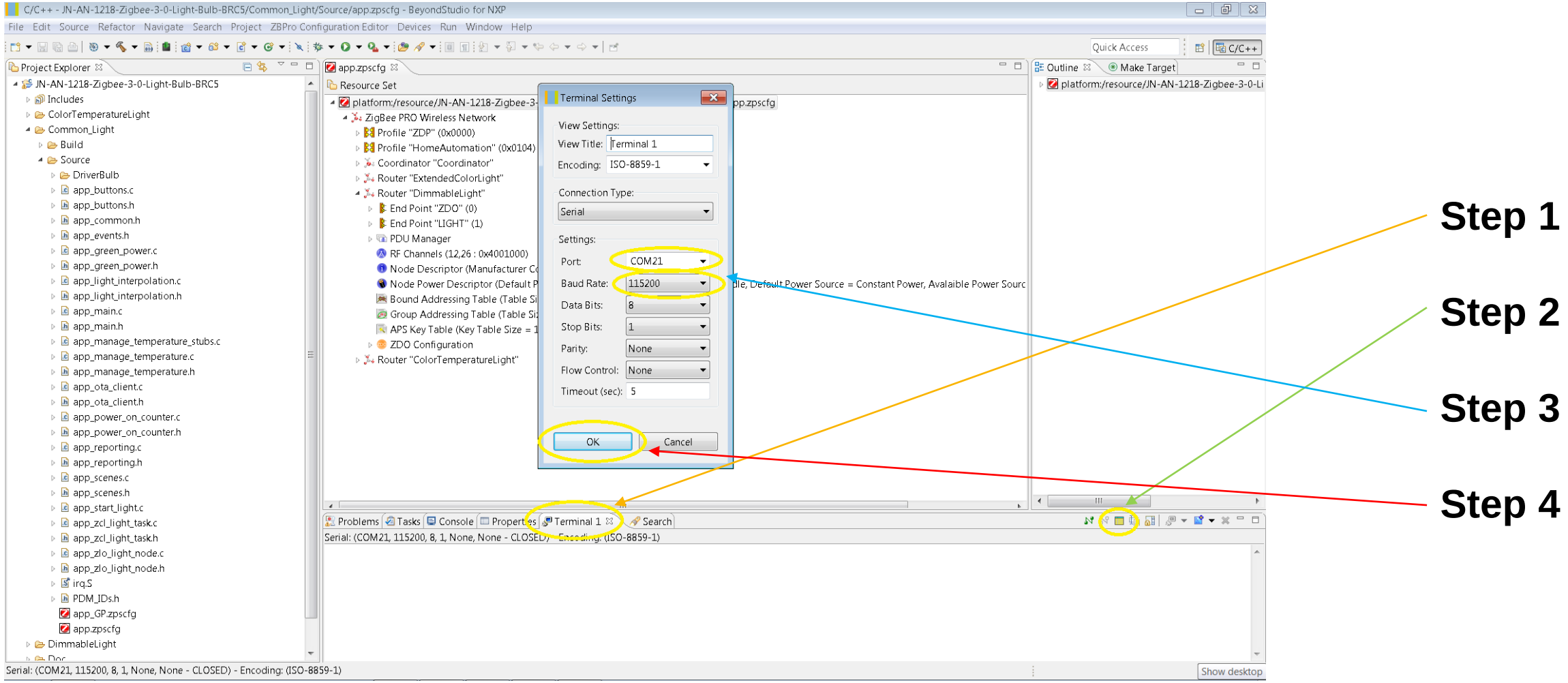
Open Network for Joining using ZGWUI Tool



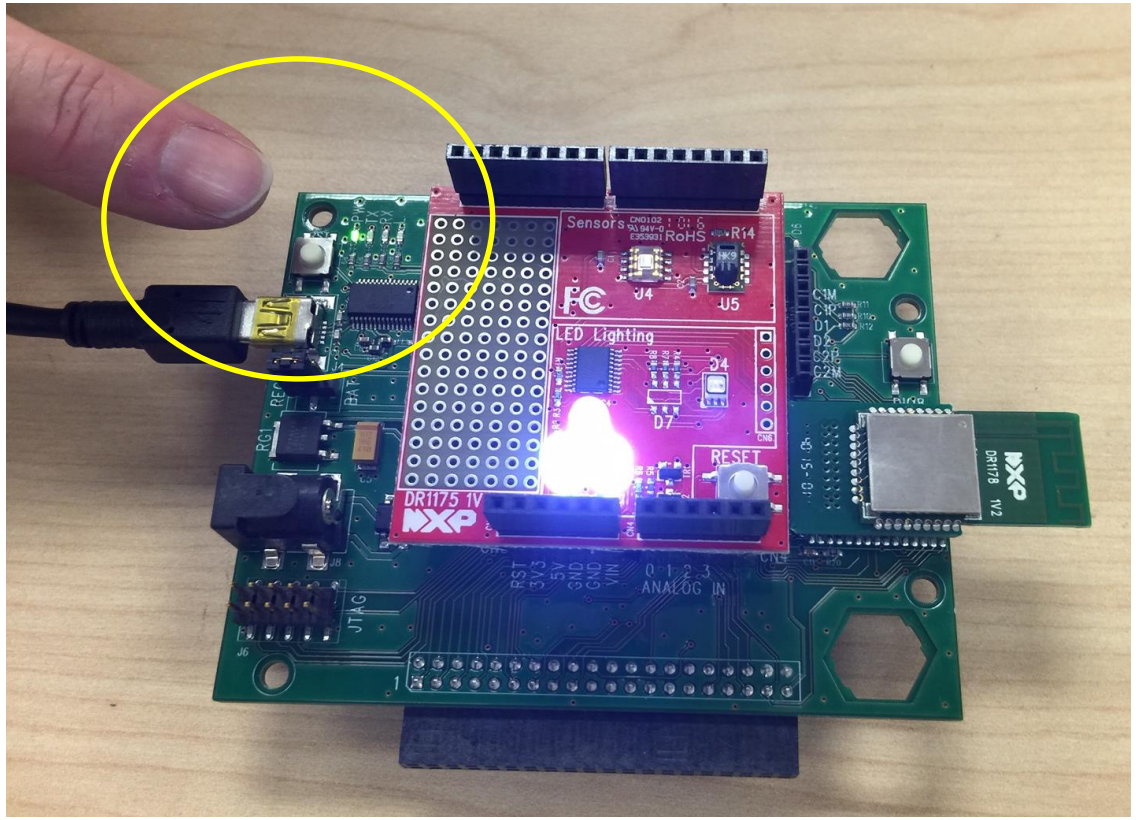
Step 1

Step 2

Monitoring Light Bulb Serial Debug Output

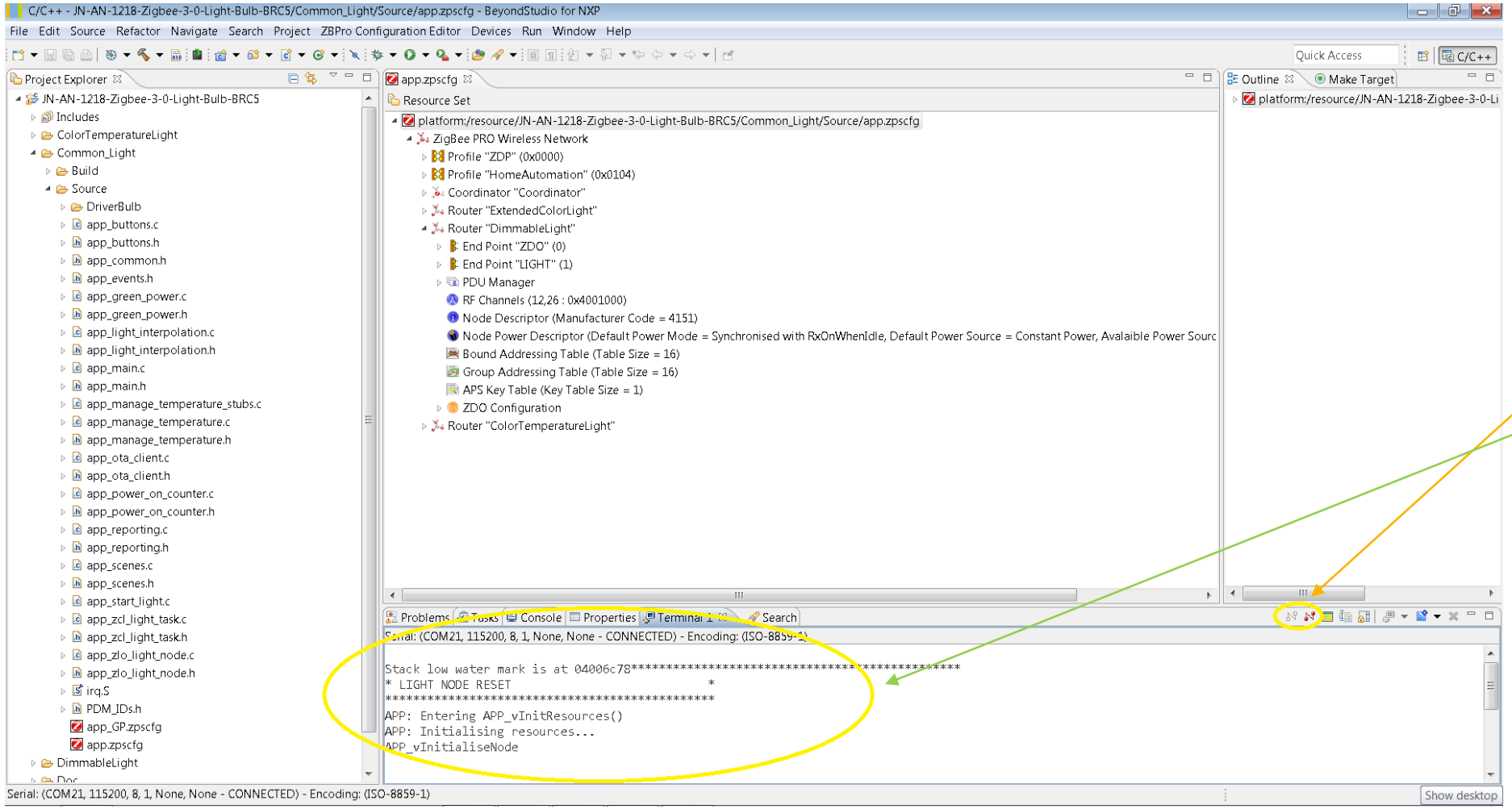


Join Light Bulb to Network Coordinator

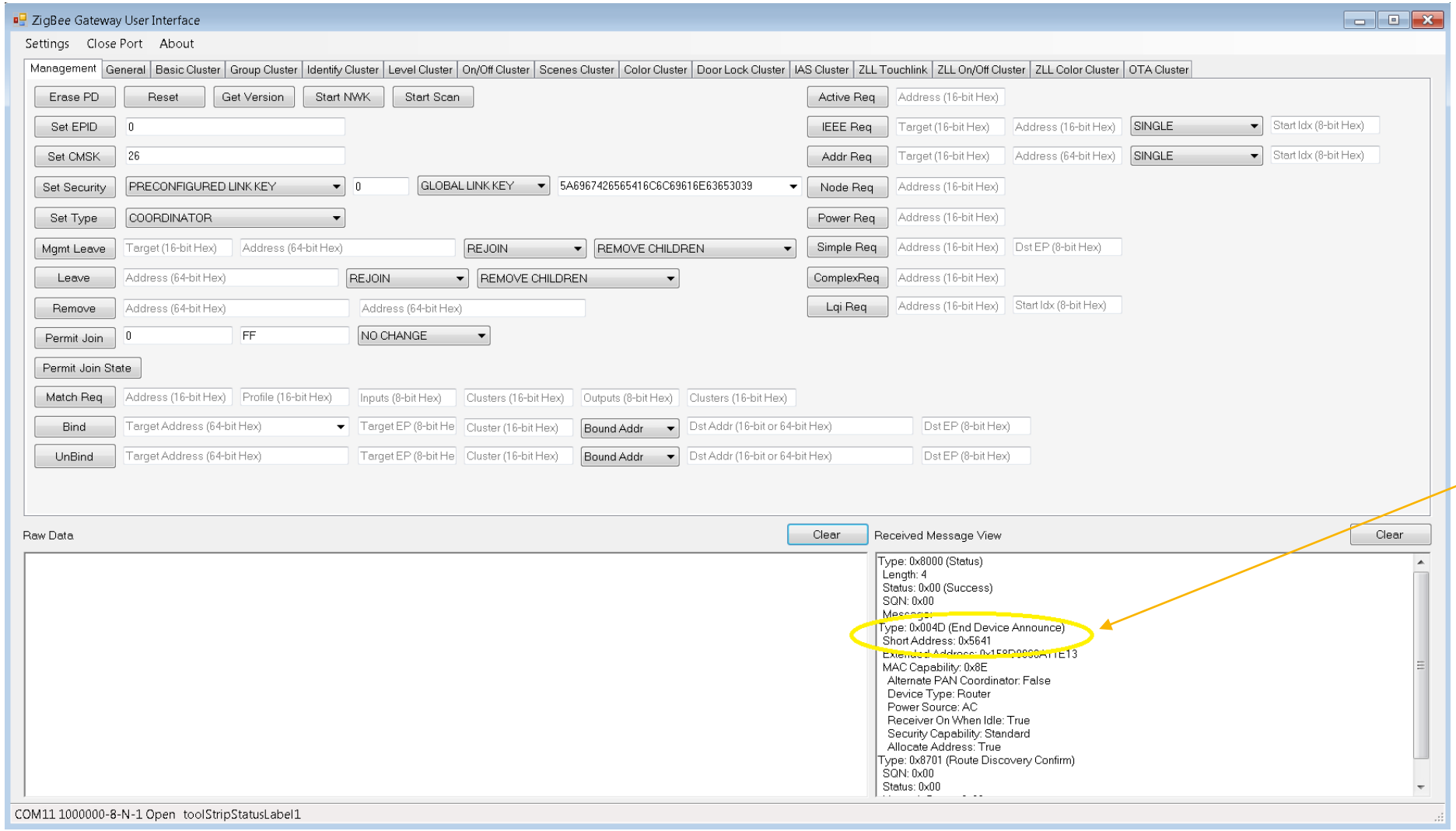


Press Reset button (RST) to initiate join sequence

Monitor Light Bulb Serial Debug Output

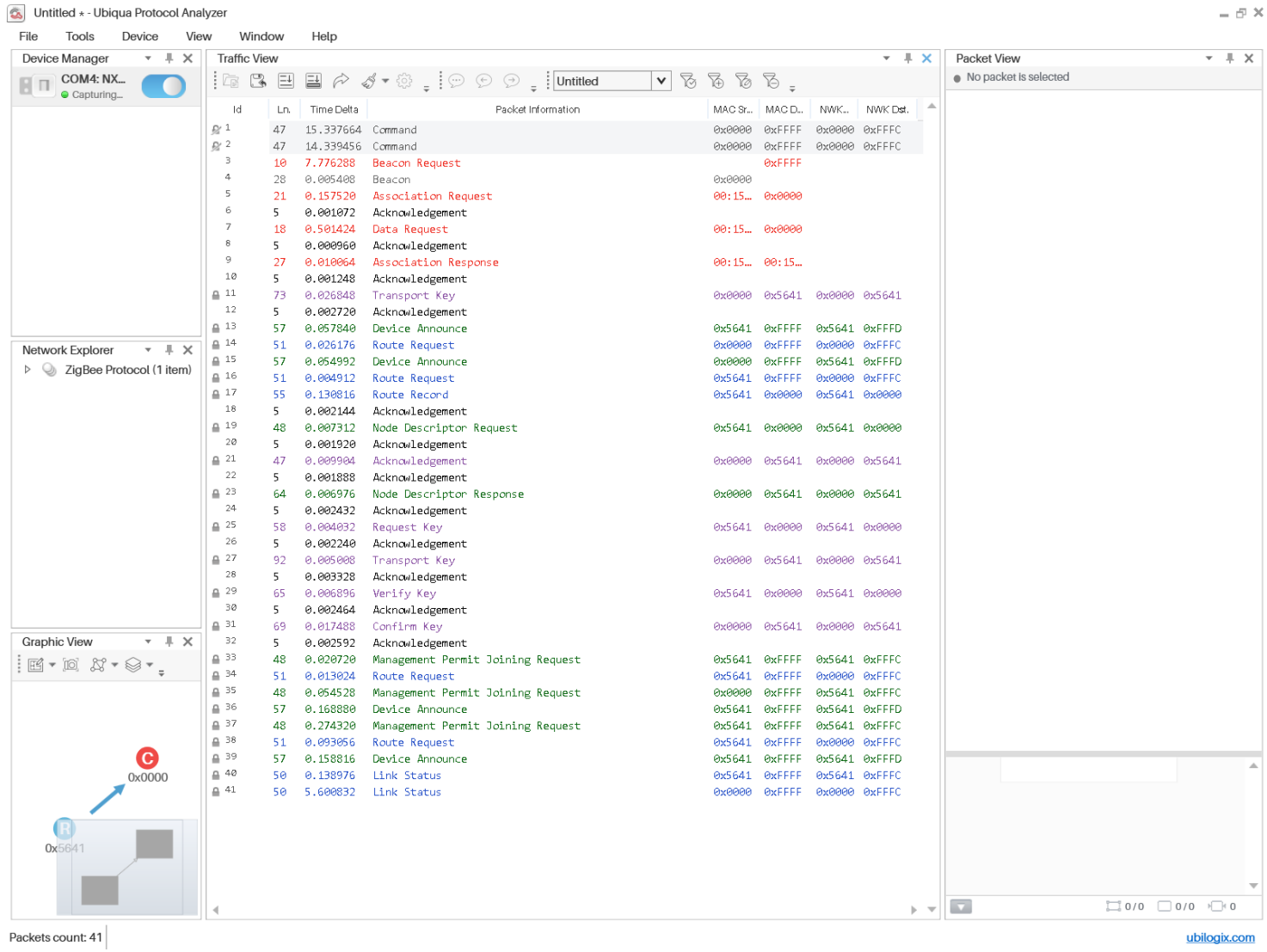


Determine Address of Joining Device using ZGWUI



Step 1

Packets Exchanged during Join Process

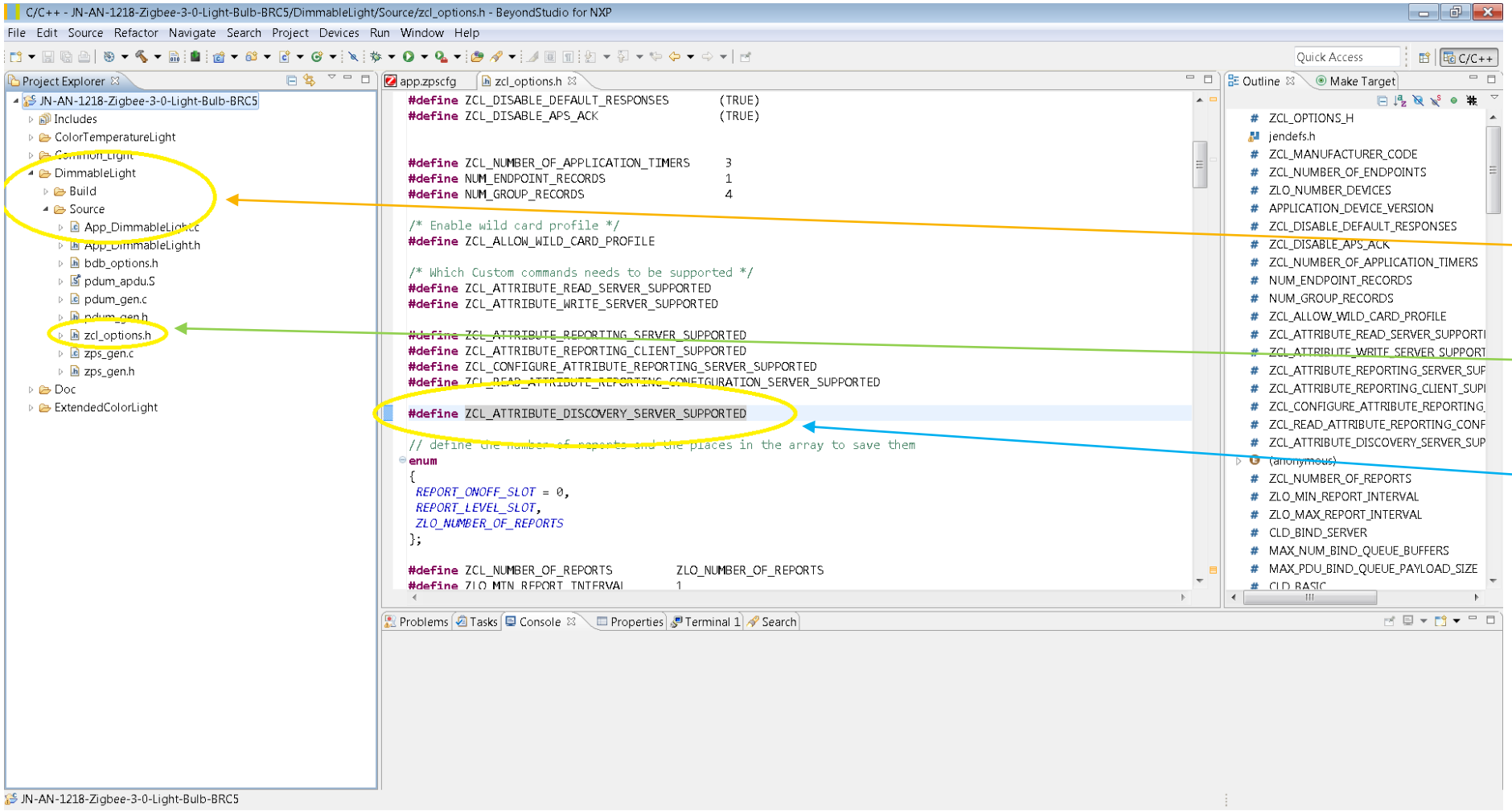


HANDS ON MODULE 2:
DEVICE AND SERVICE DISCOVERY

Overview

- Module 2: Device and Service Discovery (30 mins)
  - Enable Attribute discovery, rebuild binary and download (preserving context)
  - Discovery device using MGMT LQI request
  - Discover active end points
  - Discover clusters (simple descriptor)
  - Discover attributes (On/Off Cluster)
  - Toggle On/Off attribute in On/Off Cluster

Enable Attribute Discovery [1]



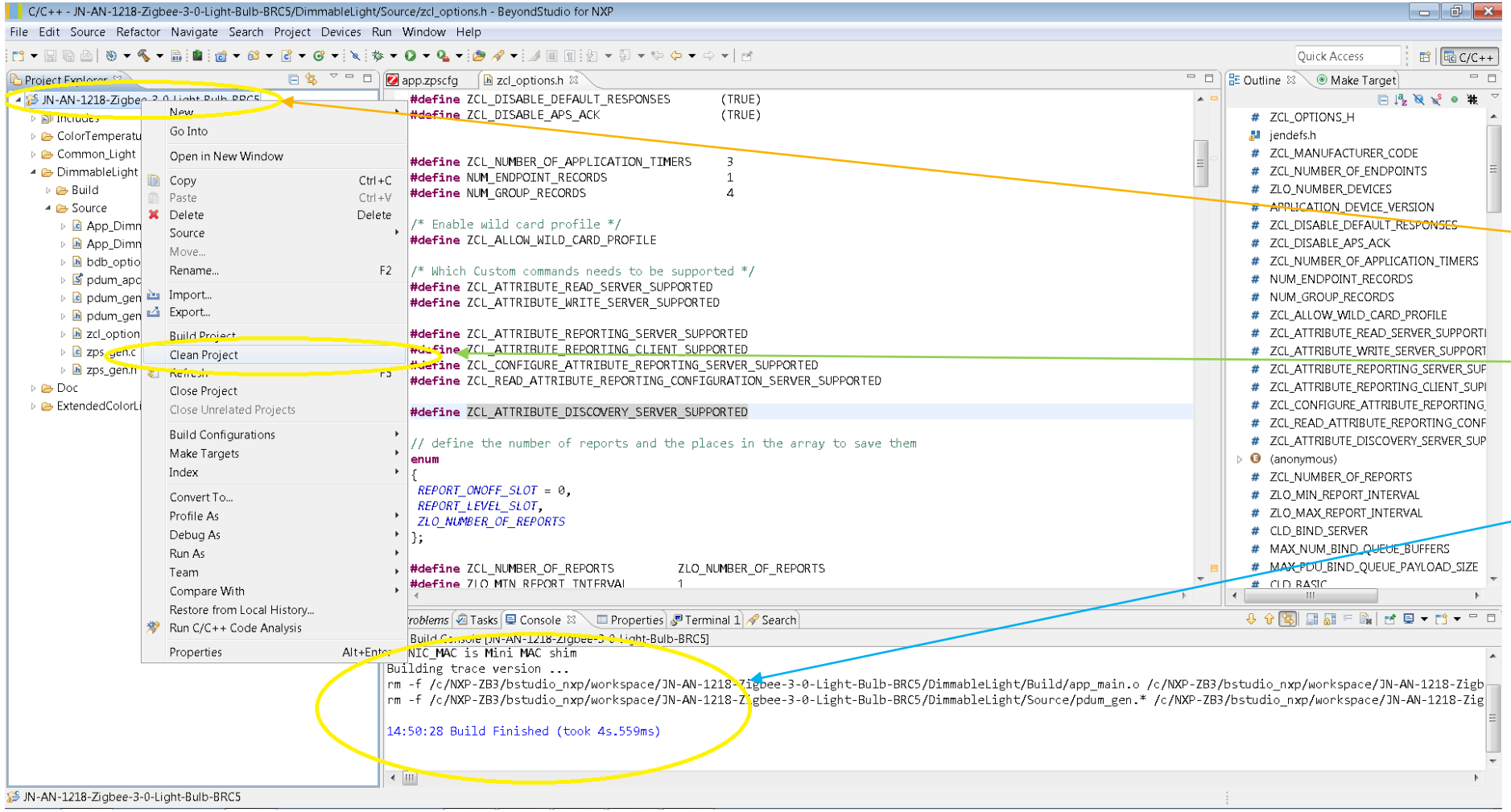
Step 1

Step 2

Step 3

```
#define ZCL_ATTRIBUTE_DISCOVERY_SERVER_SUPPORTED
```

Enable Attribute Discovery [2]

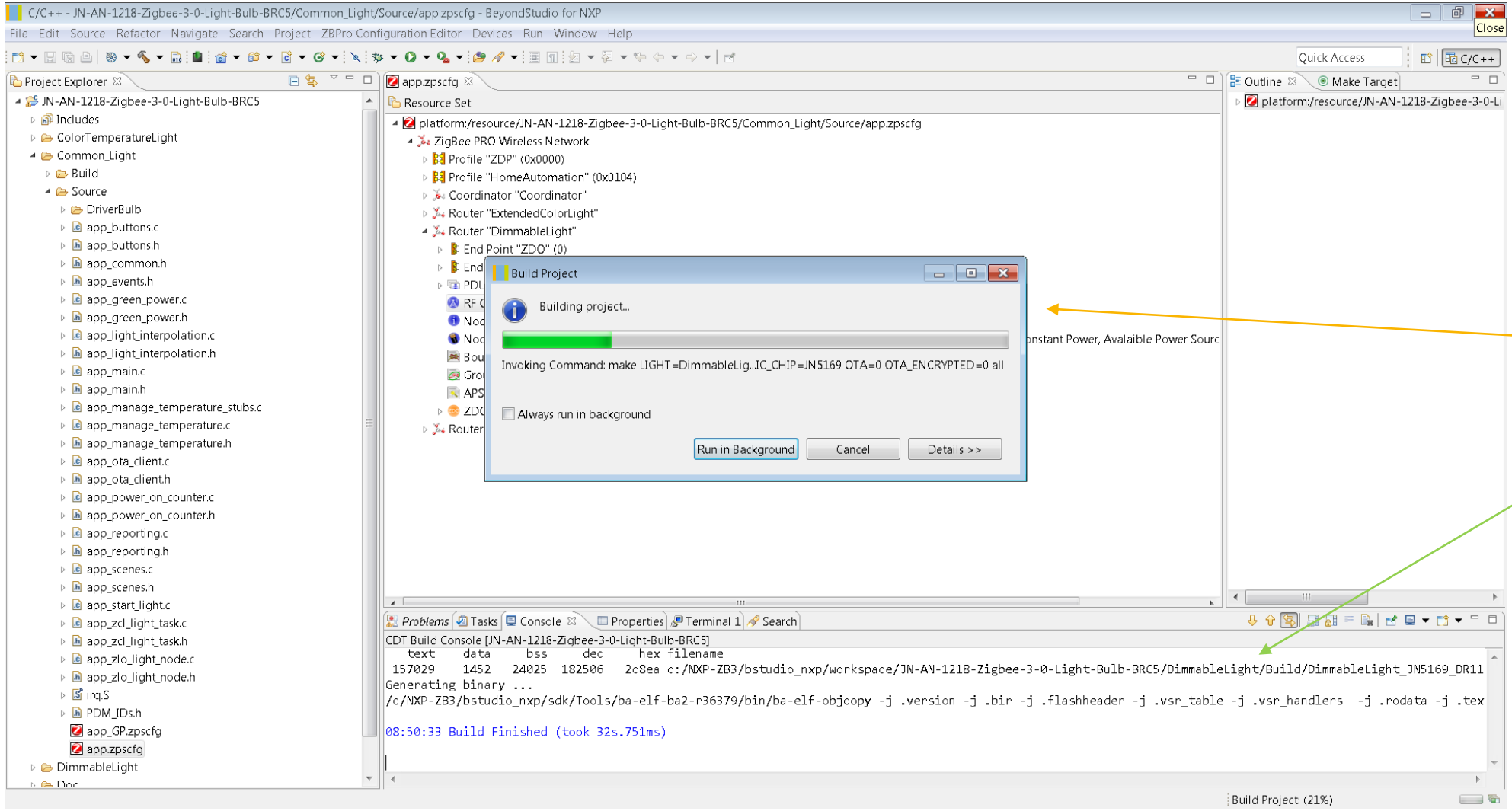


Step 4

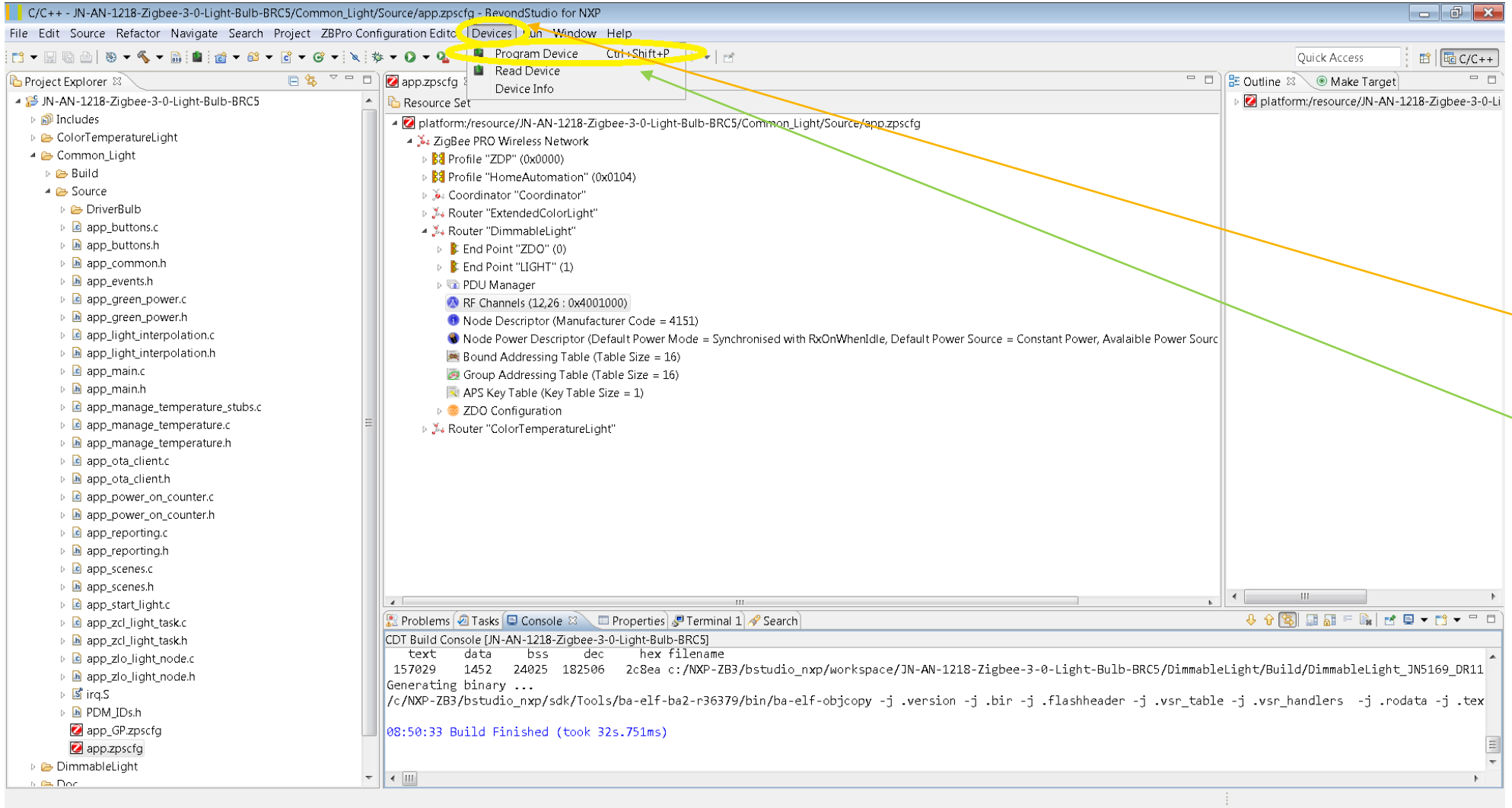
Step 5

Step 6

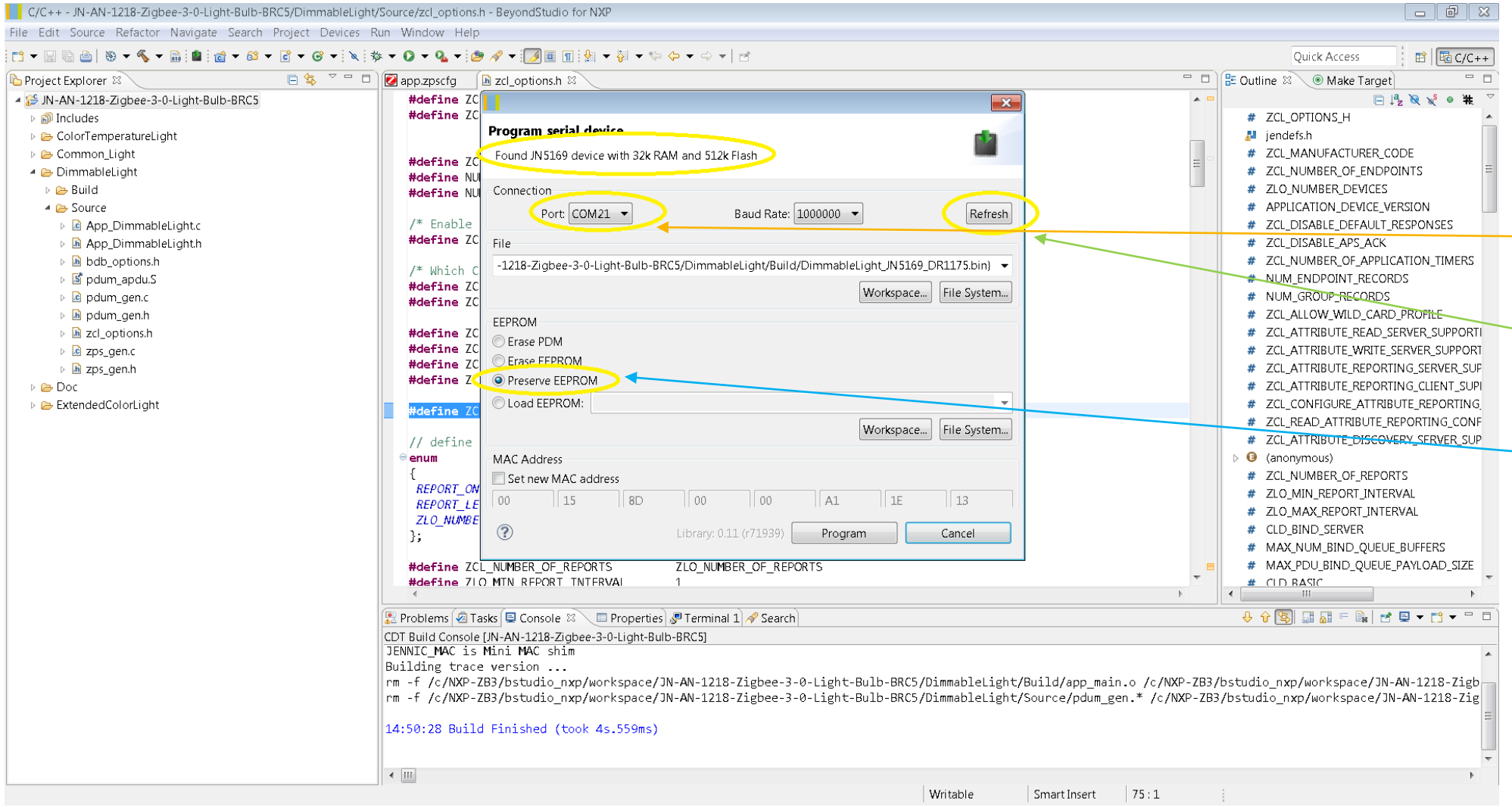
Enable Attribute Discovery [2]



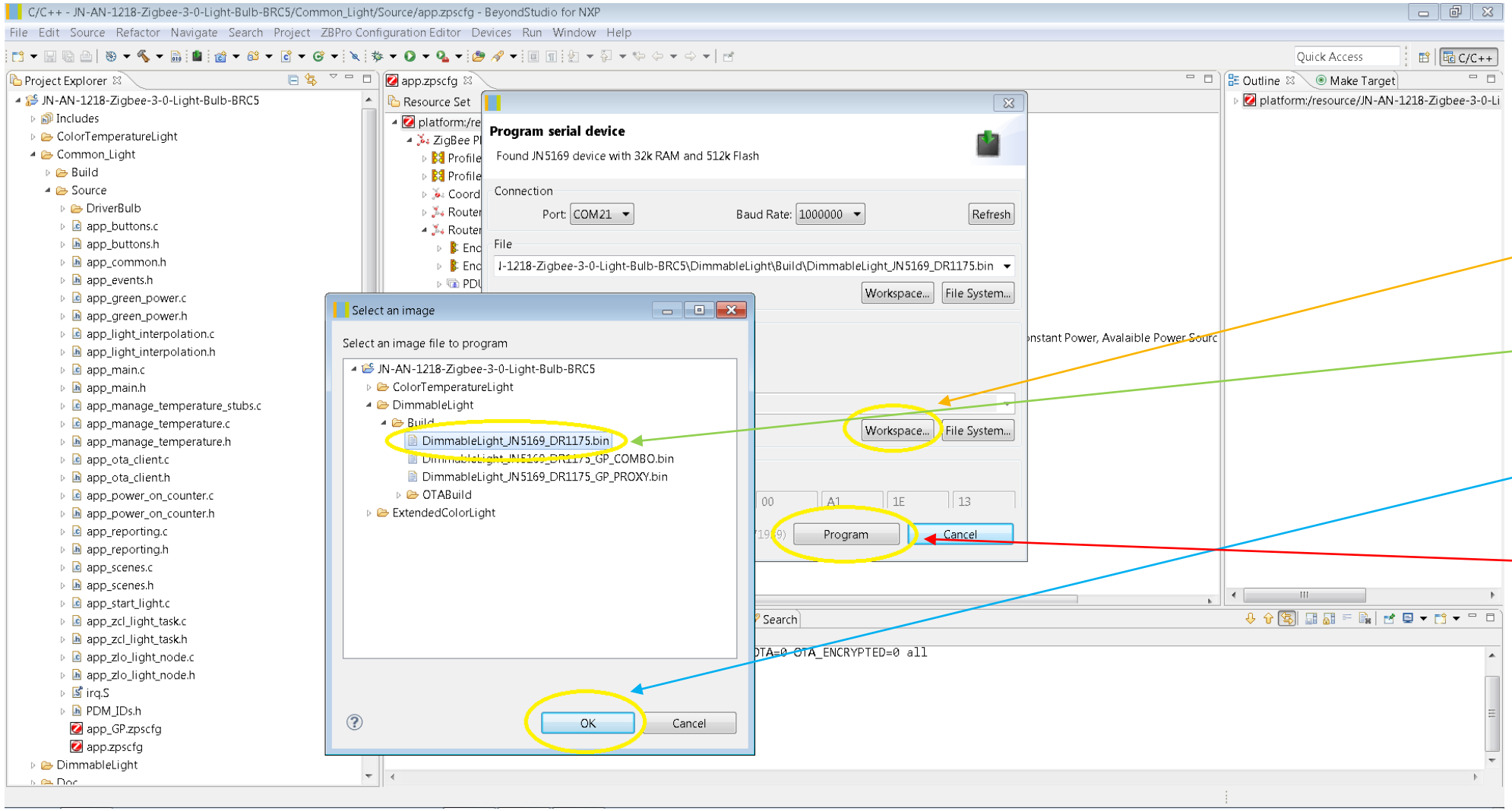
Program the Light Bulb Binary into the Target Hardware [1]



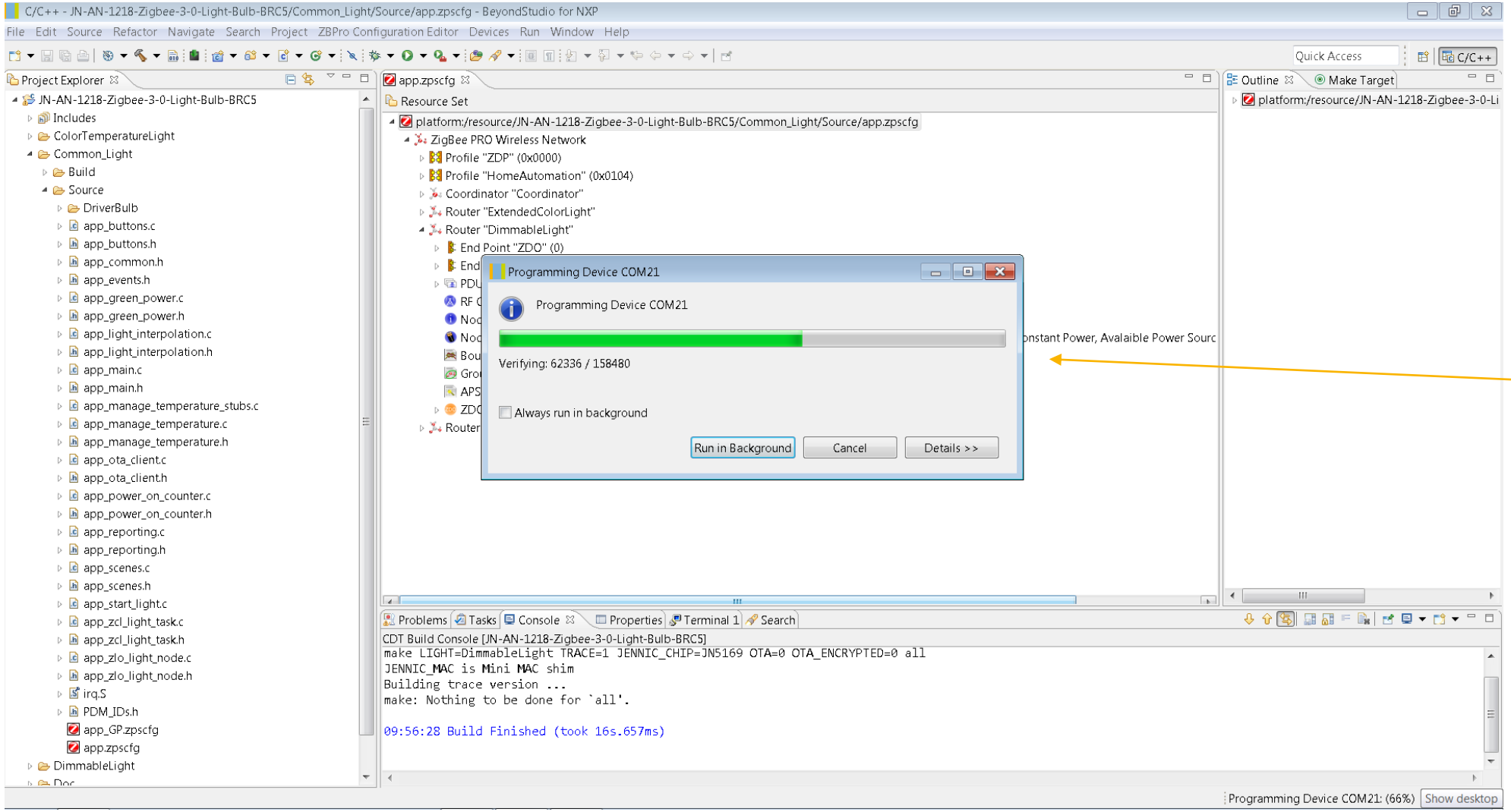
Program the Light Bulb Binary into the Target Hardware [2]



Program the Light Bulb Binary into the Target Hardware [3]

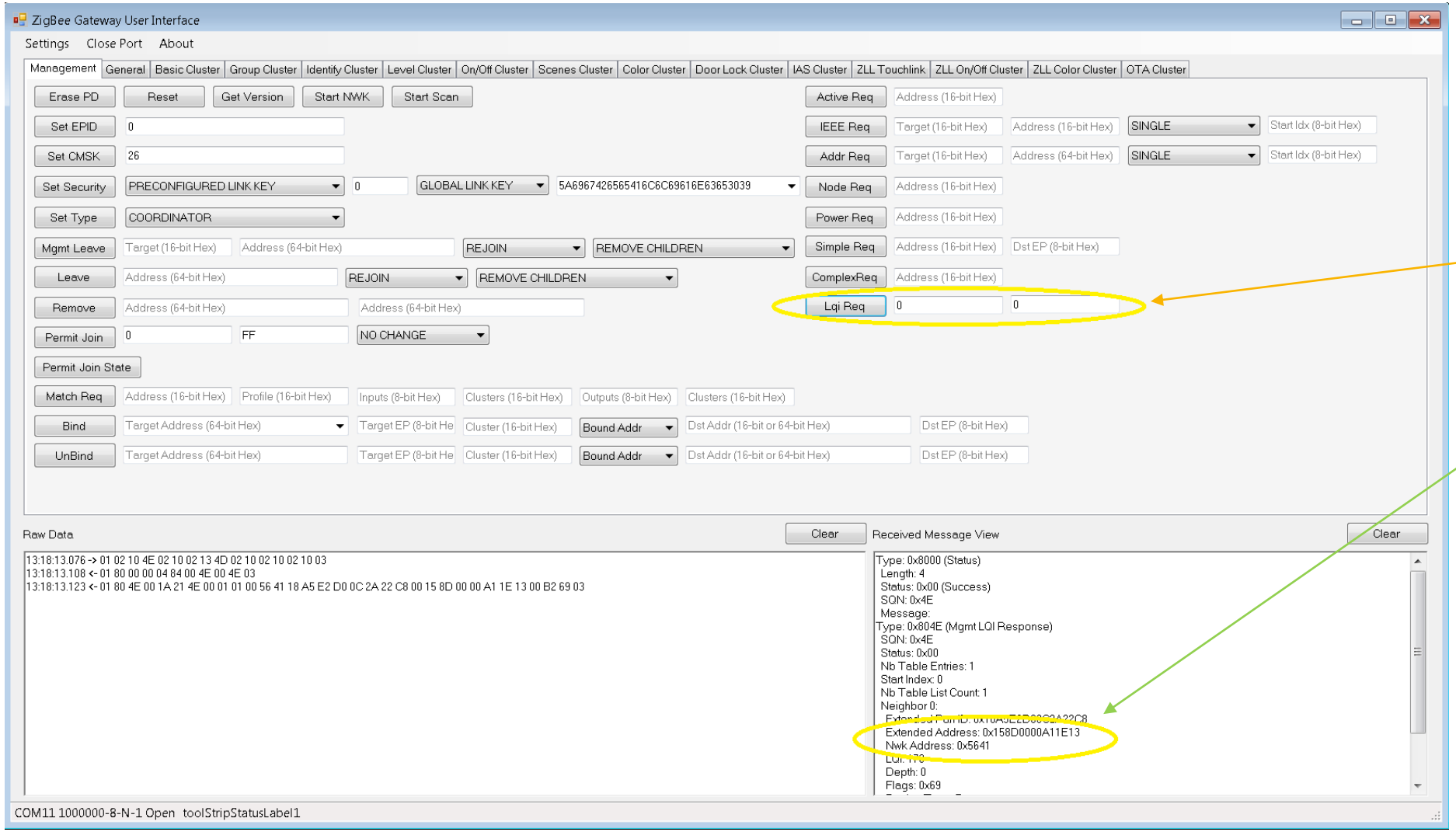


Program the Light Bulb Binary into the Target Hardware [4]

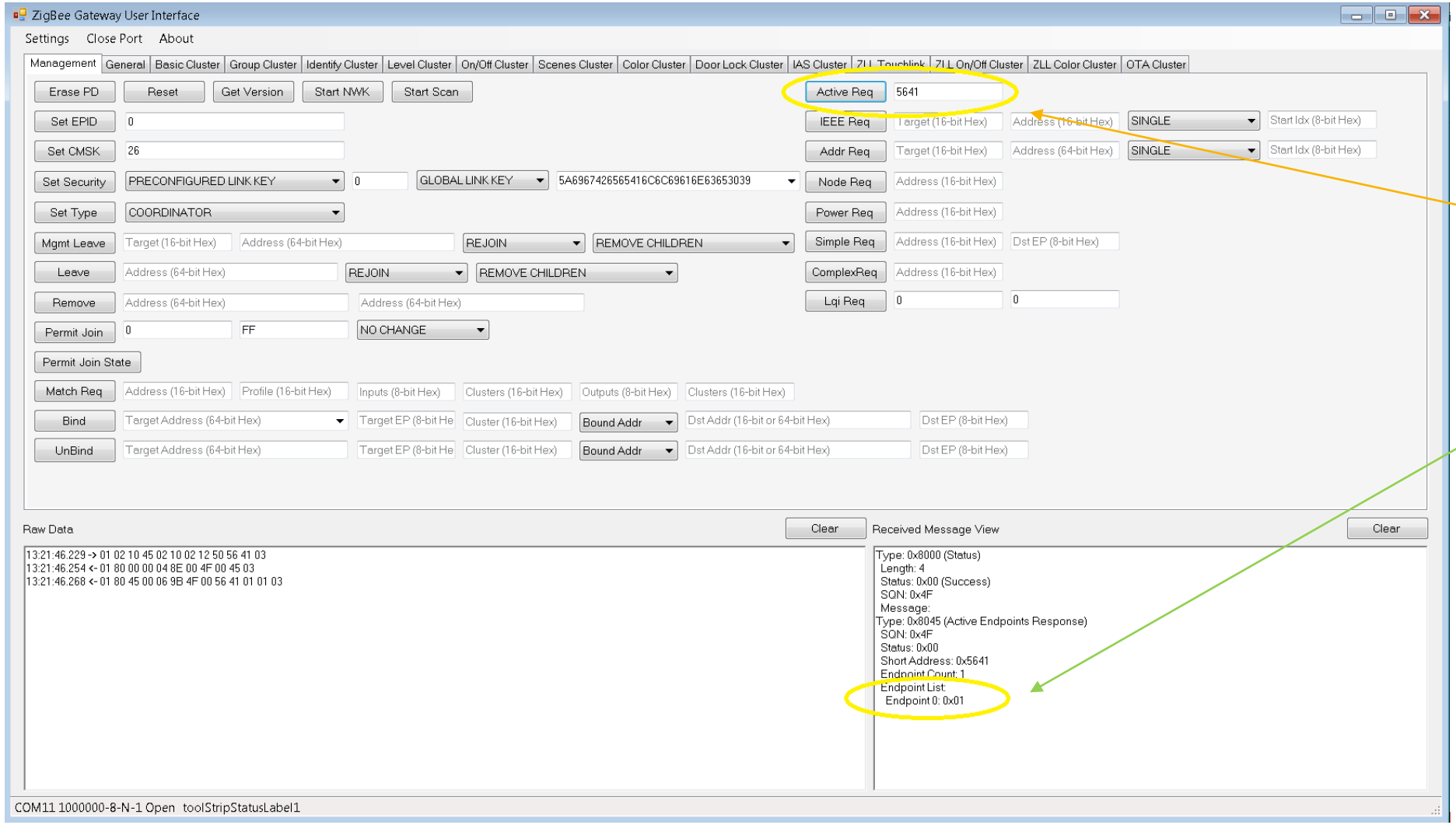


Step 10

Device Discovery



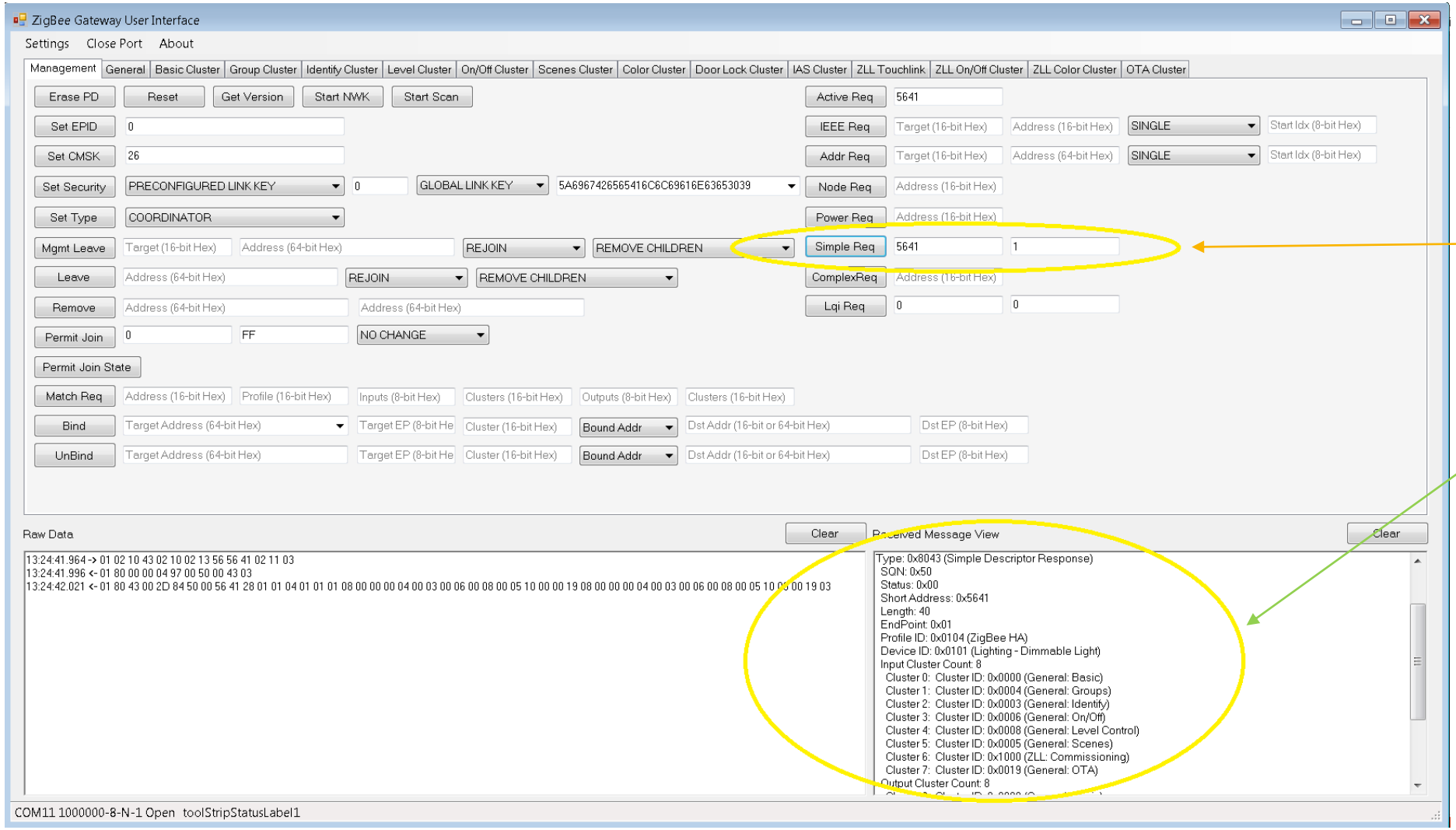
End Point Discovery



Step 1

Step 2

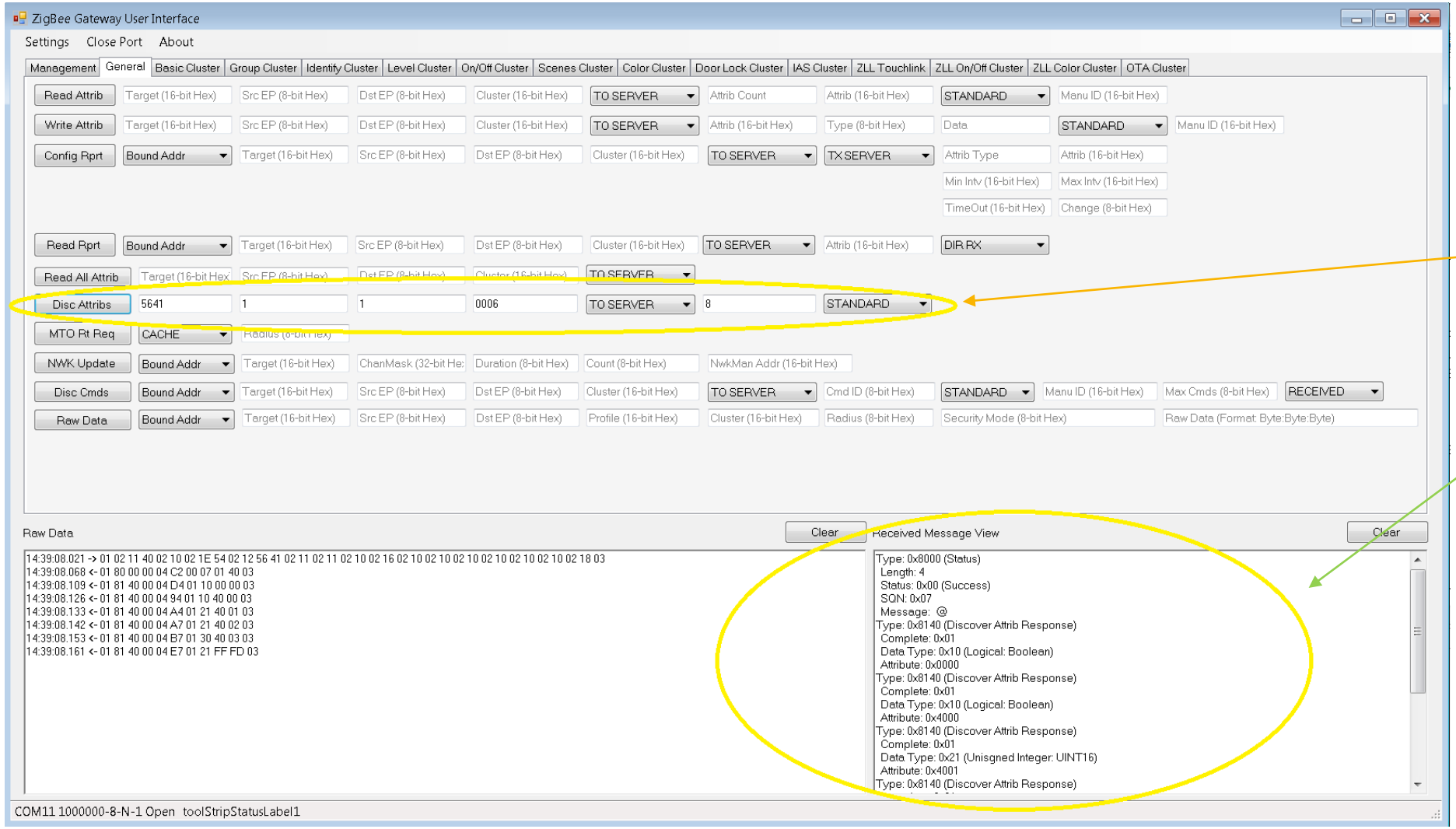
Simple Descriptor Discovery



Step 1

Step 2

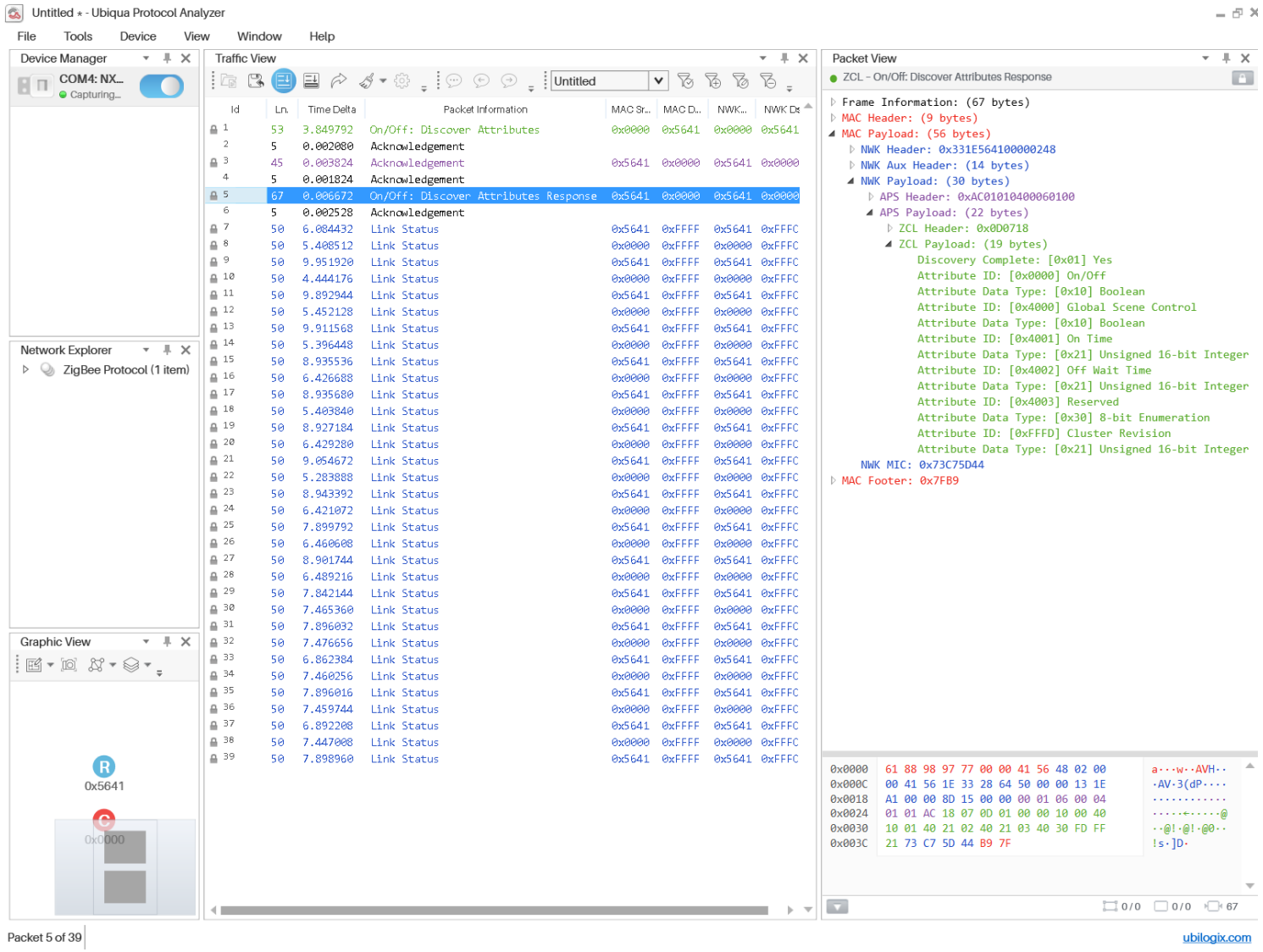
Attribute Discovery



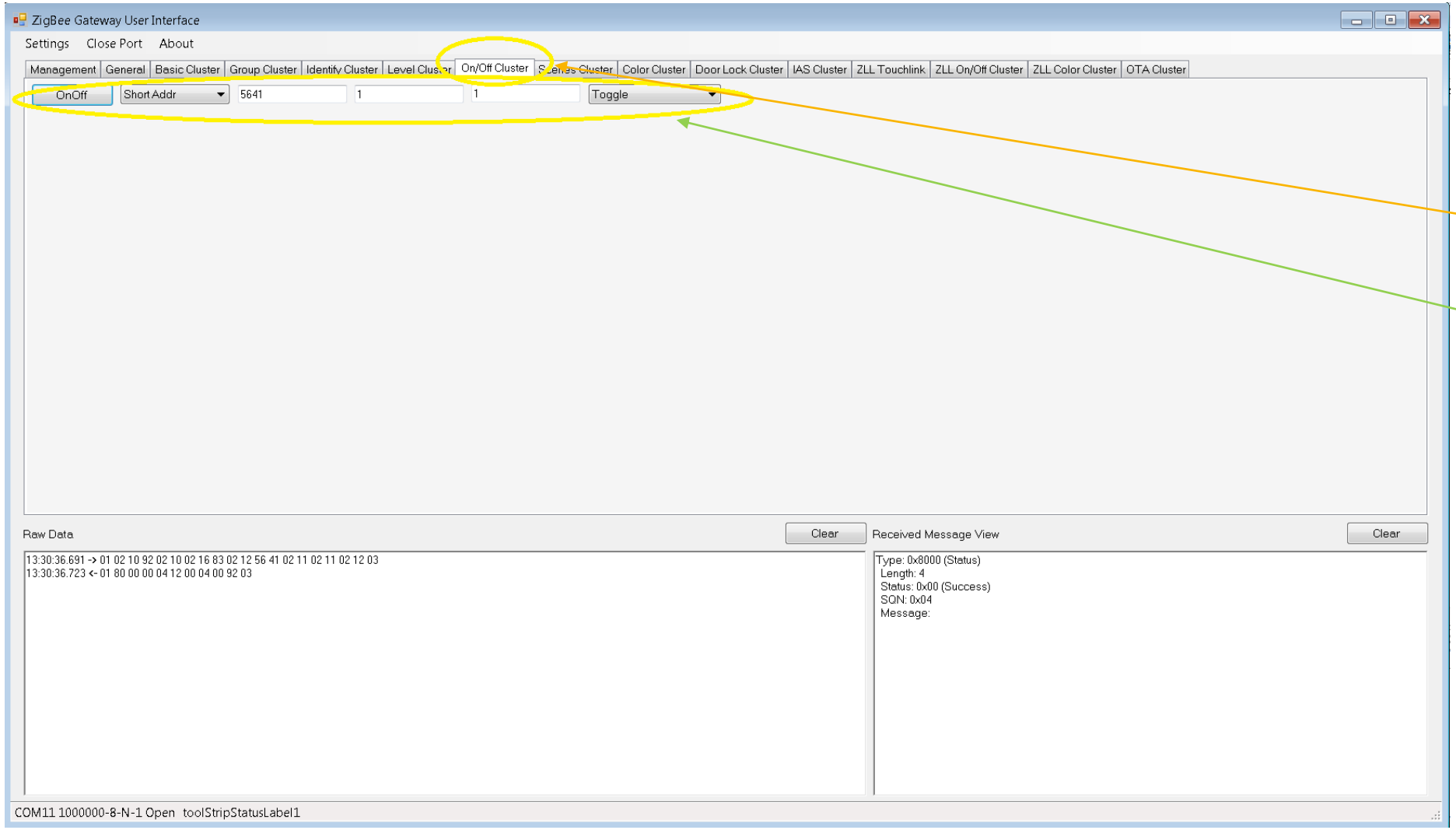
Step 1

Step 2

Attribute Discovery Packet Exchange



Control Light Bulb via On/Off Attribute of On/Off Cluster



Step 1

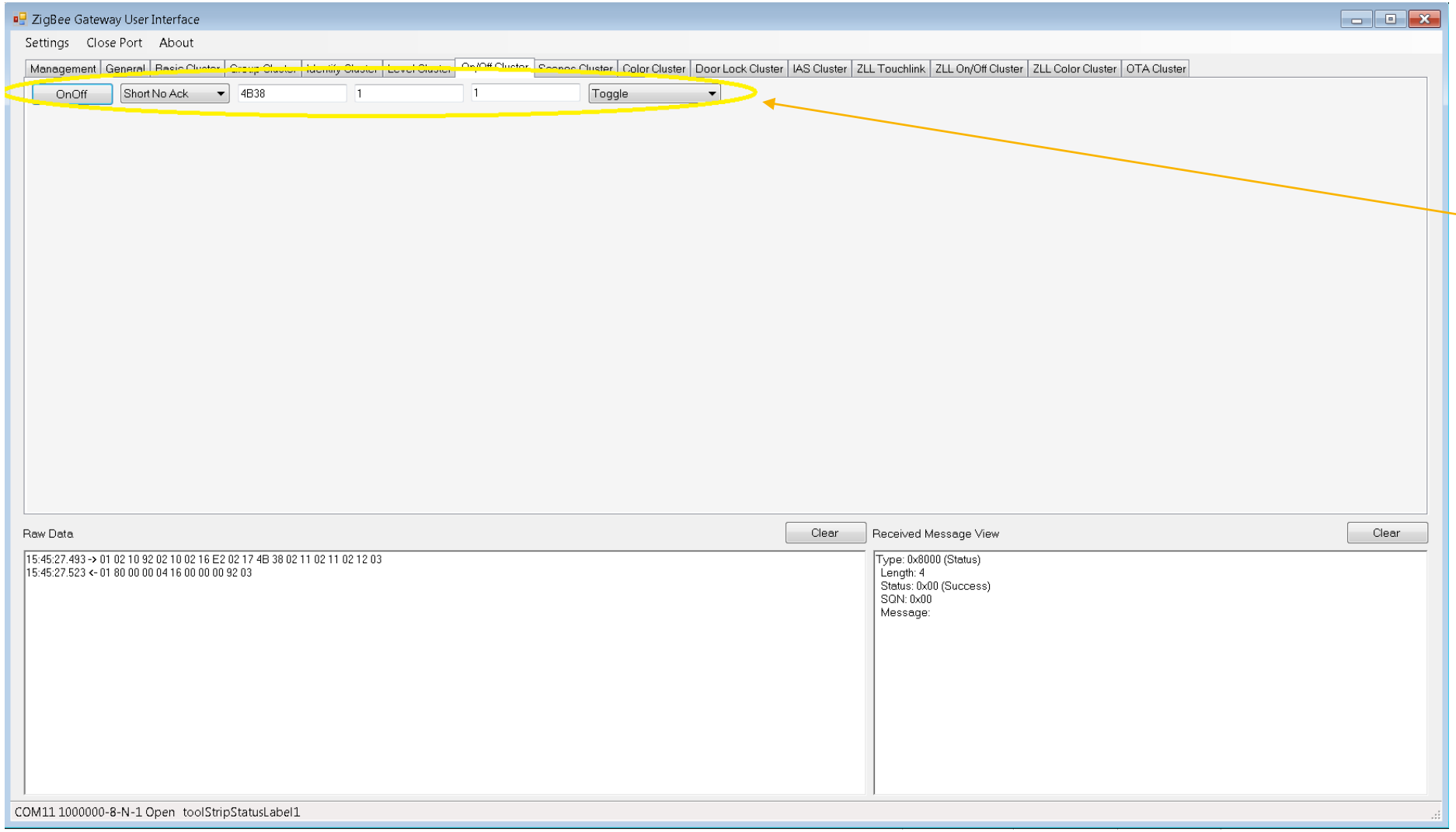
Step 2

HANDS ON MODULE 3: ADDRESSING MODES

Overview

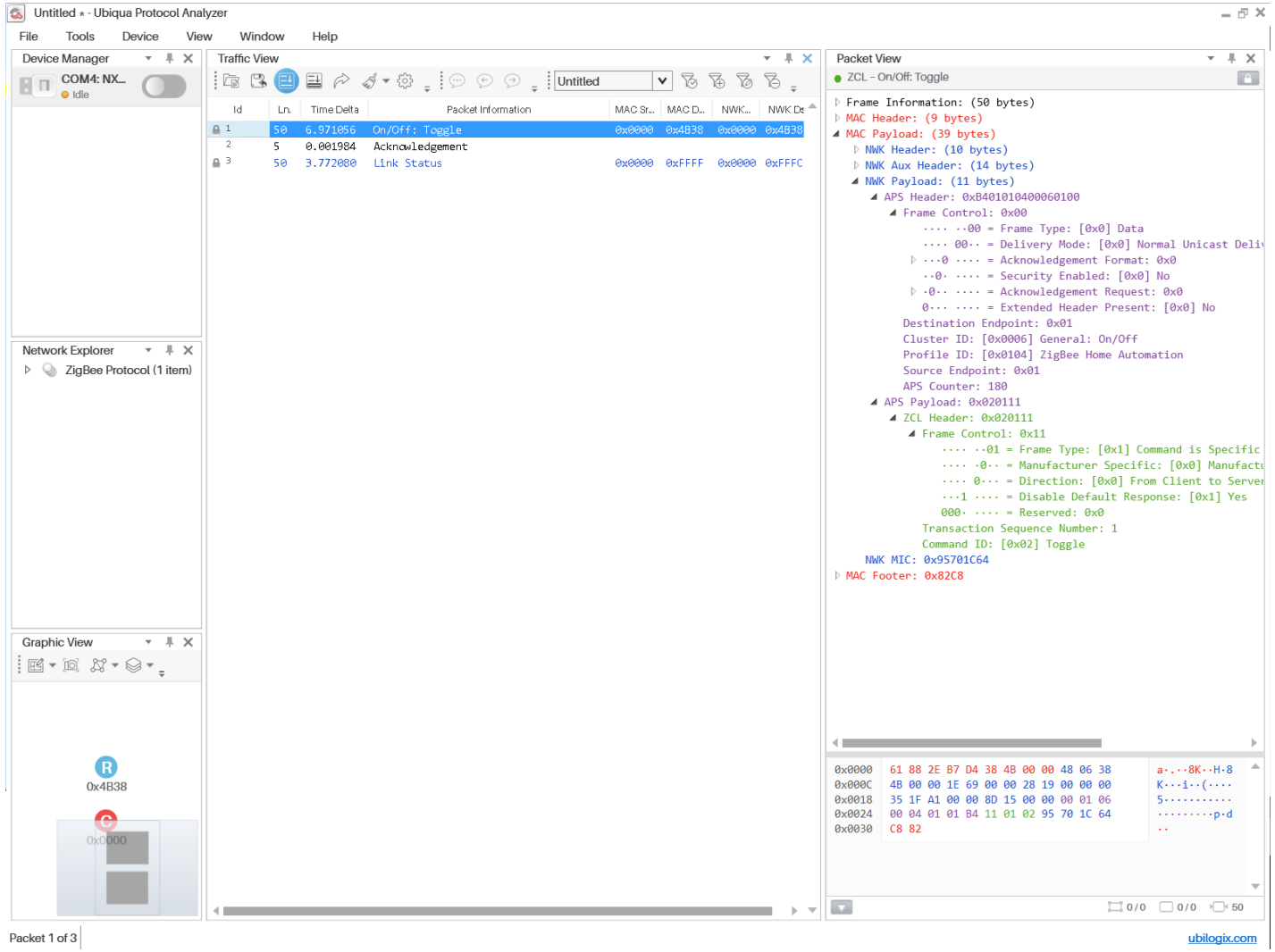
- Module 3: Addressing modes (30 mins)
  - Toggle Light Bulb state by sending On/Off command using short address (no APS ACK)
  - Add Light Bulb to a group
  - Toggle Light Bulb state by sending On/Off command using the group address

Toggle Light Bulb – Short Address No APS ACK [1]

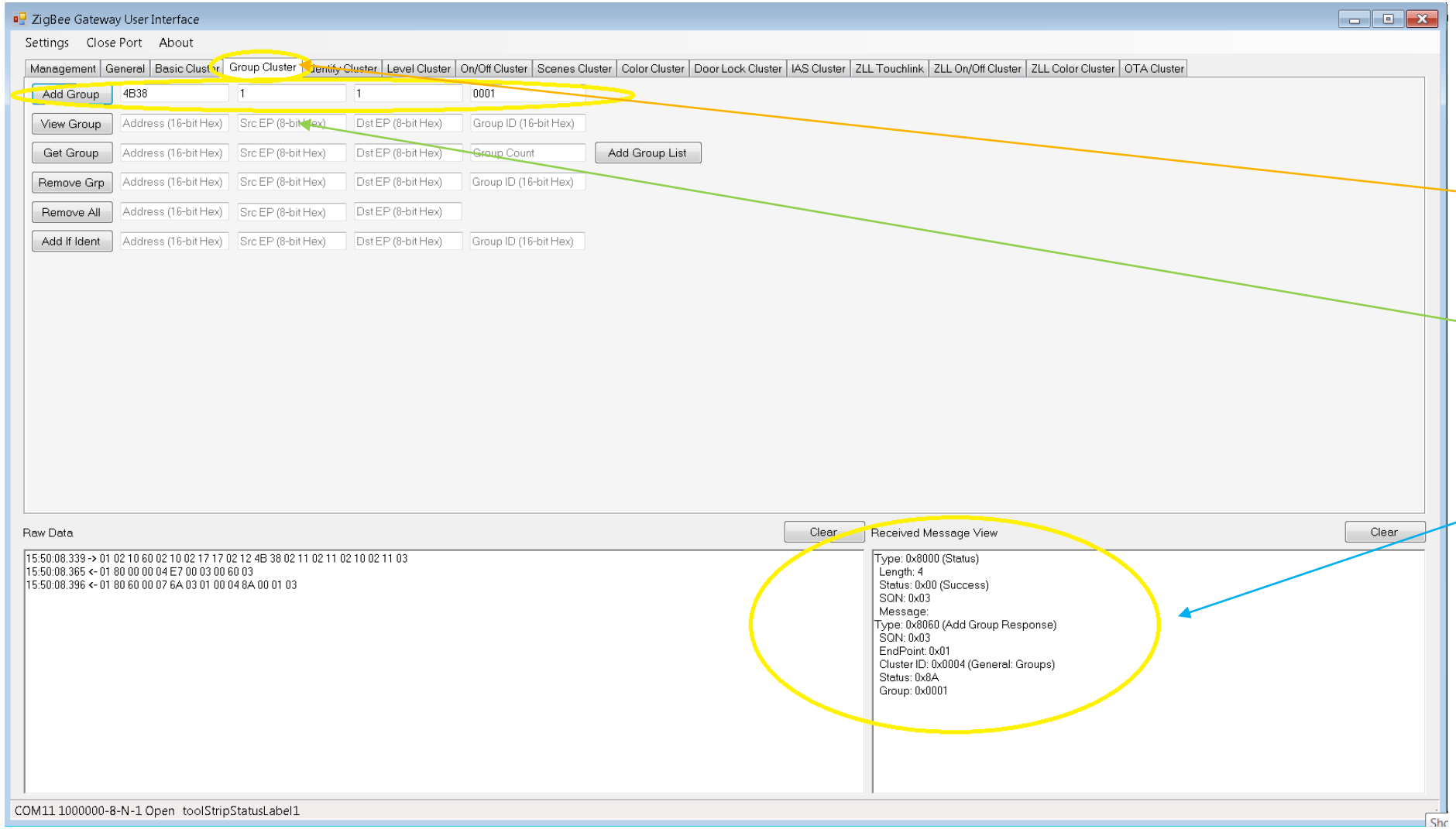


Step 1

Toggle Light Bulb – Short Address No APS ACK [2]



Add Light Bulb to a Group

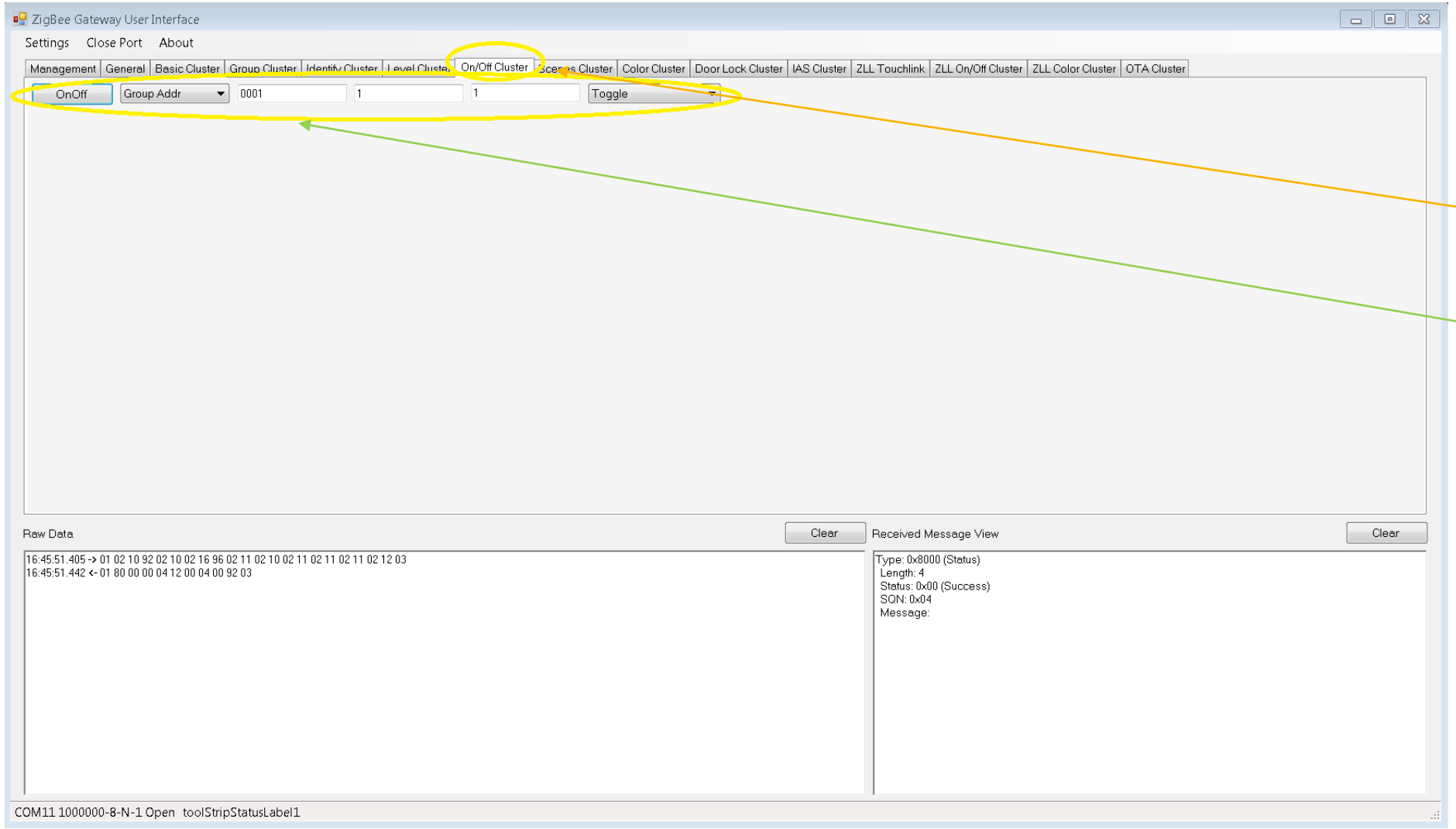


Step 1

Step 2

Step 3

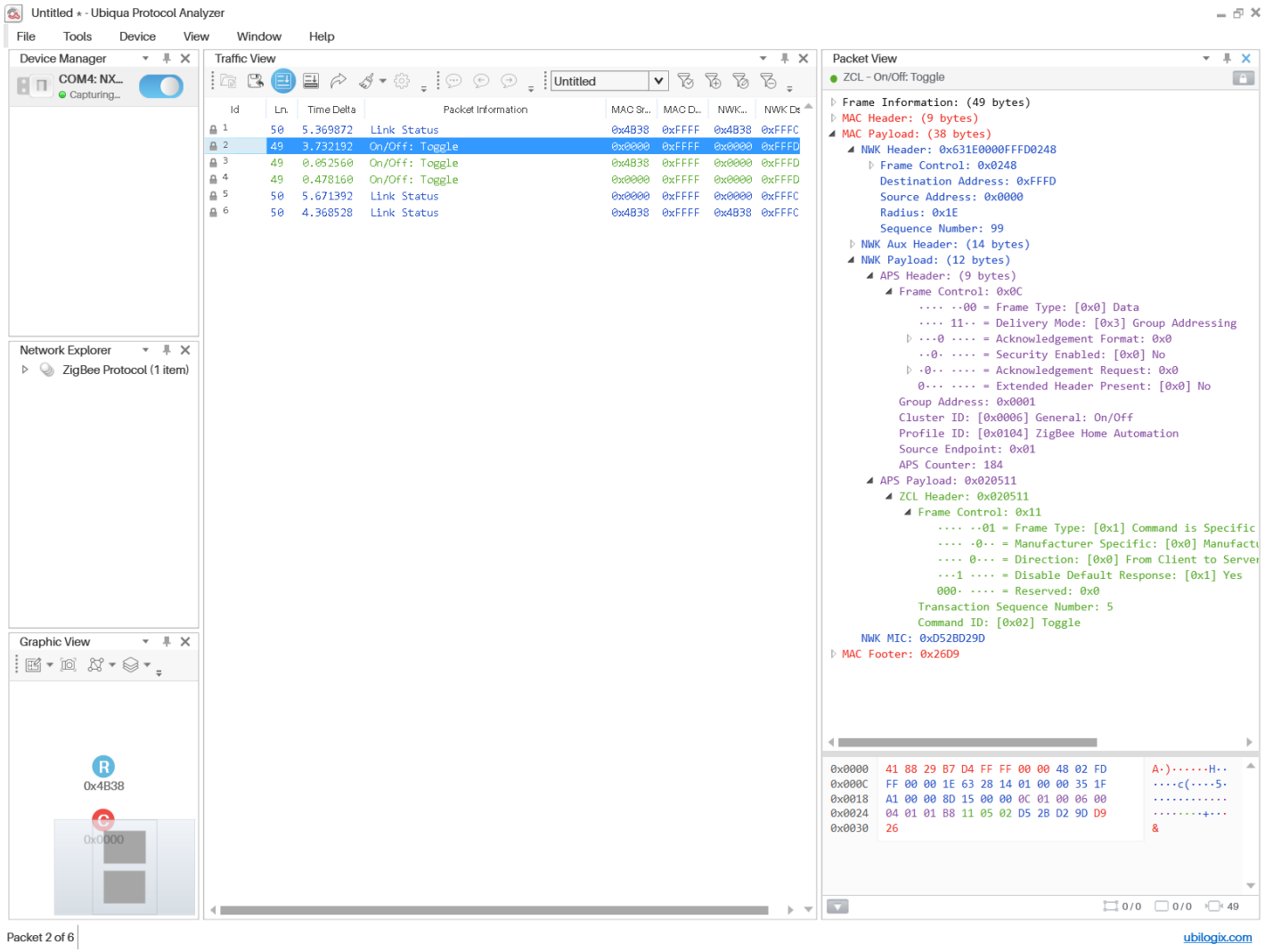
Toggle Light Bulb – Group Address [1]



Step 1

Step 2

Toggle Light Bulb – Group Address [2]

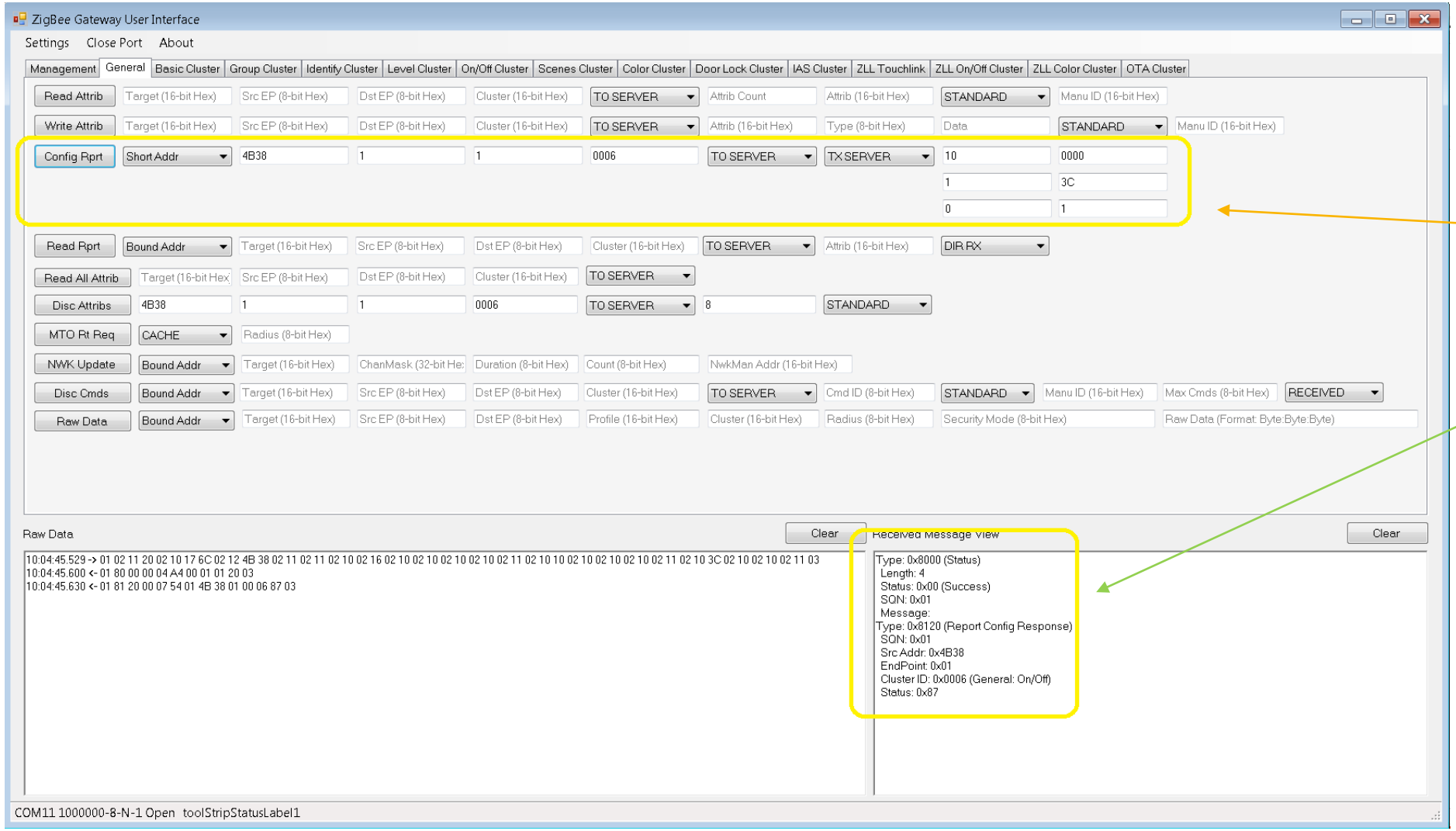


HANDS ON MODULE 4:
USING REPORTING

Overview

- Module 4: Using Reporting (30 mins)
  - Use ZGWUI Tool to configure Light Bulb to report change of On/Off state.
  - Bind Light Bulb address to coordinator.
  - Wait for periodic report and view using ZGWUI and sniffer.
  - Send Toggle Command to Light Bulb to generate report and view using ZGWUI
  - Power cycle Light Bulb to generate report and view using ZGWUI.

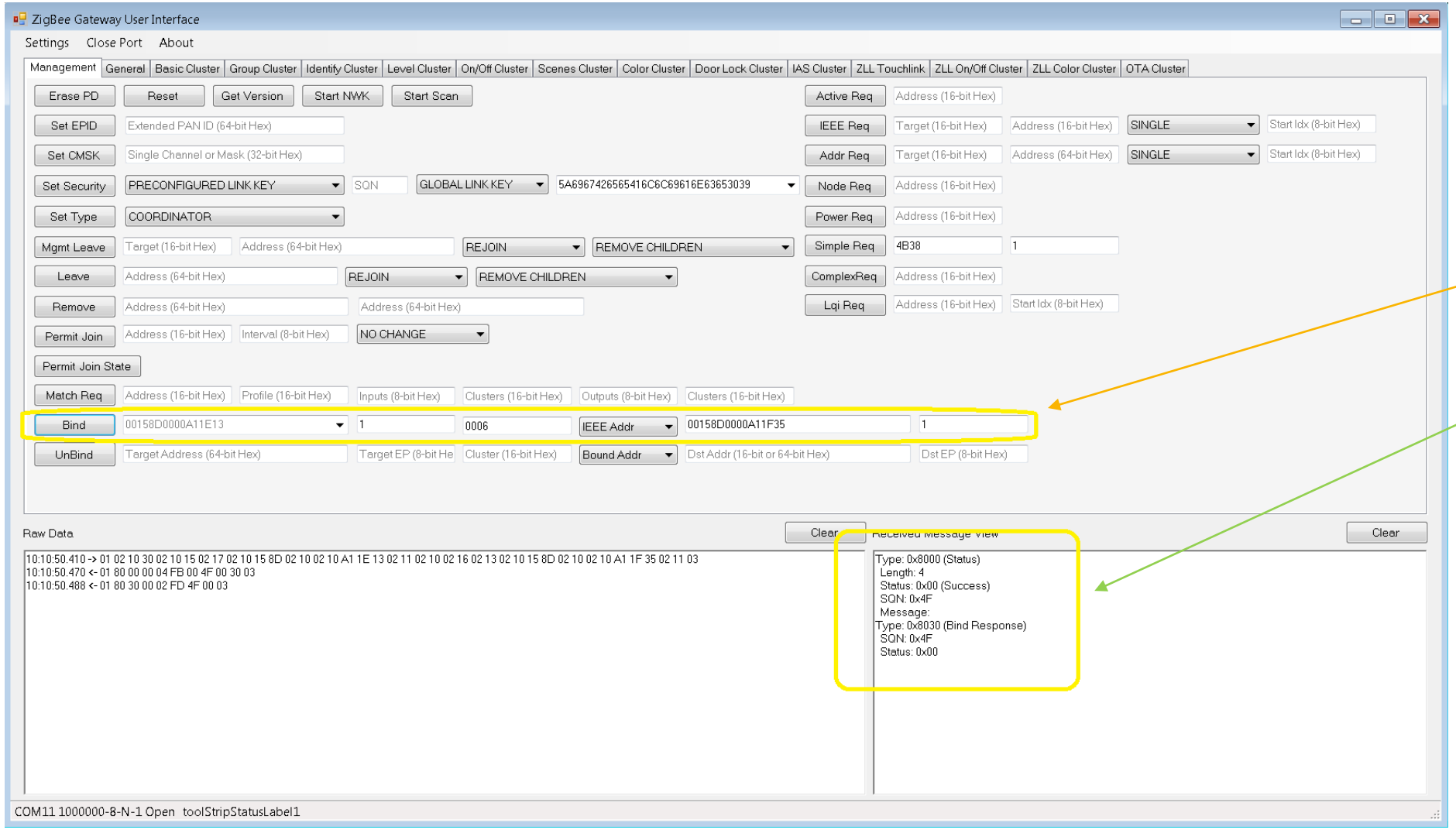
Configure Reporting



Step 1

Step 2

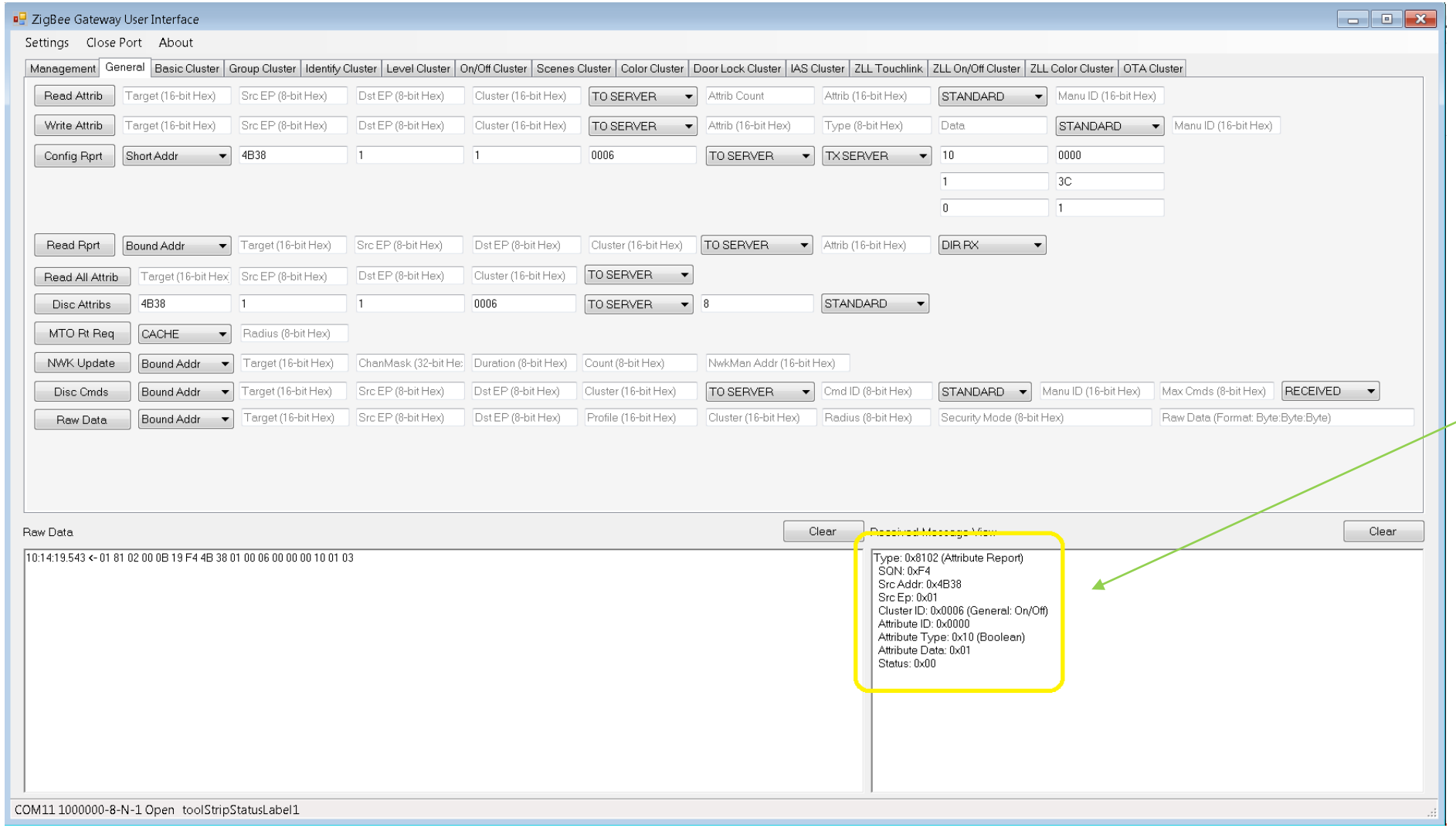
Bind Light Bulb to Coordinator



Step 3

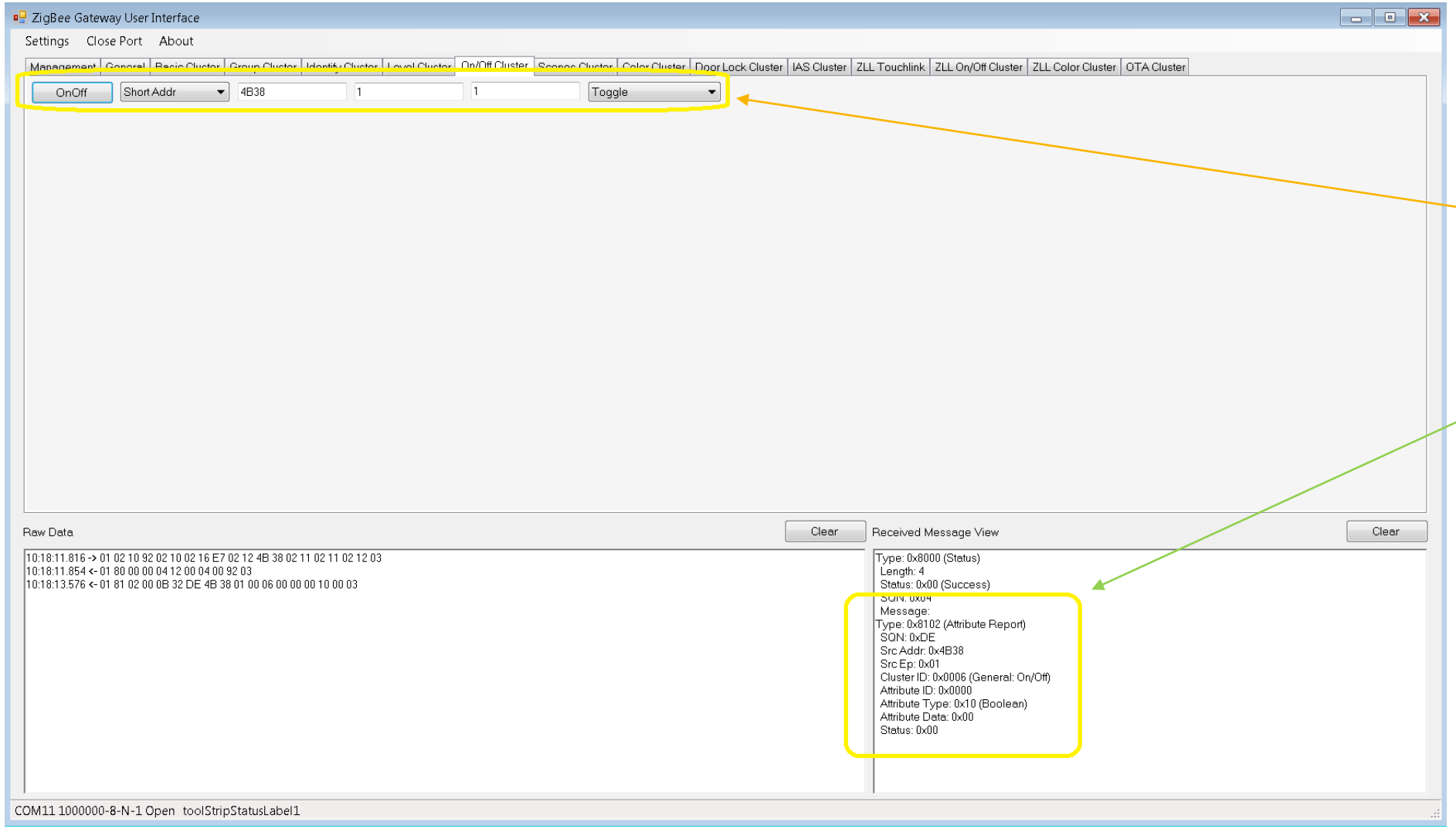
Step 4

View Periodic Report



Step 5

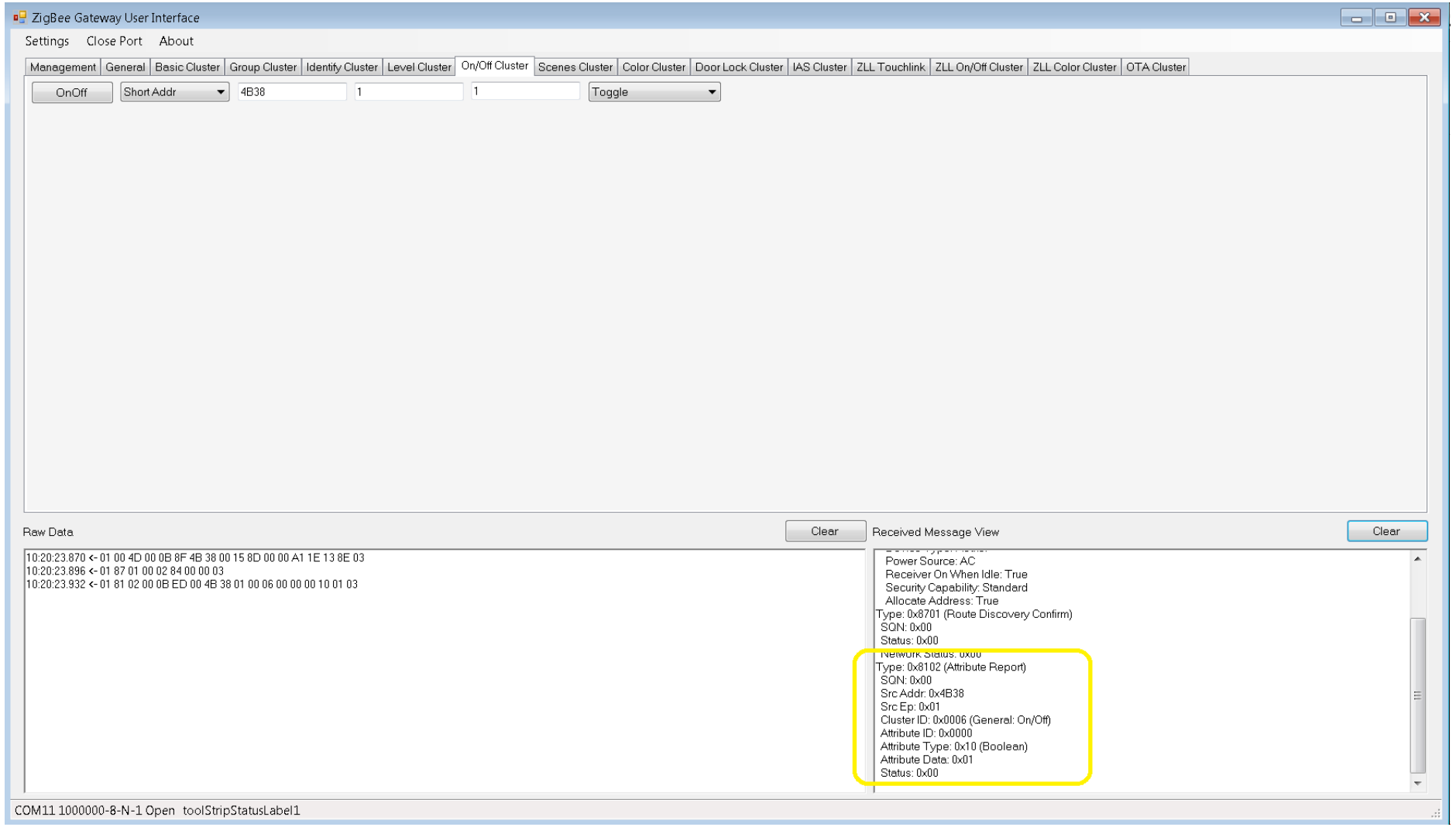
Toggle Light Bulb State and View Report



Step 1

Step 2

Power Cycle Light Bulb & View Report



NXP ZigBee 3.0 Development Tools (Hardware and Software)

- Details regarding development tools are available at:

http://www.nxp.com/products/interface-and-connectivity/wireless-connectivity/2.4-ghz-wireless-solutions/zigbee/zigbee-3.0:ZIGBEE-3-0

SECURE CONNECTIONS
FOR A SMARTER WORLD