CHAPTER 8

Commissioning ZigBee Networks

Commissioning is the process of connecting ZigBee applications to each other. This process, while simple in concept, can be fairly complicated to do well and can involve quite a few steps. Commissioning includes:

- Searching for appropriate networks to join
- Joining the correct network
- Determining which nodes on the network to talk to
- Determining how to talk to those other nodes (groups, bindings, or directly)
- Determining if communication has stopped and what to do about it, perhaps even searching for the network on a different channel

Before diving into commissioning, it's worthwhile to talk about a man who served on another commission, and is rumored to have been the inspiration for the name ZigBee.

> *Zbigniew Brzezinski, Born in Warsaw Poland in 1928, was the National Security Advisor to Jimmy Carter. He later served during Ronald Reagan's administration on the NSC-Defense Department Commission on Integrated Long-Term Strategy for the United States.*
>
> *Always a human rights activist, in 1989 Brzezinski toured Russia and visited a memorial to the Katyn Massacre. He asked the Soviet government to acknowledge the truth about the event for which he received a standing ovation from the Soviet Academy of Sciences. The Berlin Wall fell not ten days later, significantly changing the political landscape in Eastern Europe.*
>
> *Brzezinski participated in the formation of the Trilateral Commission during the 1970s and 1980s, in order to more closely cement U.S.-Japanese-European relations, just as the ZigBee Alliance works toward an international standard developed and supported by companies in all these regions.*

> *Brzezinski is currently a professor of American foreign policy at Johns Hopkins University's School of Advanced International Studies.*
>
> *The rumor goes that in his concern for national security he conceived of a network of sensor devices that could be easily deployed, and if a device was destroyed or removed, the network would continue to operate. It was his inspiration, along with a grant to MIT, which was the kernel of an idea that was later to be named for him. But since no one would be able to spell or pronounce Zbigniew Brzezinski, they shortened it to Zig B. or ZigBee.*

8.1 Commissioning Overview

Commissioning is the process of configuring the nodes in the network so they can communicate data to each other. Imagine that you are a ZigBee node. You want to join a network, find an application to talk with, and get to work. So you send out a beacon request. Perhaps dozens of ZigBee networks respond, all within hearing range (see Figure 8.1). Which one should you join?

Figure 8.1: Which Network to Join?

Okay, so somehow, magically (to be explained in this chapter), you join the right network. You also need some security so you don't control the neighbor's network and they don't control yours. Somehow you obtain a security key for the network. Now, how do you find which other nodes in that network to talk to? Perhaps you are a light switch. Well, if you searched the network you would see there are 43 lights on it. Should you control them all? A group of them? Only one? Which one? How is a simple switch to decide?

ZigBee provides a robust set of primitives to accomplish all aspects of commissioning. To help understand these primitives in common usage, I'll describe three distinct scenarios which cover the majority of commissioning issues:

- Simple commissioning
- Butterfly commissioning
- Custom commissioning

The first two scenarios deal with devices that must interoperate between multiple manufacturers (or OEMs), without knowledge of which other devices may be in the network before they join. The last one assumes the manufacturer knows something about the network, and perhaps even defines all the nodes in it.

ZigBee commissioning follows a concept called the "Butterfly Model" (see Figure 8.2). The idea is that a device is "born" with whatever information can reasonably be configured in a static manner and at manufacturing time. At its most basic, and in the absence of any other configuration information, a fresh device will join the first network that offers itself. After joining, it "digests" information that prepares it for its next life, and so on. After some number of these phases it arrives at its stable operational state.

Figure 8.2: ZigBee Commissioning Uses Butterfly Mode

Every device has the ability to get back to the original “factory” state, in case something goes wrong, or the installer wants to start over (perhaps to move the device to another network). This is often called a factory reset. Remember, ZigBee devices store their current state in non-volatile memory to survive power outages, so just removing power and rebooting won’t do it. The device has to forget what it has learned.

The commissioning primitives can be found in a variety of ZigBee components, including the NWK layer, APS, ZDO, ZDP, and the ZigBee Cluster Library. An application running in a node has full access to all of these primitives. Most commissioning primitives are available in both local and over-the-air forms:

- *ZigBee Device Object (ZDO)* contains methods for finding and joining the network in various ways, calling on the NWK layer to do some of this work.
- *ZigBee Device Profile (ZDP)* contains device (node) and service (application) discovery, as well as remote table-management functions.
- *ZigBee Cluster Library (ZCL)* provides over-the-air group and scene management.
- *The Commissioning Cluster* provides a standard over-the-air means for setting up security keys, PAN IDs, the channel mask, and manager addresses.

ZDP and ZDO were largely written by Don Sturek, currently of Texas Instruments. He was the technical editor for the APS, ZDP, and ZDO portions of the ZigBee specification, and was chair of The Architecture Committee (TAG) for ZigBee.

ZDO is really the portion of ZigBee that decides which network to join. It bases this decision on a set of fields from the various information bases found in the ZigBee layers. If a commissioning tool (such as a remote PC or a handheld) is used, the commissioning cluster can remotely instruct ZDO to start up in various ways, but it’s always ZDO that does the work inside the node.

The fields used by ZDO to form or join the network are shown in Table 8.1.

For ZigBee public profiles, these are normally set at the factory to include nothing but a MAC address in a widget. Other fields are set to all 0x00s, or all 0xffs (to indicate that they are not set). But, in a private profile, they may be set.

ZDP, ZCL, and the Commissioning Cluster are used only after a node is on a network.

Table 8.1: ZDO Startup Fields

Field	Description
MAC Address	The MAC address is a unique 64-bit number assigned at manufacturing time. It is never changed.
apsChannelMask	The channel mask describes which channels should be scanned when forming or joining. The field is a 32-bit bit-mask of 802.15.4 channels. Only channels 11 through 26 are valid for ZigBee, which operates in the 2.4 GHz RF spectrum. Examples: 0x00000800 = channel 11 (bit 11) 0x04001000 = channels 12 and 26 (bits 12 and 26) 0x07fff800 = all channels (11–26)
apsUseExtendedPANID	This is a 64-bit address (similar to the MAC address). Set this to all 0x00s for ZigBee Application Profiles. It is sometime set to a specific number for private profiles.
nwkPANId	Always set to 0xffff (no PAN ID). This will choose a random PAN ID.
nwkNetworkAddress	Set to 0xffff to indicate no short address.
nwkStackProfile	Set this to the preferred stack profile (0x01 or 0x02). Most Application Profiles allow the node to join either stack profile.
TrustCenterAddress	Normally set to 0x0000. Can be set to a different short address if the TC is not on the ZigBee Coordinator.
NetworkKey	Set to all 0x00s if no network key. Set to a specific network key for preconfigured keys. Usually given to the node by the network in Home Automation.
NwkKeySeqNum	Only relevant to over-the-air commissioning tools.
TrustCenterMasterKey	Only relevant High Security networks. This is defined either by the Application Profile or a commissioning tool.

The major ZDP commissioning commands are shown in the following list. See Chapter 5, " ZigBee, ZDO, and ZDP," for a more detailed explanation:

- **ZDP-Bind**, **ZDP-Unbind**, and **ZDP-End-Device-Bind** add and remove entries from remote binding tables.
- **ZDP-Simple-Descriptor-Request** and **ZDP-Active-Endpoint-Request** and **ZDP-Match-Descriptor** determine which applications exist on remote nodes.

- **ZDP-IEEE-Address-Request** can find all the nodes on the network (just start at the ZC).
- **ZDP-Mgmt-Bind** can determine which applications are bound to which.
- **ZDP-Permit-Joining-Request** enables and disables permit-join in the network.

The major ZCL commissioning commands include the following. See Chapter 6 for a more detailed explanation of these commands.

- **ZCL-Add-Group-Request** and **ZCL-Remote-Group-Request** allow nodes to add and remove groups

Use a combination of **ZDP-Mgmt-Bind** and **ZCL-Get-Group-Membership** to determine which groups are used in the network.

ZCL-Add-Scene and **ZCL-Remove-Scene** and **ZCL-Remove-All-Scenes** to add and remove scenes for disparate devices (a switch, thermostat, and window shade can all react in some way to a scene).

Which brings us to the Commissioning Cluster, and the next section.

> Commissioning is the process of setting up ZigBee nodes so that they can communicate data.

8.2 The Commissioning Cluster

The specification for ZDP did not include the ability to query or set the various startup parameters used by ZDO, including PAN ID, extended PAN ID, security key, and network address. This was done purposely, because in simple commissioning (see the next section) it is not needed. Since ZDP is required in every ZigBee device, it was decided to keep ZDP small and instead add this optional feature to the ZigBee Cluster Library as the Commissioning Cluster.

The Commissioning Cluster is very useful in commercial building environments. Network deployment and operation in these environments are often planned to a high degree and they often involve multiple, simultaneous installers during construction or retrofit. They often have more nodes in a given network and often contain many separate, but overlapping networks.

- Startup Attribute Set (SAS)
- Join Parameters Attribute Set

- End-Device Parameters Attribute Set
- Concentrator Parameters Attribute Set

The startup attribute set is used to find the right network and to set up any preconfigured security material. The join parameters control the frequency of joining. The end device parameters determine how often the end device polls its parent and when it searches for a new parent. The concentrator parameters are unique to data concentrators (or gateways as they are sometimes called).

The following table briefly describes the SAS parameters. Although the Commissioning Cluster is optional, every node contains these fields, as they are needed to form or join a network. They are set through the APSME-SET or NLME-SET functions, which affect the APS and NWK information bases.

The SAS parameters (see Table 8.2) only take effect after the node is reset by the Commissioning Cluster command Restart Device Request. Until then, they are merely settings in a table in RAM in the remote node.

The startup control is the most interesting field. It contains instructions on the way to reset the node. Mode 0x00 means *silent join*, which is the node contains everything it needs to know, just as if it had already joined that new network, including all security keys, etc.… Mode 0x01 is for ZC capable nodes, and tells it to form a network. Mode 0x02 tells it to rejoin, as if the node had lost contact with its parent. The nice thing about this mode is permit-join can be disabled and the node can still "join." Mode 0x03 tells the node to join from scratch. Permit-join must be enabled for the node to join.

If the node will be joining, the Join Parameters attributes may also be set (see Table 8.3).

The rejoin interval determines how often the node will attempt the join or rejoin. Usually, the node will attempt this frequently at first, then gradually slow down so as not to flood the air with useless rejoin attempts, in essence, assuming that the network will come back later.

The End Device Parameters Attribute set (see Table 8.4) is used only on ZigBee end devices.

Sleepy (RxOnIdle = FALSE) ZigBee End devices (ZEDs) poll their parent at intervals for messages. This allows the Commissioning Cluster to determine the interval, which affects battery life. The ParentRetryThreshold applies to both sleepy and wakeful ZEDs. If they just can't communicate with their parent (their only link to the network), then they must eventually find a new parent.

Table 8.2: Startup Attribute Set

SAS Attribute	Description
ShortAddress	This is the 16-bit network short address. Set to 0x0000–0xfff7 for valid addresses, or 0xffff to indicate it is not yet established.
ExtendedPANId	Which extended PAN ID will the node form or join when reset? ExtendedPANId 0x00f0c27710000000 is the commissioning extended PAN. Set to all 0x00s to indicate any extended PAN.
PANId	Normally starting out as 0xffff, which means choose a random PAN ID. Most applications care only about the extended PAN ID. Both PANId and ExtendedPANId are found in beacons.
ChannelMask	A 32-bit mask for deciding which channels to search when forming or joining. Only bits 11–26 may be set. It takes time (about 1/3 to 1 second) for each channel scanned.
ProtocolVersion	Always set to 0x02.
StackProfile	Set to 0x01 or 0x02, the preferred stack profile.
StartupControl	0x00—silent join. 0x01—form a network. 0x02—rejoin a network. 0x03—associate join a network.
TrustCenterAddress	A short address to find the trust center. This is required in high security. Normally 0x0000 (the ZC).
TrustCenterMasterKey	The master key is used to establish a link key with the trust center through SKKE.
NetworkKey	The network key
UseInsecureJoin	Set to TRUE for standard security, FALSE for high security.
PreconfiguredLinkKey	Assumes SKKE has already been performed.
NetworkKeySeqNum	Key sequence number for the network key. A node may have more than 1 network key (old and new).
NetworkKeyType	Set to 0x01 for standard security, 0x05 for high security.
NetworkManagerAddress	Normally set to 0x0000 (the ZC). This node is in charge of frequency agility, if enabled.

ZigBee has the concept of a gateway. Many-to-one routing allows concentrators to easily operate as a gateway without consuming too many mesh-networking resources. This process is described in detail in Appendix A, "ZigBee 2007 and ZigBee Pro."

Table 8.3: Join Parameters Attribute Set

Join Parameters Attribute	Description
ScanAttempts	From 1 to 0xff. 0xff means forever.
TimeBetweenScans	From 1 to 0xffff, in milliseconds.
RejoinInterval	Lower bounds for rejoining, in seconds. Defaults to 60.
MaxRejoinInterval	Upper bounds for rejoining, in seconds. Defaults to 1 hour.

Table 8.4: End Device Parameters Attribute Set

End Device Parameters Attribute Set	Description
IndirectPollRate	The rate, in milliseconds, to poll the parent
ParentRetryThreshold	The number of failed attempts to contact a parent that will cause a "find new parent" procedure to be initiated

Any router node may serve as a gateway. Data concentrators, or gateways, have a few special parameters, as shown in Table 8.5.

The idea behind using the commissioning cluster is so that a commissioning tool (either a handheld or PC) can act as a commissioning network. This network then uses the Commissioning Cluster (via the ZCL-Set-Attributes command) to set up the new node. Then the commissioning tool tells the new node to reset using one of the Commissioning Cluster commands shown in Table 8.6.

Restart Device tells the remote node to restart using the parameters set up in the commissioning attributes. Some nodes can handle more than one attribute set for the commissioning cluster. If this is the case, then the save/restore save and restore to

Table 8.5: Concentrator Parameters Attribute Set

Concentrator Parameters Attribute Set	Description
ConcentratorFlag	After restarting, will this node be a concentrator or not? Assumes that the commissioning tool already knows.
ConcentratorRadius	To what radius will this discover the many-to-one route? The default (and maximum) is 5.
ConcentratorDiscoveryTime	Many-to-one route discovery can occur automatically (1 through 0xffff seconds) or manually (0x0000).

Table 8.6: Commissioning Cluster Commands

ID	Command	Mandatory/Optional
0x00	Restart Device Request	Mandatory
0x01	Save Startup Parameters Request	Optional
0x02	Restore Startup Parameters Request	Optional
0x03	Reset Startup Parameters Request	Mandatory

the primary commissioning cluster attribute sets. The Reset Startup Parameters restores the “factory” defaults.

> The Commissioning Cluster allows over-the-air setup of the startup procedure.
>
> The Commissioning Cluster is optional.

8.3 Example 1: Simple Commissioning

One of the simplest commissioning techniques is that employed by the ZigBee Home Automation Application Profile. The installer of this network is expected to be either a home owner or a professional installer. Either way, the installer is definitely *not* expected to be computer expert. In the simplest case, the installer turns on the devices and they just work. In more complicated situations, pressing a few buttons does the trick.

As an example, assume I just bought a ZigBee system from The Home Depot. The “Porch Light” starter pack comes with two battery-operated switches, four lights, and a remote control (see Figure 8.3). The purpose of this simple home automation system is to turn on the front and back porch lights when I drive home (for safety), and then to turn them off automatically after I’ve left (for energy savings). The porch lights can also be turned on or off from either of the switches or from the remote. One switch is to be placed just inside the front door, and the other is placed in my bedroom. The next time I go to bed and realize that I didn’t turn out the porch lights (again), pressing a bedside button solves the problem.

Since this is a retro-fit system, the ZigBee lights are little gizmos that screw into an existing light socket (probably ceiling cans), and then the light bulb screws into the gizmos. The switches look like normal wall switches and can either be stuck on a cement or brick wall, or recessed into the drywall by cutting a small square hole. The remote unit clips onto the sun visor in my car.

Figure 8.3: ZigBee Simple Commissioning

The kit also comes with a small little box, called a ZigBee Network Controller. The ZNC plugs into my DSL modem, cable modem, or WiFi™ router, if I have one. If not, it just plugs into the wall.

The user instructions that might come with the kit could be quite simple:

1. Warning! Remove power to the porch lights (either at the circuit breaker or wall switch).
2. Plug the ZigBee Network Controller into a wall socket. A green LED comes on, to indicate that everything is okay.
3. Unscrew the existing light bulbs. Screw the ZigBee Light Base-Units into sockets. Screw the light bulbs back into the ZigBee Light Base-Units.
4. Remove the clear plastic tab from the battery holders on the switch units.
5. Press the "Add ZigBee Devices" button on the ZigBee Network Controller. The green LED will flicker. When it becomes solid again, all the ZigBee devices are connected to the network.
6. Test each light switch. Be sure each switch turns the lights on and off. If not, see the troubleshooting section.
7. Install one light switch by the door. Install one light switch in the bedroom. Place the remote control in the car.

Even if the purchaser lost the instructions, they'd probably try what's described above: plug everything in, turn it on, and start pressing buttons. It's that simple.

So how does this work from a ZigBee developer perspective? It starts with setting up the right commissioning parameters. The Home Automation Application Profile specifies this very clearly.

The node starts out knowing nothing but its own MAC address and Application Profile. It has no PAN ID (set to 0xffff), it has no extended PAN ID (set to 0x0000000000000000), it has no channel (channel mask is set to 0x07fff800), it has no network key, and it has no profile ID (it will join either stack profile 0x01 or 0x02).

When power is applied, the node begins scanning for networks. As soon as it finds one with permit-join enabled, it joins that network and receives its short address and network key (in the clear with a transport key) from its parent.

The only widget which forms a ZigBee network in the example above is the ZigBee Network Coordinator (ZC). The ZC will form a network using its MAC address as its extended PAN ID, and a random PAN ID. It will form the PAN randomly on one of the preferred channels for HA: 11, 14, 15, 19, 20, 24, or 25. The ZC will leave permit-join enabled for two minutes when it is first plugged in or after the user presses the "Add ZigBee Devices." Once it has formed the network, it saves this information to non-volatile memory, so that power can be interrupted with no harm to the network. It also includes a recessed "reset" switch that can put it back into factory reset.

The switches and remote control devices are sleepy ZigBee End devices (ZEDs). The lights are ZigBee Routers (ZRs), both of which are device types that attempt to join a ZigBee network. Remember, they will join any network, any PAN ID, any channel. The ZRs and ZEDs will scan for networks on all the channels and then attempt to join. They'll repeat this several times before backing off to trying once a minute. If the user presses a switch and it still hasn't joined a network, then it will attempt to do so right then (immediate user response).

The only potential problem with this easy-joining scheme occurs if two or more installers (perhaps neighbors in an apartment or condominium) attempt to add ZigBee devices to their network at the same time. This is unlikely, but possible. The troubleshooting solution is simply to reset the joining devices and try again.

Once on the network, the ZRs and ZEDs record the channel, PAN ID, extended PAN ID, network (short) address, and network security key into non-volatile memory. They can be powered off or the batteries can be changed, and they still retain the knowledge of which network they are on.

The ZED switches have been preconfigured in their binding tables to use a specific group ID for the destination of the HA:On/Off:Toggle command. A hard-coded group ID can be used because group IDs are unique to each network.

The smart automobile remote takes more explanation. How can a remote automatically turn on the lights when I drive home, without pressing any buttons? There is a fairly easy way to do this.

> The device has two modes: "home" and "away."
>
> Every 30 seconds, the device wakes up and attempts to communicate with its parent.
>
> If it was "home" and can no longer communicate, it goes to "away" state. If it is "away" and it can now communicate, it goes to "home" state.
>
> If "away," it will attempt a rejoin every 30 seconds, which may get the same parent or not. When rejoin is successful, the node goes to "home" state.
>
> When the device goes from "away" to "home," it sends the HA:On/Off:On command. Voilà!

Later, this system could be expanded to include more lights and switches, heating and cooling control, and a complete home theater system. The devices are standardized, regardless of who the manufacturer might be. They all work the same way, and they can all be configured over-the-air with bindings, groups, and scenes.

Even the original devices from that starter pack can be reconfigured to include new bindings. Remember how the lights began as belonging to a group, and the switches sent to the group? Well, the switches can be set to control other devices with simple button presses. Press a button on a light, another on a switch, and they can be bound using ZDP-End-device-Bind. The same goes for a thermostat and a temperature sensor.

The ZigBee Network Coordinator, because it has an Ethernet connection to plug into a wireless router or DSL modem, can be monitored or controlled by a laptop, PC, or even over the Internet with a little bit of PC software. A nice drag-and-drop, menu-driven program on the PC (with a ZigBee USB dongle) or a television (ZigBee-enabled) makes reconfiguring the house easy, even for novices.

The primitives are all there. It just takes a little imagination to use them to great effect.

You've seen all these ZigBee primitives before. See Chapter 5, "ZigBee, ZDO, and ZDP," for ZDP (binding) commands; Chapter 6, "The ZigBee Cluster Library," for ZCL (groups and scene) commands; Chapter 5, "ZigBee, ZDO, and ZDP;" and Chapter 7, "The ZigBee Networking Layer," for ZDO and NWK (joining) commands.

To see the "Porch Lights" example in full source code, complete with an over-the-air capture, go to the http://www.zigbookexamples.com Web site. Keep in mind the source code is for the Freescale platform, but the concepts apply to all ZigBee platforms.

The example uses four ZigBee boards, all NVM-enabled so they will remember their settings even across battery changes or reboots:

- **ZcNcbZnc**: The ZigBee network controller. It starts up automatically. SW1 opens the network for one minute to add other devices.
- **ZrSrbLight**: A porch light. Uses a random MAC address, so multiple lights can be programmed. It has no switches.
- **ZedSrbSwitch**: The switch that is placed by the front door, or in the bedroom. SW1 toggles the remote light.
- **ZedSrbCarRemote**: The remote that operates in the same way as the switch, except that it automatically turns on the light when in range. To get it out of range, use a coffee can to cover it.

LSW4 puts all of these devices back to factory reset (forgetting what they learned in NVM).

Simple commissioning nodes will join any network.

Binding applications is often preconfigured or handled through button presses.

8.4 Example 2: Commercial Commissioning

Simple commission works well for the small, unplanned network. For larger networks, or large groups of multiple networks, such as those found in commercial buildings, hotels, and hospitals, simple commissioning is just, well, too simple.

Networks need to be more secure to operate in a commercial environment. Networks also must be planned, both for the installation process, and for operations and maintenance. Down time or delays in deployment can be very expensive.

For these applications, the network is planned out ahead of time. Blueprints are used to decide where the network components will go. Tests are run ahead of time to make sure there is sufficient ZigBee Router coverage. The rooms are usually built one-by-one as autonomous units, and the installer must not only verify that the network is functioning, but that all the equipment installed (lights, switches, thermostats, heating and cooling units, door locks, etc.) works as expected. A very detailed checklist is built and written with the training of the installer in mind who will usually be an electrician.

In larger installations, it is common to have multiple installers all working in the same area. Figure 8.4 shows a typical set of hotel rooms. Two installers are setting up adjacent rooms, taking devices from their boxes, installing them and then commissioning them to network together.

Each device is tested, repaired if there is a problem, and replaced if the problem can't be resolved. The installation tool may have a bar code scanner on it, so they can scan the device when getting it out of the box. In this way the commissioning tool knows about the device and can check it off a list, or do other intelligent behavior.

These devices (lights, switches, door locks, thermostats, etc.), which may originate from multiple OEMs (perhaps Philips lights, Trane thermostats, and Schneider Electric air conditioners) are set up out-of-the-box to automatically join the commissioning network on extended PAN ID 0x00f0c27710000000. ZigBee defines this special Extended PAN ID so that devices built by many OEMs can all be commissioned in the same way.

Think of it as two step process. First, get on the commissioning network and receive the commissioned data. Then, reset the node to join the operating network, and complete the commissioning process (see Figure 8.5).

If it is expected that there will only be multiple installers in any given vicinity, then the ZigBee standard commissioning extended PAN ID, 0x00f0c27710000000, is used only to

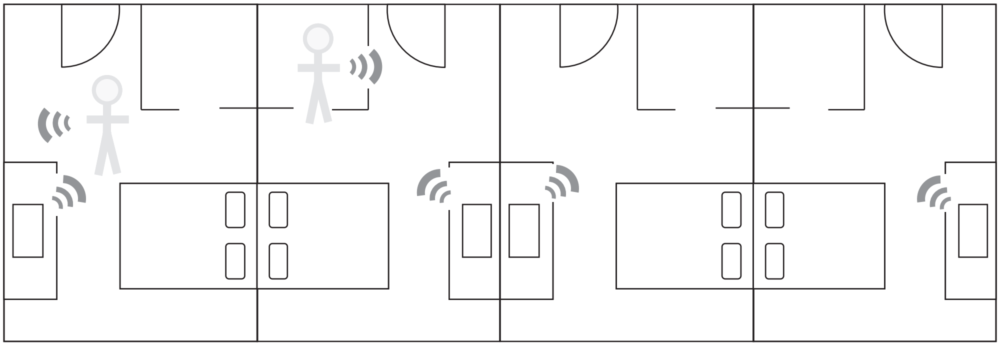

Figure 8.4: ZigBee Commercial Commissioning

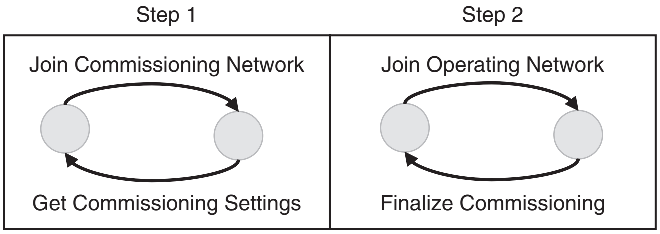

Figure 8.5: Commercial Commissioning Requires at Least Two Steps

get the extended PAN ID of the commissioning tool itself. Then the node is reset (via the Commissioning Cluster) to go onto the commissioning tool's PAN. ZigBee reserves the Extended PAN ID range of 0x00f0c27710000001 to 0x00f0c2771000ffff for the purpose of commissioning tools.

In ZigBee Stack Profile 0x01, the node usually receives its network short address when the node rejoins the new network via the Commissioning Cluster Reset Device command (with StartupControl set to mode 0x02).

But in stack profile 0x02, also called ZigBee Pro, it is possible to set the network short address explicitly. In this case, the commissioning tools would know the short address to assign to each node, perhaps based on some formula involving hotel room number. Then, when the node is reset, it can either use the rejoin as above, or even silent join (with StartupControl set to mode 0x00) so that it can be done a little faster (no waiting for negotiation with the trust center). It's assumed in this case that the tool has all the necessary keys and has communicated them to the nodes.

The example in this section, "CommercialCommissioning," uses the following nodes:

- *ZcNcbCommissioningTool*: Commissions (over-the-air) the light and switch. The LCD displays which node joined the commissioning network. A press of SW1 commissions the joined node and informs it to reset to the operating network. It has permit-joining on.
- *ZcSrbOperatingNetwork*: Pre-commissioned as the operating network. It has permit-joining off.
- *ZrSrbLight*: "Generic" light that is commissioned over-the-air.
- *ZrSrbSwitch*: "Generic" switch that is commissioned over-the-air.

To see the full source code and capture, go to the http://www.zigbookexamples.com website. As usual, the source code is for the Freescale platform. See Chapter 3, "The ZigBee Development Environment," for general instructions on using the source code. The basic steps for the demo are:

1. Download all the images. Turn all boards off.
2. Capture with Daintree (optional) on channel 25.
3. Boot the ZcSrbOperatingNetwork board. This lights all LEDs to indicate that it's the operating network.
4. Boot the ZcNcbCommissioningTool. This displays that information on the LCD.
5. Boot either the light or the switch. The LCD on the commissioning tool displays the fact that a light or switch has joined the commissioning network.
6. Note in the over-the-air capture that the node is associated with the commissioning cluster.
7. Press SW1. Watch as the new extended PAN ID is communicated over-the-air, and the node is instructed to reset. If the node is a switch, notice that the binding command is also given to it.
8. Boot the other board (either light or switch). Press SW1 on the commissioning tool again to provide the information to the other board, and to reset it.
9. Press SW1 on the ZrSrbSwitch. Notice that it toggles the light! This node is commissioned.

This is the basic process. Of course, there is much more to it, and commercial tools are beginning to come out from various vendors. At the time of this writing, both Daintree and Atalum sell commissioning tools.

Commercial Commissioning allows full planning of the network.

A special commissioning tool and a temporary commissioning network complete the process.

8.5 Example 3: Custom Commissioning

Devices that work with ZigBee public Application Profiles must work in a specific and generic way in order for them to interoperate with devices made by other OEMs. ZigBee specifies how commissioning works in each public profile very clearly. Private profiles do not have this restriction.

A private profile may commission the network in much more creative ways. For example, the entire network can work right out-of-the-box. Just order a kit: It forms a unique network on a unique PAN, and everything in the box is already connected and ready to communicate.

An application of this could be a manoverboard system for small pleasure craft and commercial fishing boats. Every router and battery-operated man-overboard device knows in advance exactly which network it is on, which extended PAN ID, which node address, which network key, everything. All the nodes essentially “silent join,” and the network begins operating correctly the moment the devices are turned on.

The example for custom commissioning is a merchandise quality-tracking system. In this system, a variety of sensors monitor the physical environment inside tractor trailer trucks, perhaps including temperature, humidity, and shock. The goal is to ensure that a load of frozen salmon arrived frozen, and has stayed frozen the entire journey, or that a cargo of produce (such as lettuce) was never too warm, too cold, or too dry (see Figure 8.6).

Today, the majority of these systems depend on manual spot-check inspection. An inspector will open boxes, examine some of the fish or lettuce, and make an educated guess as to whether the goods are within specification or not. This process is both

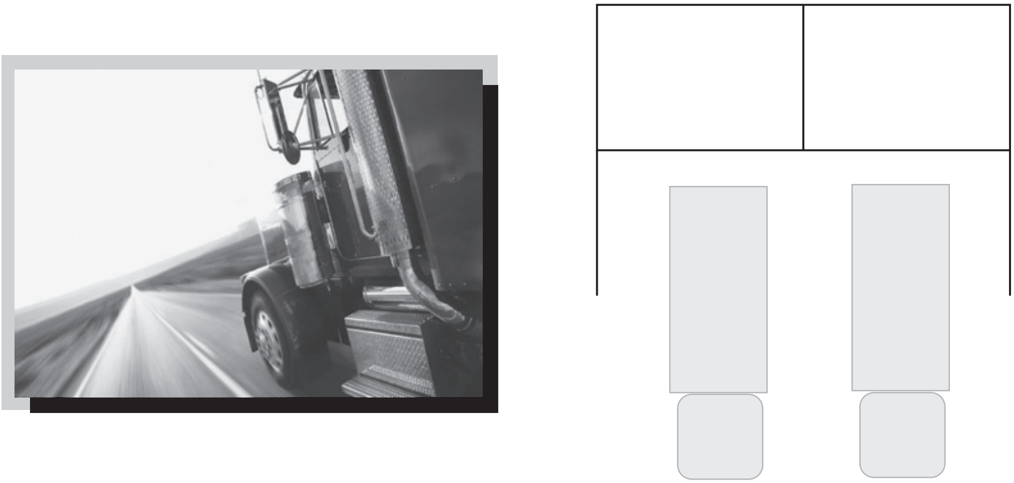

Figure 8.6: Loading Bays Could Use Custom Commissioning

time-consuming and is subject to error. Placing sensors in the vehicles to do this work makes sense. But how do we get that sensor data out of the truck? In this case, wireless monitoring makes perfect sense.

The sensors record data during the journey, and when the rear doors are opened, the sensor nodes automatically connect to a network in the docking bay, and download the data to a gateway, which is attached to a PC and a central tracking system.

All the nodes in this system are produced by the same manufacturer. So how do these nodes travel from one network to another? How do they get on the network in the loading bay and automatically download information? The procedure makes a few assumptions, an acceptable variation because it is a private profile:

- The gateway (to where the data will be sent) is assumed to be node 0x0000.
- Sensors, while in sensing mode, gather data once each minute.
- Once every three minutes, the sensors also send a beacon request (active scan) to scan for networks.
- Only networks in Extended PAN ID range 0x12345678xxxxABCD will be considered as potential networks to join.
- If the same network that was exited previously is found within 30 minutes, it is not joined.
- If a new network is found, or the same network is found 30 minutes or more after the sensor lost contact with all networks, the new network will be joined using the NWK-Rejoin command. This NWK-Rejoin is used so that permit-joining may always remain off, in the networks in the loading bays.
- Once joined, the sensors send all their data to the gateway (acknowledged). After the data is transmitted, it is forgotten, and the node leaves the network to go back into sensing mode.

Using these simple rules, the sensor nodes will always join a proper network for transmitting their data. They conserve power while not on the network, saving it for sensing operations and occasional data transmission.

Notice there is no binding. Nodes just send directly to the gateway (node 0x0000). Notice there is no over-the-air commissioning. The nodes know what they want to do before they are even started. These nodes don’t even need the standard ZigBee non-volatile memory to be enabled, except perhaps as storage for the data. But probably another storage mechanism, such as serial flash, will be used for data storage. Everything was commissioned into the nodes at the time of manufacture.

The example in this section, "Commercial Commissioning," uses the following nodes:

- *ZcNcbGateway*: Gathers the data. It transmits the data over the serial port (simple ASCII).
- *ZedSrbSensor*: Gathers temperature data once each minute and transmits it when it gets in range of the gateway.

Simply set the ZedSrbSensor outside, down the hall, or anywhere away from the Gateway node. Attach the Gateway and use HyperTerminal on that USB/COM port (mine is COM4) at 38,400 baud, 8N1. When the sensor comes within range, it will download the gathered data.

I've accelerated the process so that it works better for a demo. The demo gathers data once every 20 seconds, and only needs to be away from the gateway for one minute before it will link to it again.

"Custom Commissioning" is very flexible. All the ZigBee Device Object (ZDO), ZigBee Device Profile (ZDP), and ZigBee Cluster Library (ZCL) commands are at your disposal. Use them creatively. If you like, tell me about them. I love hearing of new applications for ZigBee, and new creative ways to use it. My email address is drewg@sanjuansw.com.

> Custom Commissioning allows a wide range of commissioning options.
>
> Use Custom Commissioning in private application profiles.