Wireless Sensor Networks for Condition Based Maintenance: Security and Reliability

By
Tung Doan and Thomas Jäger

June 4, 2009

Thesis submitted in Partial Fullfillment of the Requirements for the Degree
Master of Technology in Information and Communication Technology

Faculty of Engineering and Science
University of Agder

Grimstad

Contents

List of Tables

List of Figures

Summary

In this thesis we present the requirements needed for a Condition Based Maintenance (CBM) solution by using Wireless Sensor Networks(WSN).

The sensor network needs to be secure and reliable to be implemented in a CBM solution, hence we have chosen the popular ZigBee protocol to see how it fulfill the reliability requirements of transmission delay, lost packets and battery consumption. The security requirements is based on the CIA(confidentiality, integrity, availability) triad for security objectives, and in this thesis we will give a general overview of the security mechanisms used in ZigBee to achieve these requirements. Also a brief overview of threats a WSN faces is also presented.

Practical experiments for testing delay, battery consumption and environmental interference from magnetic fields has been conducted using XBee units from Digi International. These tests shows that the transmisson delay is within acceptable levels, and the battery consumption for sensors allows them to run for several years.

Another contender for use in industrial processes is WirelessHART by HART Communication foundation. Like ZigBee, WirelessHART is based on the IEEE 802.15.4 standard for wireless personal area networks. However WirelessHART has a different specification for the data-link layer than ZigBee and offers fundamental differences like Time Division Multiple Access and Frequency Hopping Spread Spectrum while ZigBee uses Carrier Sense Multiple Access and Direct Sequence Spread Spectrum.

1 Introduction

Wireless Sensor Networks is an emerging technology with many uses within industrial automation. The networks consists of low power and low rate sensor nodes. makes them suitable for running on a battery for up to several years, and makes them suitable for many applications such as monitoring. The most common type of wireless communication technology used for sensing and monitoring is known as Low-Rate Wireless Personal Area Networks (LR-WPAN). LR-WPAN is characterized by low-cost, low power wireless devices that self-organize into a short-range wireless network that supports low powered applications as sensing and monitoring. Networks can be a simple one-hop star topologies to more complex multi-hop mesh networks.

While Wireless Sensor Networks is more cost effective compared to wired networks the use of wireless sensors has yet to have a breakthrough in the industry. The reason for this is the security and reliability challenges in these networks. In a closed wired network an outsider needs to gain physical access to the equipment or abuse a compromised machine.

Key concerns within the automation industry has been a lack of suitable standards to fulfill the demand of reliable and secure communications. Two standards that are used in the industry are ZigBee and WirelessHART. Both provides a set of protocols, services, and interfaces for vendors to create LR-WPAN hardware platforms and software applications that will enable companies to deploy Wireless Sensor networks for monitoring and control processes.

In the industry today most companies exchanges equipments or parts of an equipment based on a schedule regardless of the condition of the machine. Because of this parts that are still in working condition will be replaced even though there is no need for replacement yet. The most effective way is to replace parts based on their condition and not on a schedule. This solution is called Condition Based Maintenance (CBM). To make CBM possible there is a constant need of monitoring and wireless sensors are perfect for these purposes.

1.1 Research Question

The first question we need to answer is what features a Wireless Sensor Network (WSN) for use in Condition Based Maintenance (CBM) applications need. To ensure that a WSN is viable as a replacement for wired sensor networks there have to be a certain level of reliability and security required from the WSN. But what exactly are these requirements, below we outline some of the features we need to consider.

- Reliability
  - The transmission delay in a wireless sensor network is higher than in a wired network. This can be due to unforeseen circumstances such as interferences from external sources, objects blocking the way of the signal, etc. In an industrial environment is important that crucial data is received as fast as possible to be more effective. How important is this for CBM, which situations require very low transmission delay?
  - The networks needs features that ensure messages is safely delivered to the destination. Therefore some sort of detection is needed for lost packets.
- Security
  - Resilience against attacks. For example: DDoS, Man in the middle, Sybil attack. All these attacks may cause maintenance systems to break down and report a bad state. What kind of security is needed for CBM applications. Major threats to WSN for CBM applications needs to be identified.
  - Incorrect data may lead to severe damages to a company. We need to be able to trust that the data is correct, and not corrupt or have been tampered with. Take for example in CBM applications a critical part may not be replaced before breaking down because the data a sensor has been sending is incorrect or have been tampered with.

The next part of the research is deciding if a WSN actually meets the requirements outlined in the first part. To be able to decide this we need to get background information about protocols and standards and design tests and later implement the tests to see if the WSN meets the requirements for CBM applications.

1.2 Importance of Research

Today most industries still use wired sensors for industrial processes, and while it is effective it also may cost more than a wireless sensor network. This thesis will take in consideration the security and reliability in wireless sensors networks as an replacement for wired sensor networks for industrial purposes.

One of the main future uses for these wireless sensor networks is in Condition-based maintenance (CBM). The wireless sensors will be used to monitor the equipment condition or status. For example an electric motor may have a vibration sensor, if the vibrations detected by this sensor goes beyond certain parameters it may have to be exchanged. Instead of letting it run until it breaks down completely, halting production or be a safety risk. It also may be cheaper than exchanging a motor on a schedule (Preventive maintenance).

Laying down wires for sensors for monitoring tens or hundreds of parts in a complex system may be prohibitively expensive or impossible (Independent moving parts), here is where the wireless sensor networks come into play.

The importance of this thesis is to be able to find the most reliable(delay, robustness, timing etc) wireless sensor standard for an industrial environment. Based on our research we want to be able to conclude if WSNs are reliable and secure enough for use in CBM.

1.3 Assumptions

This thesis will rely on existing literature for many aspects. We have to assume that published literature, especially on the security aspect, are correct.

We also have to depend on protocol specifications for evaluating the ZigBee protocol, especially since some features might not be compatible with or available on the equipment used in this thesis. We will assume that specifications are correct and come with Errata if there are any issues not in the main specification.

We will also use existing papers and research for our comparison of WirelessHART and ZigBee, as we can only conduct tests on the ZigBee we have to assume that the research papers care correct and are not biased.

1.4 Goals and Limitations

This thesis is very extensive and several limitations have to be done.

The requirements for this project is as follows:

- Find the requirements for a wireless CBM system.
- Research reliability mechanisms
  - Design tests for reliability issues such as delay and battery consumption.
- Research security mechanisms.
  - Encryption
  - Authentication
- Set up a Wireless Sensor Network for CBM.
- Research differences between WirelessHART and ZigBee.

Below we outline some limitations:

- We will only implement a small number of tests in the network.
- Research security threats, but implementation of these will not be conducted.

2 Technology overview

2.1 IEEE 802.15.4

The IEEE 802.15.4 standard defines the protocol and interconnection of communication devices using low-data-rate, low power, and low complexity short-range radio frequency (RF) in a wireless personal area network (WPAN). The standard describes the physical layer (PHY) and the media access control (MAC).[4]

The standard support star as well as peer-to-peer topology and carrier sense multiple access with collision avoidance (CSMA-CA) media access mechanisms. The media access is contention based, but by using the optional superframe structure, time slots can be allocated by the network coordinator to devices with critical data. The network coordinator can also provide connectivity to higher performance networks.

2.1.1 Physical layer

The PHY includes two services: the PHY data service and the PHY management service. The PHY management service interfaces to the physical layer management entity (PLME) service access point (SAP) (PLME-SAP). The PHY data service transmits and receives PHY protocol data units (PPDU) across the physical radio channel.

Features of the PHY includes:

- Activation and deactivation of the radio transceiver
- Energy detection(ED)
- Link quality indication(LQI)
- Channel selection
- Clear channel assessment(CCA)
- Transmitting and receiving packets across the physical medium
- The optional UWB PHY has the feature of precision ranging.

IEEE 802.15.4 includes six alternative PHYs. The PHYs are as follows:

- An 868/915 MHz direct sequence spread spectrum (DSSS) using binary phase-shift keying (BPSK) modulation, and support data rates of 20kb/s, 40kb/s, and optionally 100kb/s and 250kb/s

- An 868/915 MHz DSSS PHY using offset quadrature phase-shift keying (O-QPSK) modulation, and support data rates of 20kb/s, 40kb/s, and optionally 100kb/s and 250kb/s
- An 868/915 MHz parallel sequence spread spectrum (PSSS) PHY using BPSK and amplitude shift keying (ASK) modulation, and support data rates of 20kb/s, 40kb/s, and optionally 100kb/s and 250kb/s
- A 2450 MHz DSSS PHY using O-QPSK modulation, and support data rate of 250kb/s
- A 2450 MHz Chirp spread spectrum (CSS) PHY that supports date rates of 1000kb/s and optionally 250kb/s
- Ultra-wide band (UWB) PHY at frequencies of 3 GHz to 5 GHz, 6 GHz to 10 GHz, and less than 1 GHz. Support data rates of 851 kb/s with optional data rates of 110kb/s, 6.81 Mb/s, and 27.24 Mb/s

2.1.2 Media access control

The MAC sublayer handles access to the physical radio channel and is responsible for the following tasks:

- Generating network beacons for the coordinator
- Synchronization of network beacons
- Support of PAN association and disassociation
- Device security
- Employing CSMA-CA for channel access. ALOHA is used for UWB PHY.
- Handing and maintaining the Guaranteed time slot (GTS) mechanism
- Providing reliable links between two MAC entities.

2.2 ZigBee

The ZigBee protocol was developed by the ZigBee Alliance and is based on the IEEE802.15.4 standard for low-cost, low-power, wireless communications. [13]

The ZigBee stack architecture contains a set of blocks called layers. Every layer performs a specific set of services for the layer above. A data entity provides services for data transmissions and a management entity provides services

for all other types of services. Each entity exposes an interface to the upper layer through a SAP, and each SAP has a number of service primitives to accomplish the required functions. The IEEE 802.15.4 standard defines the physical layer and the media access control. The ZigBee protocol defines the network layer (NWK) and the framework for the application layer.

2.2.1 Network layer

The network layer provides functionality to ensure correct operation of the IEEE 802.15.4 MAC sub-layer and also provides a service interface to application layer. The network layer includes two service entities; the network layer data entity (NLDE) and network layer management entity (NLME).

Network Layer Data Entity The NLDE provides data services to applications to be able to transport application protocol data units (APDU) between two or more devices. The devices themselves has to be on the same network.

The NLDE provides the following services:

- *Generation of the Network level PDU (NPDU)*
  The NPDU is generated from an application support sub-layer PDU through the addition of a protocol header.

- *Topology-specific routing*
  The NDLE is responsible for transmitting an NPDU to the device that is the final destination or the next step in route towards the final destination.

- *Security*
  The NLDE ensures that both the authencity and confidentiality of a message or transmission.

Network Layer Management Entity The NLME provides management services to applications to be able to interact with the stack.

The NLME provides the following services:

- *Configuring new device*
  The NLME is responsible for configuring the stack for required operations. Configuration settings include beginning an operation as ZigBee coordinator or joining an existing network.

- *Starting a network*
  The ability to start a new network.

- *Joining and leaving a network*
  To be able to join, rejoining or leave a network. ZigBee coordinator or ZigBee router can also request that a device leave the network.

- *Addressing*
  So the ZigBee coordinators and routers can be able to assign addresses to devices that is joining the network.

- *Neighbour discovery*
  The ability to discover, record and report information regarding one-hop neighbour devices.

- *Route Discovery*
  To be able to discover and record routes through the network, to see if messages are efficiently routed.

- *Reception control*
  Allows a device to control when a receiver is active and for how long, enabling MAC sub-layer synchronization or direct reception.

- *Routing*
  To be able to use different routing mechanisms as unicast, broadcast, multicast or many to one to exchange data in the network.

2.2.2 Application layer

The application layer consists of the application support sub-layer (APS) and the ZigBee device objects (ZDO).

Application support Sub-Layer

The applications support sub-layer provides an interface between the network layer and the application layer through a set of services that are also used by the ZDO and manufacturer-defined application objects. The services are provided by two entities; the APS data entity (APSDE) and the APS management entity (APSME).

Application Support Sub-Layer data Entity

The APSDE provides data service to the network layer, ZDO and application objects to be able to transport application PDYS between two or more devices. The APSDE provides the data transmission service via its associated SAP, the APSDE-SAP.

The APSDE provides the following services:

- *Generates application level PDU (APDU)*
  The APSDE takes an application PDU and generates and APS PDU by adding the appropriate protocol overhead.
- *Binding*
  When two devices are bound, the APSDE is able to transfer messages from one device to the other device.
- *Group address filtering*
  Provides a function to filter group-based messages based on endpoint group membership.
- *Reliable transport*
  Improves the reliability of transmissions of that offered by the NWK layer alone by using end-to-end retries.
- *Duplicate rejection*
  Messages that are transmit will no be able to received more than once.
- *Fragmentation*
  Enables segmentation and reassebly for messages longer than the payload of a single NWK layer frame.

Application Support Sub-Layer Management Entity

The APSME provides management services to applications to be able to interact with the stack. The APSME provides the management service via its associated SAP, the APSME-SAP. The APSME provides the binding service. The matching of two devides are based on their services and needs, and the APSME is able to construct and maintain a table to store this information. It also maintains a database of managed objects, called the APS information base (AIB).

The APSME also provides the following services:

- *Binding management*
  Match two devices togehther based on their services and needs.
- *AIB management*
  To be able to get and set attributes in a device's AIB.
- *Security*
  Set up authentic relationships between devices by using secure keys.
- *Group management*
  Declares a single address used by several devices, to add devices to the group and also remove devices from the group.

ZigBee Device Objects

The ZDO provides an interface between the application objects, the device profile and the APS. ZigBee Device Objects are applications that uses network and application support layer primitives to implement ZigBee End Devices, ZigBee Routers and ZigBee Coordinators.

The ZDO is provides the following:

- Initializing the application support sub-layer, the the network layer and the Security Service Provider.
- Assembling configuration information from end applications to determine and implement discovery, security management, network management and binding management.
-

2.3 XBee

In this master thesis experiments have been conducted on XBee modules from Digi International Inc. The XBee ZB RF modules are engineered to work with the ZigBee protocol and supports the needs of low-power wireless sensor networks. The modules operate within the ISM 2.4 GHz frequency band.[2]

There is a different version of the XBee ZB RF modules, there is a Pro version with higher ouput power, 60mW (50mW for European markets). All tests in this thesis is conducted wit the standard version with 2mW output power.

Key features of of the XBee XB RF modules are:

Figure 1: Three XBee ZB 2.5 Series 2mW devices. Two in USB explorer converters.

- Indoor range up to 40 m
- Outdoor line-of sight range up to 120 m
- Transmit Power Output: 2 mW (3 dBm)
- Receiver sesitivity: -96 dBm
- Data rate 250 kb/s

2.4 XBee Power supply

Before explaining our methodology for testing power usage of the XBee devices, we need to explain some things about batteries and power supplies that are suitable. The XBee requires between 2.8 volt and 3.4 volt to operate.[2] This means it can be run directly from two 1,5 volt primary (non-rechargeable) battery cells.

But some of the most common types of rechargeable battery cells has some problems for use with the XBee as explained below.

2.4.1 NiMH (nickel-metal hydride cell)

NiMH cells has a voltage of 1.2 volt, two cells in series produces 2.4 volts so two cells are not enough for the XBee. Three NiMH cells produce 3.6 volt which is too much for the XBee, so with NiMH cells a voltage regulator is needed.

NiMH cells has a self-discharge rate of 30% per month.

2.4.2 Li-ion

Li-ion battery cells has higher energy density then NiMH cells, and a very low self-discharge rate (5-10% per month). The voltage of Li-ion cells depends on the material used for the cathode in the cell.[7] There are 4 common cathode materials as detailed in Table 1. For the XBee the $LiFePO_2$ cell is the best suited as its typical voltage is right in the range of the supply voltage of the XBee.

Li-ion batteries have some issues when using this cell for XBee. The output voltage of Li-ion cells vary over the discharge time.[7] From very high right after full-charge, to below the typical voltage. The high voltage right after a charge is a problem because it can damage the delicate circuitry of the XBee so a voltage regulator is required for this type of battery as well.

Cathode Material	Average Cell Voltage
$LiCoO_2$	3.7 V
$LiMnO_2$	4.0 V
$LiFePO_2$	3.3 V
Li_2FePO_4F	3.6 V

Table 1: Li-ion cathode materials

2.4.3 Voltage Regulators

In the previous sections we mentioned the need of voltage regulators, in this section we explain the different types and their operations.

Linear Voltage Regulators Linear voltage regulators are electronic solid-state devices that step down DC voltage to a fixed voltage.[1, 7] In Figure 2 is a picture of a common linear regulators the LD33V in a TO-220 package. This regulator gives 3.3V output and is a so called a fixed Linear Regulator.[10]

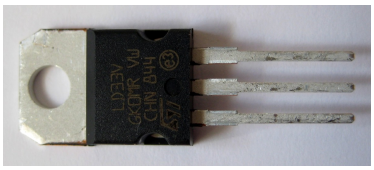

Figure 2: 3.3 volt fixed Linear Voltage Regulator LD33V

Linear regulators work by converting the excess voltage to heat. The input current is the same as the output current,[1, 7] so if you have a 9 volt input to a 3.3 volt fixed regulator the extra input power is dissipated as heat in the heat sink. The input voltage to the LD33V regulator in figure 2 has to be at least 1 volt higher then the output voltage (Drop-out voltage)[10]. Other regulators will have a different drop-out voltage the common drop-out range for normal regulators is 1 volt to 2 volt.

If you have a 7V input voltage to a 5V regulator it's going to be 71% efficient, with a 9V input it's 56% efficient, as much of the power goes to heat.

A more specific example: You have a device that draws 100mA at 5V. The input to the linear regulator is 9V. The input to the regulator is 9V * 100mA = 0.9 watts, but your device only uses 5V * 100mA = 0.5watts. 0.4watts is lost to heat.

Quiescent Current If your device can sleep or uses very little power what will matter most is the quiescent current. That is the current running through the voltage regulator even if there is no load on the regulator drawing power. It's also called ground current. The most common regulators have a quiescent power of between 3mA and 10mA.[7] The quiescent current of the LD33V regulator is typically 5mA.[10] This means 0.046 watts are lost in the 9V to 5V regulation even if you device is sleeping and not drawing current.

A standard 9V battery has a capacity of 625mAh. When regulating 9V to 5V a 9V battery would be drained after just over 120 hours or 5 days, without any load and just the quiescent current draining the battery.

If the device can sleep or uses very little power one need to consider how much the power supply is leaking current and find a power supply option with low quiescent power.

Switching Voltage Regulators Switching regulators are usually more efficient then linear regulators. A switching regulator works by switching the input current on and off, therefore reducing the output current. A feedback loop tells the regulator how much the regulator should turn the current on or off.[7]

The output voltage can be unstable (Voltage Jitter) and could potentially skew things that need a stable voltage supply. (For example reading a ADC input). A output capacitor should reduce the jitter in the current.

The data sheet states that the Quiescent Power is typically 5mA. This means that if the device uses very little power or has a sleep mode (Like most micro controllers) and runs off a battery, the switching regulator will drain your battery about as fast linear regulators.

Ultra low Quiescent power Linear Regulators The alternative for low power battery applications is devices such as MAX883 / MAX882 Linear Regulators from Maxim. These devices work the same way as standard Linear Regulators, but are optimized for low Quiescent Power. These devices has a Quiescent Power of 11µA or 0.011mA.[7] Or 1/500 less than the standard linear voltage regulators and switching regulators. This means one can let the micro controller sleep without the Linear Regulator draining the power out of the battery.

3 Requirements of CBM

Scenario A: The sensor wakes up and do tests every 15 minutes, if the readings are in a optimal parameters it goes back to sleep. The sensor only sends a message if the readings are in critical condition or the battery is getting low, the frequency of readings can be changed to suit different scenarios.

Scenario B: The sensor wakes up in a set interval and takes tests and sends the information to a base station. The sensor does not decide the equipments condition, but an external device does.

3.1 Reliability requirements

3.1.1 Transmissions

In our scenario the sensor only sends a message in two cases when the equipments that the sensor is monitoring is in a critical condition or when the battery is low. In a CBM solution the urgency of these messages can differ from task to task. For example in a case of a furnace, where the sensor sends an alarm when the the furnace is overheated, and its only a matter of minutes before the furnace is damaged permanently. In this case the message have a high level of urgency and it's important that the packet sent has a low transmission delay or doesn't go missing. The requirements is that the network has a low transmission delay, and have countermeasures against lost packets.

In other cases where the sensor readings are less important a long delay may not be an issue.

3.1.2 Range

The range of devices can vary from equipment to equipment. The topology of the network can depend on the range, if all sensors are in range of the base station it is not necessary with a peer-to-peer topology such as wireless mesh networks, a simple star topology might be enough. While range is not as important as other aspects, we still need to see how we can overcome the shortcomings of a low-range network.

3.1.3 Battery

An important requirement for wireless sensors are the lifetime of the battery. These sensors may be placed in places hard to reach, and is time consuming to replace.

List of factors that can deplete battery:

- *Sensor readings:* The sensors are asleep until it's time to take a reading. The frequency of readings will affect battery life.
- *Encryption:* Encryption requires more computing power and might affect battery life.

3.2 Security requirements

Three concepts forms what is often referred to the CIA(confidentiality, integrity, availability) triad.[9] These three concepts embodies the fundamental security objectives for both data and information and computing services. In this section we will describe these concepts and if these concepts are required in a CBM solution.

The concepts and the loss of security objectives are explained below.

3.2.1 Confidentiality

Preserving authorized restrictions on access and disclosure of information. This term covers two related concepts:

- *Data confidentiality:* Assures that confidential information is not disclosed to unauthorized individuals.
- *Privacy:* Assures that individuals has control or influence over information that is related to them may be collected and stored and who can gain these information.

A loss of confidentiality is the disclosure of unauthorized information. Only Data confidentiality is a concern when it comes to CBM. Privacy only concerns information of individuals.

In a CBM solution confidentiality might not be as important depending on the information sent by the end devices. In Scenario A the messages only contain the condition of a machine might not need to be confidential.

In Scenario B the messages are sent more often and contains information about the machines that need to be confidential.

3.2.2 Integrity

Guarding against improper modification or destruction of information. This term covers three related concepts:

- *Data integrity*: Assures that information and programs are changed in a authorized manner.
- *System integrity*: Assures that a system performs the intended function in a safe manner, free from unauthorized manipulation of the system.
- *Authenticity*: The property of being genuine and being able to verified and trusted. The validity of a transmission, message or message originator needs to be trusted. This means to be able to verify that the the message comes from a trusted source.

The loss of integrity is the unauthorized modification or destruction of information.

In a CBM solution integrity is an important requirement. If the integrity is compromised the system might change or shut down leading to company losses.

3.2.3 Availability

Ensuring timely and reliable access to information and use of information. The loss of availability is the loss of access to information or loss of use of information in a system.

Availability is important in a CBM solution. If you lose availability the system might be disrupted and users have no information about the state of the equipments, this might lead to shut down of industrial processes leading to lost productivity.

3.3 Threat Assessment

The threats against a CBM system are many and often very complex problems. Here we are going to present some of the security threats against a wireless CBM system.

3.3.1 Confidentiality threats

In a wireless CBM system it's much easier to get access to the data, since it's not necessary with physical access. An attacker can passively monitor the network to gather information if it's not encrypted. The attacker just need to be within range of of the wireless network. The range of Wireless networks varies, but in a wireless sensor network one should expect a range of about 300 meters and above. This means that an attacker just needs to be within 300m of the network to monitor the information.

Even if the network is encrypted an attacker can do traffic analysis [12], to get information about communication patterns and sensor activities. This information about the traffic patterns can be used to identify important nodes for further attacks.

3.3.2 Integrity threats

In a CBM system it is important that sensor readings are not corrupted in transport to the central. If the data is corrupted and central gets the wrong data parts may be exchanged before it's really necessary thus incurring a loss. Also consider that if the central gets erroneous data there may be automatics in the central that triggers and shuts down the process or gives the wrong instructions to operators.

The main threats against wireless CBM systems is that someone intentionally changes the sensor readings to confuse maintenance systems. A common attack against networks is replay attacks where an attacker captures one packet from the network and then resend it to try and confuse the network and cause disruptions. In the case of a replay attack, encryption wont necessarily help since the attacker just captures the encrypted data and re-sends it. Consider an encrypted wireless sensor network, an attacker wishing to disrupt the readings could capture a packet then constantly resend it to the network confusing the node that is interpreting the sensor data.

Another possible attack against wireless networks is spoofing attacks. In

this type of situation an attacker forges its identity and masquerades as another device or in some cases as many devices. This forged identity/device could in theory fool other nodes on the network to communicating with the attackers forged device instead of a real device. A spoofing attack could harm the integrity of the sensor data, but also opens the network to many other attacks. A special type of spoofing attack is the Sybil attack, in this type of attack a malicious node creates multiple forged identities in the sensor network by fabricating or stealing identities of legitimate nodes. Sybil attacks can be used against routing algorithms and topology maintenance. [12] It can also reduce the effectiveness of fault tolerant systems.

3.3.3 Availability threats

The threats against availability in the case of a wireless CBM system is intentional disruption, or accidental disruption. Intentional disruption is when an individual or group intentionally tries to interrupt the communication between nodes in the CBM system so that sensor data does not get trough to the control room or the operators responsible. There are many ways to cause a disruptions in a wireless sensor network, but the most significant is jamming. Jamming is done by using a powerful radio transmitter that operates in the same frequency as the wireless sensor network to try and disrupt the communications between the nodes in the sensor network. In most cases jamming is only effective if the signal between sensor nodes are already weak and the jammer device is close to the network.

Routing loop attacks target the information exchange between nodes.[3] False network messages are generated to trick the other nodes in the network to route information wrong causing traffic increase and delays.

Accidental disruption is when an accident or unforeseen action causes the sensor network to fail. There can be many reasons for accidental disruptions. For example if a node in the network malfunctions, if this node is a cluster leader or router, large parts of the network can become unavailable unless an alternative route can be used.

3.3.4 Physical threats

This class of threats are easy to understand but can also be hard to protect against. These threats are also valid for wired CBM systems. If an attacker has access to the physical sensors they may change the sensors input and cause

disruption. On wired CBM systems they can hook into the wires carrying the sensor data and read it and change it. Modbus the most used serial protocol for industrial automation and monitoring only has very basic readability checks on data, 16bit CRC.[8] An determined attacker with physical access to the wires carrying Modbus protocol data could easily intercept signals, change the data, resend it. It's impossible for the receiver of the packet to know if the data has been intercepted and changed because the protocol doesn't use any kind of Encryption or Message Authentication Codes. If the attacker can subvert or steal a node it may reveal cryptographic keys thus compromising the rest of the sensor network.

In any CBM system good physical security is essential.

3.4 Attacks on sensor networks

We have already described the requirements of a CBM network when it comes to confidentiality, integrity, availability and threats to these requirements. In this section we will describe attacks that aims for these security requirements.[12]

1. *Passive Information Gathering*
   An outsider with enough resources can use a packet sniffer to gather data and collect information from a sensor network.

2. *Node subversion*
   Capture of a node to reveal its information. Usually to gain knowledge of the cryptographic keys used in the sensor network.

3. *False Node*
   An outsider can gather data by adding a malicious node to the network. This node can then send malicious data to other nodes, and lure other nodes to send data to it.

4. *Node Malfunction*
   A malfunctioning node can cause inaccurate data that would compromise the integrity of the network, especially if the node is a network coordinator.

5. *Node Outage*
   When a node stops operating. This could lead to broken link from source device to destination device.

6. *Message Corruption*
   When the contents of a message is modified by an outsider.

7. *Traffic Analysis*
   Even when a a message is encrypted, an outsider can do analysis of the communication patterns and sensor activities to cause harm to the network.

8. *Routing loops*
   In sensor networks routing loops attacks target the information exchanged between two nodes. False error messages are generated when an outsider alters and replays routing information.

9. *Selective forwarding*
   Selective forwarding is a method to influence the network traffic by believing that all the participating nodes in the network are reliable to forward the message. In a selective forwarding attack a malicious nodes simply drop certain messages instead of forwarding every message. When a malicious node receives a message it deceives the sender by reducing the latency to make it believe it's on a shorter route. The closer the malicious nodes is to the base station the more traffic it will attract. When it drops more messages and forwards less, it retains its energy level thus remaining powerful enough to trick neighboring nodes.

10. *Sinkhole attacks*
    An attack that attracts the traffic to a compromised node. The simplest way is to place a malicious node where it can attracts most of the traffic, possible close to a base station, or the node can imitate the base station. One of the reasons for this attack is to make selective forwarding possible to attract traffic to a compromised node.

11. *Sybil attacks*
    A type of attack where a node creates multiple illegitimate identities in a sensor network by fabricating or stealing identities from legitimate nodes. Sybil attacks can be used to target routing algorithms and topology maintenance. It also reduces effectiveness of fault tolerant schemes such a distributed storage and dispersity.

12. *Wormhole attacks*
    In these attacks two nodes create a link over a low latency link (Ethernet cable, optical cable or long range wireless transmission). These two nodes convince the neighboring nodes they are closer to the base station and will attract more traffic.

13. *Hello flood attacks*

    By broadcasting a message with stronger transmission power and pretending the source of the HELLO message is from the base station. Nodes that receive this message assume that node sending the HELLO message is closest and they try to send all their messages through this node. All nodes will be responding to HELLO floods and waste energy. The real base station will also be broadcasting, but will only have a few nodes responding to it.

14. *DoS attacks*

    Denial of service attacks occur at a physical level causing radio jamming, interference with the network protocol, battery exhaustion etc.

4 Reliability

4.1 Data Transmissions

ZigBee data packets can be sent as either unicast or broadcasts transmissions. Unicast transmissions route data from one source device to a destination device. Broadcast missions are sent to all the the devices in the network.

Data transmissions sent to a end device is treated differently because the end device might me in sleep mode. Hence the retransmission timeout is longer for messages meant for end devices.

4.1.1 Broadcast Transmissions

Broadcast transmissions are intended to be propagated through the entire network such that all nodes receive the transmissions. To achieve this, all devices that receive a broadcast transmissions will retransmit the packet three times.

Every nodes in the that transmits the broadcast will also make an entry in a local broadcast transmission table. This table is used to keep track of each received broadcast packet to ensure the packets are not endlessly transmitted.

4.1.2 Unicast Transmissions

Unicasts transmissions are sent from a source device to a destination device. The destination device can be an immediate neighbor or several hops away from the source device.

Every ZigBee device in the network has a 16-bit network address and a 64-bit unique address assigned during manufacturing. Unicasts transmissions are always routed to the 16-bit address of the destination. To ensure that data is received by the correct device, the destination 64-bit address is often included in the RF transmissions. Network address is not necessary static, so if a receiving device has a matching 16-bit address of the incoming packet, but not a matching 64-bit address, it will discard the packet and obtain a new 16-bit address.

XBee ZB firmware requires that data be sent to the destination devices 64-bit address. However, since the actual RF transmission requires 16-bit addressing, the 16-bit address will be discovered by the XBee if not known.

4.2 Acknowledgment

In a CBM solution we have to ensure certain frames is received, to guarantee this we can set the acknowledgment request sub-field to 1. When the destination device determines the frame is valid it will generate and send an acknowledgment frame to the originator of the frame. If the acknowledgment frame is not received, the source device will re-transmit the data.

4.3 Retransmissions

A frame with its acknowledgment request sub-field set to 0 will not be acknowledged, the device will assume that the transmission was successful.

If the acknowledgment request sub-field is set to 1, the device will wait a certain amount of time for the acknowledgment frame to be received. If an acknowledgment frame is received within the a set duration containing the same cluster identifier and APS counters and has the source endpoint equal to the destination endpoint of the original frame, the transmission is seen an successful. If an acknowledgment frame is not received within the duration, or an acknowledgment is received but contains a wrong cluster identifier or APS counter or has a source endpoint that is not equal to the destination endpoint of the original frame, the device will assume that the transmission has failed.

When a transmission fails, the device will repeat the process of transmitting the frame and waiting for acknowledgment, up to a set maximum retries. If an acknowledgment is not received after the maximum retries, the APS sub-layer shall assume the the transmission failed and notify the higher layer of the failure.

Retransmission of a secure frame shall use the same frame counter as the original frame.

4.4 Modes of Operation

This subsection is written by using the XBee manual as a reference.[2] Some of these functions might not be compatible with other ZigBee devices.

4.4.1 Idle Mode

When not receiving or transmitting data, the RF module is in Idle Mode. The device switch into other modes of operation under the following conditions. During idle mode the device will frequently check the radio channel for incoming data.

- Transmit Mode: Serial data in the serial receive buffer is ready to be sent.
- Receive mode: Valid RF data is received through the antenna.
- Sleep Mode: End device only. Used to lessen power consumption.
- Command Mode: Command Mode sequence is issued.

4.4.2 Transmit Mode

When serial data is received and ready to be sent, the device will exit Idle Mode and attempt to transmit the data. The destination address determines which node will receive the data.

Before the device attempt to transmit the data it ensures that a 16-bit network address and a route to the destination have been established. If the destination address is unknown, network address discovery will start. If a path is not known, path discovery will take place to establish a path to the destination device. The packet will be discarded if the a matching address is not discovered. If path discovery fails to establish a path to the destination device the packet will also be discarded. If acknowledgment request is set the device will wait for an acknowledgment frame before entering Idle Mode again.

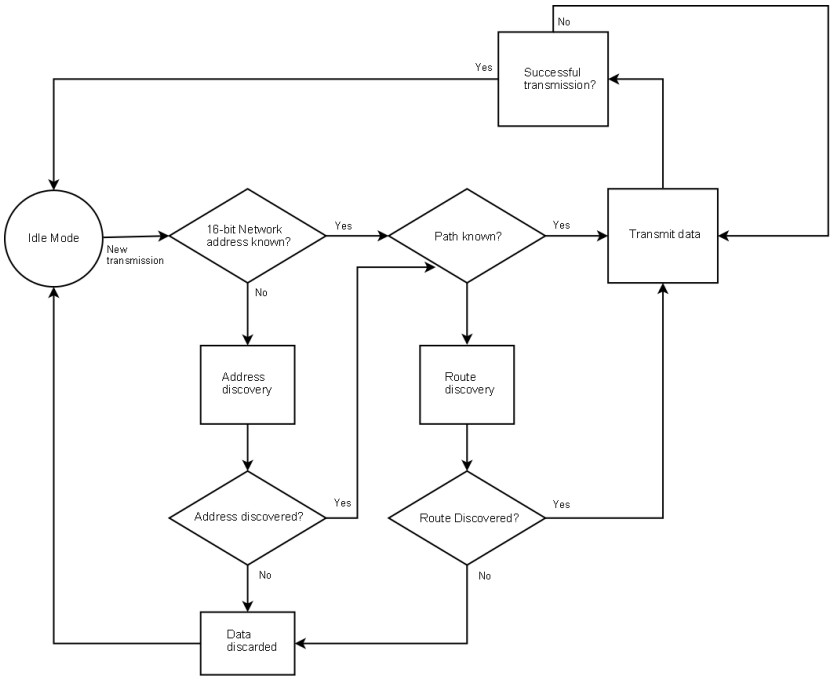

Figure 3: Transmit mode sequence

4.4.3 Receive Mode

A device goes from Idle mode to Receive Mode when it detects valid data through the antenna. The data is transferred to the serial transmit buffer.

4.4.4 Command mode

To read or modify RF modules parameters, the device must first enter Command Mode. Command Mode is a state where incoming characters are interpreted as commands.

4.4.5 Sleep Mode

Sleep mode is only supported by end devices. Sleep mode is important for devices running on batteries to lower power consumption. ZigBee routers and Coordinators can not use sleep mode.

XBee end devices support two different sleep modes:

- Pin Sleep

- Cyclic Sleep

Pin sleep lets an external micro controller to determine when the XBee unit should sleep and when it should wake by controlling the Sleep_RQ pin. Cyclic sleep allows the sleep period and wake times to be configured through AT commands.

During sleep mode XBee devices poll their parent (by default 100ms) to check if there is any buffered data.

5 ZigBee Security

5.1 ZigBee security architecture

The ZigBee security architechture includes security mechanisms in the NWK and APS layer of the protocol stack. Each layer is responsible for the secure transport of their respective frames. The APS sublayer is responsible for the establishment and maintenance for security relationships. The ZigBee Device Object (ZDO) manages the security policies and the security configuration for the device.

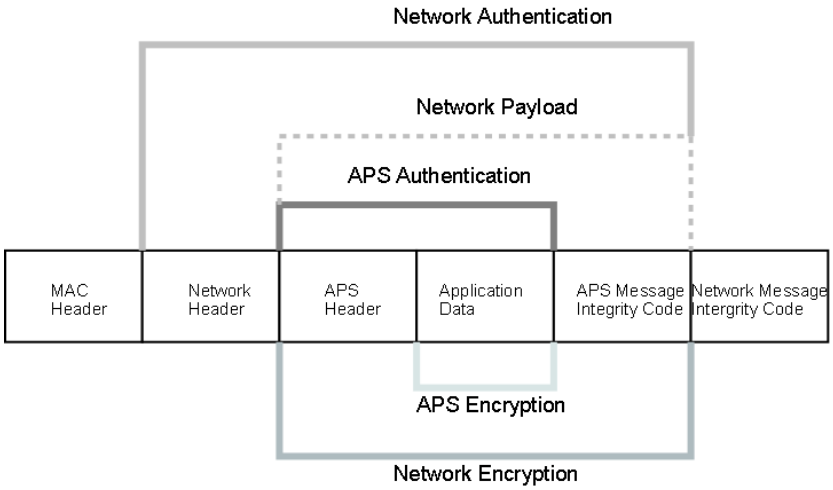

Figure 4: Authentication and encryption with Network and APS layer security enabled

5.2 Encryption

ZigBee uses the Advanced Encryption Standard (AES) for encryption. The AES standard is adopted by the United States government and is one of the most popular algorithms used for symmetric key cryptography. AES was developed to succeed the Data Encryption Standard (DES).

The AES encryption standard AES uses a block length of 128 bits and a key length that can be 128, 192 or 256 bits. The ZigBee standard uses a key length of 128 bits. [9]

The ZigBee standard uses AES with Counter with CBC-MAC (Cipher Block Chaining-Message Authentication Code) (CCM). As the name implies CCM-mode combines the well-known counter mode with the CBC-MAC. Counter is used to protech against replay attacks. ZigBee uses a variation of CCM called CCM* that allows the option of encryption and intergrity only features to CCM.

5.2.1 Network Layer Security

Network layer security can be applied to all data transmissions and is de-crypted and re-encrypted by a hop-by-hop basis. The network key is used to encrypt the APS layer and application data. In addition network encryption can be applied to route request and reply messages, APS commands, and ZDO commands. Network encryption is not applied to MAC layer transmissions such as beacon transmissions, etc.

When a device receive a message with network encryption, it decrypts the massage and authenticate the message. If the message is not the destination, it then encrypts and authenticates the message, using its own frame counter and source address in the network header.

Since network encryption is performed each hop, the packet delay is slightly longer compared to an un-encrypted network.

5.2.2 APS Layer Security

APS encryption is used to encrypt application data using a key that is shared between source and destination devices. Where network layer security is applied to all data transmissions and is decrypted and re-encrypted by a hop-by-hop basis, APS security is optional and provides end-to-end encryption using a link key only known to source and destination devices.

Packets with APS layer encryption appends a 4-byte Message Intergrity Code, the maximun data payload is reduced by 4 bytes when APS encryption is applied.

5.3 Message Integrity

The network header, APS header and application data are all authenticated with AES-128. A hash is performed on all of these fields and is appended as a 4-byte message intergrity code (MIC) to the end of the packet. The MIC allows the destination devices to check and ensure that the message has not

been corrupted or tampered with. If a device receives a packet with a MIC that does not match the device's own hash, the packet is dropped. The MIC can be set to 32, 64, or 128-bit. The bitlength of the MIC determines the probability that a random guess of the MIC would be corrects. XBee however only supports a 32-bit MIC length.[2]

5.4 Address filtering

This is a low-level security mechanism that is specified in the IEEE 802.15.4 standard and is called Access Control List (ACL) mode.[4]By using ACL all nodes within the network will only accept MAC frames from authorized nodes listed in the ACL of the device.

5.5 Security keys

Security in a ZigBee network are based on Link keys and Network keys. End-to-end communication between two devices is encrypted using a 128-bit Link key. Network keys is known by the entire network and is used for broadcast transmissons and secured by a 128-bit network key. The intended recipient always know if a frame is protected by a network key or a link key.

Devices acquires link keys either through key transport, key-establishment, or pre-installation (factory installation or a physical interface). A network key is acquired through key-transport or pre-installation. A master key is acquired by key-transport or pre-installation. The key-establishment for link key are based on the master key.

5.6 Key establishment

The APS sublayer includes the key-establishment services used to provide a ZigBee device with the mechanism to acquire the secret link key with another ZigBee device. Key establishment involved two entities, the initiatior device and the responder device, and is prefaced by a trust-provisioning step.

Trust information like a master key provides a starting point for establishing a link key. The link key can be provisioned in-band or out-of-band. The key-establishement protocol involves three steps:

1. Exchange of ephemeral data.
2. The use of this data to derive the link key.

3. Confirmation that the link key was correctly computed.

The Symmetric-Key Key Establishment (SKKE) protocol uses the the master key for the iniatior device to establish a link key with the responder device. The master key can be can be pre-installed, installed by a Trust Center (for example, the initinator, the responder, or a device acting as Trust Center), or it can be based on used-entered data (PIN, password, or key).

5.7 Key transport

The network is dependent on reliable transfer of security keys. When a device that does not have an initial key pre-loaded it must receive the first key over-the-air leading to a brief moment of vulnerability. However the transportation of the key can be secured by non-cryptographic means, for example by communicating the through an out-of-band channel. After receving the master key all other communication can be encrypted to ensure secrecy.

5.8 Trust Center

The Trust Center is the device trusted by the devices within a network. The Trust Center is responsible for the distribution of the security keys in the network. All members within a network shall recognize only one Trust Center, and there shall only be one Trust Center in each secure network.

The Trust Center address can be pre-loaded along with the initial master/network key. If not pre-loaded the Trust Center defaults to the ZigBee coordinator or a device designated the by the ZigBee coordinator.

Devices within a ZigBee network only accepts initial master, network and updated network keys only from its Trust Center. Master and link keys used for establishing end-to-end security between two devices is only accepted from its Trust Center. After the firsti nitial master key or network key, additional link, master, and network keys are generally only accepted if the come from a device's Trust Center through secured key transport.

The Trust Center shall be configured to operate in either standard or high security mode and can be used to help establish end-to-end application keys by sending out link keys directly or by sending master keys. These keys shall be generated at random.

5.8.1 High Security Mode

The high security mode of the Trust Center is designed for high security commercial applications. The Trust Center maintains a list of devices, master keys, link keys and network keys that it needs to control and enforce the policies of network key updates and network admittance. In high security mode, the memory required for the Trust Center grows with the number of devices in the network. In this mode the Trust Center also mandates the implementation of key establishment using SKKE and entity authentication.

5.8.2 Standard Security Mode

The standard security mode is designed for lower-security residential applications. The Trust Center maintains a list of devices, master keys, link keys and network keys with all the devices within the network, however it maintains a standard network key and control policies for network admittance. In lower security mode the memory required for the Trust Center doesn't grow with the number of devices in the network.

The XBee units used in this thesis only supports Standard Security mode.

6 Comparison of WirelessHART and ZigBee

Although having existed since late 2004, ZigBee has yet to to prove its success in the industrial evironment. In this section we'll give you a short introduction to WirelessHART and compare it to ZigBee by discussing the issues ZigBee has been criticized for and look at how WirelessHART addresses these issues. [5]

6.1 WirelessHART

WirelessHART by HART Communication is based on the PHY layer in the IEEE 802.15.4 standard. The WirelessHART protocol specifies new Data-ling (including MAC), Network, Transport and Application layers.

WirelessHART is designed based on a set of fundamental requirements: it must be simple (easy to use and deploy), self organizing and self-healing, flexible (support for different applications), scalable (fit for both small and big networks), reliable, secure and support existing HART technology.

WirelessHART is a Time Division Multiple Access (TDMA) based network. In a TDMA based network all devices have to be time synchronized and communication is based on pre-scheduled time slots.

WirelessHART has several mechanisms to avoid interference: Frequency Hopping Spread Spectrum (FHSS) allows WirelessHART to hop across 16 channels defined in the IEEE802.15.4 standard. Clear Channel Assessment is an optional feature that is can be performed before transmitting a message, the transmission power is configurable, and a Blacklist feature that disallows certain channels.

6.2 Comparison

WirelessHART and ZigBee is based on the OSI Seven Layer Model. While ZigBee is built on top of the Data Link and Physical layers specified in IEEE 802.15.4 adding new Application and Network layers, WirelessHART replaces the Data Link layer of 802.15.4 including a new Media Access Control (MAC).

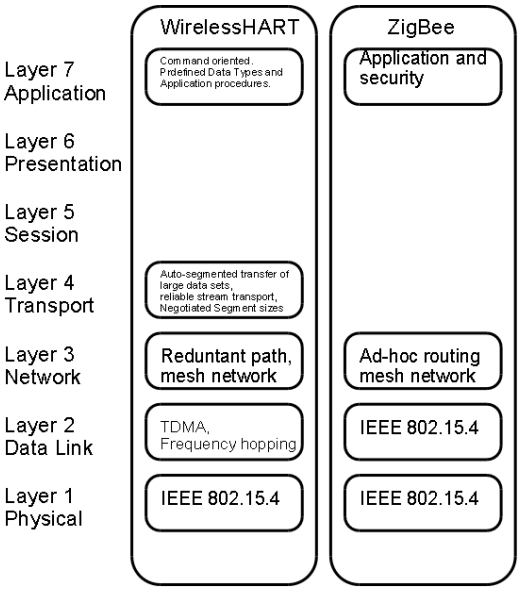

Figure 5: WirelessHART and ZigBee protocol stacks

6.2.1 Frequency agility

Frequency agility is the ability to switch frequencies of a transmitter to reduce interference from other sources. WirelessHART uses Frequency Hopping Spread Spectrum which allows it to hop between frequencies, when a frame fails to be delivered the retransmission of that frame will be retransmitted in a different frequency.

ZigBee uses Direct Sequence Spread Spectrum with only one frequency and the network shares one channel. The ZigBee network chooses the channel with the least amount of interferences at startup. Because of the lack of ZigBees frequency agility makes it highly susceptible for unintended and intended jamming.

6.2.2 Devices

ZigBee devices can be either Full-Function Devices or Reduced-Function Devices. The former can act as a router in the network and can also be the network coordinator, the latter can only communicate with known routers.

All devices in a WirelessHART network has routing capabilities and is treated equally, this makes it highly versatile. However this won't make it suitable to

run on batteries if all the devices act as routers. Fortunately WirelessHART devices can be set up with only communication functionality.

6.2.3 Path diversity

ZigBee has no path diversity meaning that if a link is broken, a new path from source destination have to be set up. ZigBee devices are responsible for route discovery, and this will delay the delivery of the frame since a new route has to discovered before retransmitting the frame.

WirelessHART uses two different mechanisms for message routing: Graph routing and source routing. Graph routing uses pre-determined paths to route messages from source to destination device. A graph route has several different paths between source and destination. If a link is broken the device will relay the message using a different path. Source routing uses ad-hoc created paths without any path diversity. Source routing is therefore only intended for network diagnostics and not process related messages.

6.2.4 Channel access methods

ZigBee uses CSMA/CA to access the radio channel. This means that if a device wishes to transmit a frame it has to listen to the channel for a certain amount of time to check if the channel is busy. If the channel is idle the transmission will start. If the channel is busy the transmission will be delayed for a random interval. Important frames can be ensured access to the radio channel by using guaranteed time slots (GTS).

WirelessHART uses TDMA for accessing the radio channel. TDMA uses pre-scheduled timeslots to avoid message collisions. This requires all devices to be synchronized, but this requires to additional network traffic because it is embedded in the automation process related traffic.

6.2.5 Security

Security in ZigBee is not mandatory. However there is support for encryption, authentication and integrity. ZigBee uses security mechanisms in 802.15.4; Counter with CBC-MAC (CCM) with AES-128 encryption.

Three keys are used in ZigBee: Master key, Link key and Network key. The Master key is used to join the network. The Link key is used for end-to-end encryption. The Network key is shared between all the devices. All keys can be

set at the factory, or be given from the trust center over the air, or through a physical interface.

Security in WirelessHART is mandatory. There is no option to turn it completely off. Counter with CBC-MAC (CCM) with AES-128 is used in WirelessHART as well.

Three keys are used in WirelessHART: Join key, Network key and Session key. The join key is similar to ZigBees Master key and is used to authenticate a device for a specific WirelessHART network. After the device has successfully joined the network the Network manager will supply it with Network and Session keys used for further communication. The actual key generation and management is handled by a Security manager, which is not specified by WirelessHART, but the keys are distributed by the Network manager. The Session key is used for authentication of end-to-end communications between two devices. For example Field device and the Gateway. Different session keys are used for each pairwise communication. (Field device and Gateway, Field device and network manager, etc). The Data Link layer uses Network key to authenticate messages on a one-hop basis.

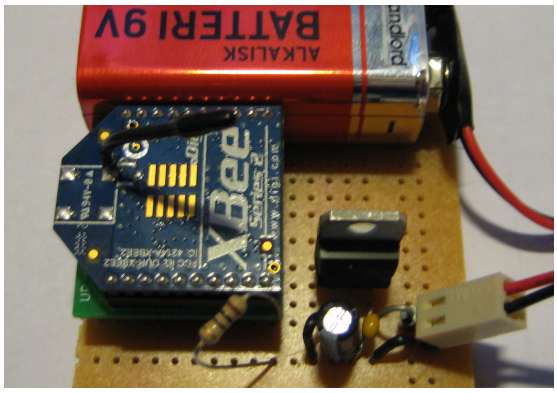

Figure 6: Standalone XBee Router with power supply circuit and battery.

7 Methodology

7.1 XBee

For all the test we use Digi XBee Series 2.5 devices with 2mW radio output and a wire antenna. There are other XBee versions that are a bit different. But the only difference is the power output. [2] The Pro version has a power output of 60mW. The 2mW version still gives us a good overview over the XBee's capabilities. As the only difference to expect from the higher radio output version is longer range.

In Figure1 we show two of the XBee devices are in USB explorer units. The USB explorer is used to connect the XBee to a computer with a USB port. The explorer works as a serial converter. We use this explorer to read and write data and API frames to and from XBee devices.

7.1.1 Standalone XBee router

For testing some of the mesh functionality we use a standalone XBee Router. Figure 6 shows a picture of the circuit with a battery attached. The schematic over this this circuit is similar to the device described in Figure 13 but without the temperature and light sensor. The circuit is basically a XBee and a power supply, powered with a battery. The XBee in this circuit is set up as a Router and is used to extend the range of the network without using a PC.

7.1.2 Setup

For all our tests with XBee devices we use the same setup. Each device is set up to use the full Zigbee stack. The name of the firmware used is XB24-ZB this firmware provides the full Zigbee protocol stack for all XBee Devices. The reasons for doing this is that we want to test how the Zigbee standard protocol works. For a Zigbee network to be functional it needs exactly one Coordinator and one or more End Devices/Routers. So at least one device in each test is set to be a coordinator.

7.1.3 Network Setup

For all tests we use mesh networking, as that configuration is the most interesting for the applications we have in mind for Wireless Sensor Networks (CBM). In CBM applications it's important that it's easy to add new devices without need for to much setup. A device should be set in the environment and find it's own route to it's destination. Mesh networking provides this.

7.2 Tools Created

For making some of the measurements we needed some new tools. Here we describe the functions of all the tools we created and used for measuring different properties of XBee devices. The tools where written in Perl. Perl is good for fast prototyping and we know the language.

7.2.1 Perl modules

Here follows a short description of the Perl modules we wrote to make tool development easier. The modules are used in the tools as a Object Oriented interface to different functions of the XBee devices and to interface to generate API frames. The modules implement most of the XBee specification for AT commands and API frames.

Xbee/AT.pm This is a modules that implements serial communications with the XBee device and implements functions for entering into AT command mode and reading/writing AT registers. AT mode is transparent meaning that all data sent to the serial port of the XBee device in AT mode is sent directly to the node specified in the Destination Node register (Usually the Coordinator). All data from the network is received trough the serial port without any added

Algorithm 1 Example usage of Xbee/AT.pm module

```
use Xbee::AT;
# Create a new Xbee::AT object connected to /dev/ttyUSB0 serial port.
my $xbee = Xbee::AT->new( {port => '/dev/ttyUSB0', speed => 115200} );
$xbee->write("data"); # Send data to the destination node.
my $dbm = $xbee->rssi; # Read the signal level register.
```

Algorithm 2 Example usage of Xbee/API.pm module

```
use Xbee::API;
my $xbee = Xbee::API->new( {port => '/dev/ttyUSB0', speed => 115200} );
# Wait for API frames and dump the received data
while ( my $hash = $xbee->read_api) {
print $hash->{data},"\n";
}
```

control structures. This makes AT mode simple to program for as one just needs to implement reading and writing to a serial port. The module enters command mode automatically if the program using the module requests reading or writing to registers.

Xbee/API.pm This module implements most of the XBee API frame specification. This makes it possible to address specific nodes in the network without setting a destination register first. The API mode is not transparent and all data sent to the the XBee in API mode needs to follow the API frame specification. API mode is the only way to get and parse join notifications and sample frames that are sent out by nodes just joining the network and nodes sending out samples. API mode is most useful on the XBee Coordinator as it's usually just this node that needs to read join notifications, sample frames and needs to address specific nodes in the network.

7.2.2 Perl Tools

Here follows a short description of the tools created for taking the actual measurements and for getting a better overview of the protocol.

api_echo.pl This Perl program uses the Xbee/API.pm module for echoing back all data it receives back to the node that sent the data. We used this program on the Coordinator node when measuring round trip time for packets traversing the network. Also records changes to the routes taken by packets

from other nodes by parsing Route Record Indicator.

ping.pl This Perl program uses Xbee/AT.pm module for sending high resolution time data to the Coordinator. After sending this time packet it waits to get the data back After getting it back it calculates the time the packed used to traverse the network using the time data in the packet and the current local time. The size of the time packet is 10 bytes and fits in one packet (ZigBee packets are up to 56 bytes).

After calculating the delay the program reads the Signal Level register. This register contains the signal level of the last packet received in dBm. This is the signal level of the ping packet sent earlier. After reading the signal level the program prints out the calculated delay and signal level. This data is then used to plot the delay on a graph/scatter plot.

analog.pl This Perl program uses the Xbee/API.pm module to receive analog measurements from XBee devices that are programmed to take periodic measurements from their analog ADC ports. The measurements are usually sent to the Coordinator, so this program runs behind the Coordinator node receiving sample frames. Sample frames contain the senders address, serial number and a bit field that tells what sample ports are enabled. From the bit field one can read the 10-bit analog samples. The XBee device has 4 ADC ports so the sample packet can contain up to 4 analog samples. You can connect many different sensors to the ADC ports, for example a temperature sensor or a light sensor.

analog.pl parses the sample packet and takes the samples and put it in a simple database, using the device serial number and the pin number as the key. This program was used to run long running tests of the analog sample taking capability of XBee sensors.

check.pl Using Xbee/API.pm, this program tries to enumerate all nodes connected to the current network by sending a Node Discovery request. After sending the requests it waits for replies from the other nodes on the network. For each packet received after the initial Node Discovery request the serial number of the answering node is printed out.

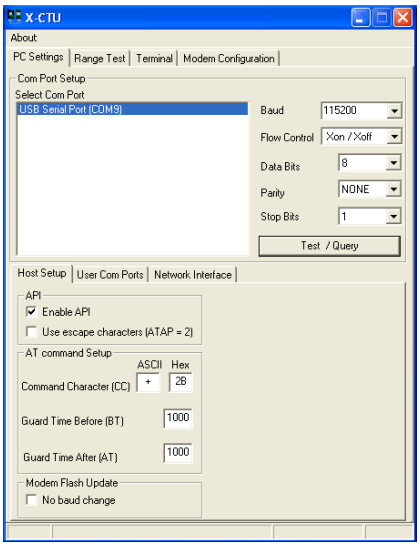

Figure 7: X-CTU Port selection screen

7.3 X-CTU

X-CTU is a Windows program created by Digi for configuring and testing XBee devices. We use this program for configuring the XBee devices we use for testing. X-CTU takes care of both flashing the XBee devices with new firmware and settings options.

Figure 7 shows the port selection screen where one can select witch serial port the XBee device is connected to and which mode it's running on. If it's running in API mode once needs to select API mode for communications to work.

Figure 8 shows the main configuration screen, the most notable options here is the "Modem:" which sets the firmware to flash the XBee device with. In the Figure it's set to XB24-ZB which is the firmware that contains the full Zigbee protocol stack. "Function Set" sets the function set and the mode the XBee device is running. This can be set to Coordinator, Router or End Device. The mode is either AT or API. The tree view shows all the options of the XBee device, some of the options are read only (Black). For example the "Operating Pan Id" is set when the node joins a network while the option "Join Notification" is read/write and can be set to 1 so that the devices sends a join notification broadcast when it joins a network.

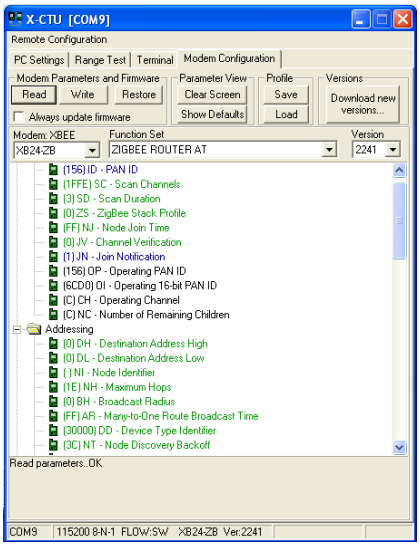

Figure 8: X-CTU Configuration screen

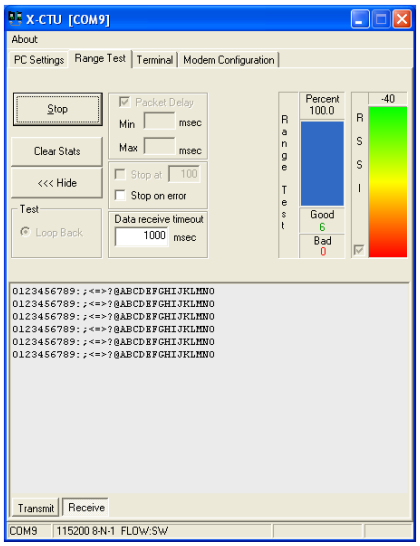

Figure 9: X-CTU Range test screen

Figure 9 shows the range test screen, this tool can be used to test the range of the XBee devices. This screen shows how many packets got through and the signal level "RSSI". The signal level indicator is in dBm, or the power ratio in decibels of the measured power to 1mW. A dBm value of -40 (Typical for a strong signal in XBee networks) is 100nW (Nanowatt) received by the antenna. -80 dBm is a weak signal at just 10 pW (Picowatt) received.

We found that this tool is not flexible enough for the tests we had in mind so we created our own tools that mimic some of the functions of this tool, plus some extra tools like Round Trip Time and data logging. These tools are described in section 7.2.

7.3.1 Register List and Descriptions

Here follows a list of the most important registers that can be set or read on XBee Devices flashed with the XB24-ZB firmware. The number of options differs according to the firmware and the mode of operation. For example Coordinators and Routers does not have some Sleep Options

Long Name	Name	Description
Pan ID	ID	Read/Write. Sets the PAN ID this node should operate on. If this register is 0 then the coordinator uses a random pan id. Routers and End Devices connect to the first available pan id if this is set to 0.
Operating Pan ID	OP	Read Only. The pan id this node is currently connected to.
Remaining children	NC	Read Only. The number of child nodes that can connect to this node. If this register is 0 no other nodes can join this node. Only valid for Routers and Coordinator nodes.
Destination Address	DH/DL	Read/Write. Sets the high and low part of the 64-bit destination address (Serial Number)
Node Identifier	NI	Read/Write. Sets a string that identifies this node. Used to look up network addresses.
Serial Number	SL/SH	Read Only. The serial number of this node. This is the serial number that should be set in the Destination ID of nodes addressing this node.
Network Address	MY	This nodes 16-bit Network Address
Encryption Enable	EE	Read/Write. If set to 0 no encryption is used. Set to 1 means encryption is used.
Encryption Key	KY	Write Only. Link key used to communicate with trust center. This register has to be set to securely distribute network key.
Sleep Mode	SM	Read/Write. End device only. 0 = No sleep. 1 - Sleep until sleep pin (Pin 9) transitions from high to low. 4 = Wake up on timer. (Cyclic Sleep). 5 - Wake up on Sleep pin or Timer.
Sleep Period	SP	Read/Write. Sleep period of the network. This value should be the same on all nodes of the network.
Time before sleep	ST	Read/Write. End device only. Sets the number of seconds of inactivity before going to sleep.
Cyclic Sleep Periods	SN	Read/Write. Sets the number of Sleep Periods a End Device should sleep before waking up and polling it's parent router for data.
Sleep Options	SO	Read/Write. Sets the sleep options bit-field for End Devices. 0x02 - Wake for Time Before Sleep on each cyclic wake (after sleeping SN*SP). 0x04 - Enable extended cyclic sleep (sleep for entire SN*SP time - possible data loss)
Route Broadcast Time	AR	Read/Write. The time between aggregation route broadcast time. An aggregation route broadcast creates a route on all devices in the network back to the devices that sends the aggregation broadcast. 0xFF=disabled. 0x00=Sends one broadcast.

Table 3: Some Important XBee Registers

7.4 Testing battery usage

Because we only have standard linear regulators for the power supply, we have to measure the power usage after the regulator to get an accurate reading. Section 2.4.3 explains how linear regulators works. We need to measure the current after the power supply because the power supply itself will use significant amount of current skewing results. This is especially important when measuring the power usage of XBee devices because it can sleep and use very little power. The nodes we use for these tests are XBee ZB Nodes from Digi. The power usage was tested using a oscilloscope and a 10Ω shunt resistor. This was done to study the power usage profile, or how the power usage changes over time. This shows us what the power consumptions is in the different modes of operation, if the Radio is on or if the CPU is active.

A shunt is a device that allows current to pass around another point in the circuit. The voltage drop over a shunt is proportional to the current running through it, and since the resistance of the shunt is known, one can measure the voltage drop and calculate the amount of current running through using Ohms Law.

For example a current of 1 ampere through a 1Ω shunt will generate a 1 volt drop over the resistor and shows up as 1 volt reading on an oscilloscope. In reverse if one measures 0.5V over the shunt and the resistance of the shunt is 1Ω the current running through the shunt is 0.5 ampere. This method of measuring current is mostly used in situations with very high current where measuring it directly is not an option. Using this method for the low currents XBee uses, is not very accurate but it clearly shows how the power usage changes in the different modes of operation.

The XBee device uses 3.3 volt as supply volatage in all tests.

7.5 Delay Measurements

7.5.1 Round Trip Time

The round trip time is measured by taking the time used from node to the destination and then back again to the original node. We take measurements using two programs listed below.

api_echo.pl This program listens for packets coming from the sensor network and sends the same packet back to the node that sent it. This program runs on the coordinator node. This program also records the route packes take by reading Route Record Indicator frames, this information is used to verify what path a packet has taken.

ping.pl This program encodes a high resolution time object representing the local time on the computer, then sends this encoded packet to the coordinator of the network. The program then waits for the packet to return and then decodes the packet and compares it to the current local time, effectively finding how long the packet took from the local node to the coordinator and back again. After calculating the round trip time the program reads the signal level on the XBee to find the signal level of the last packet received. After reading the signal level it prints out the round trip time and the signal level.

7.5.2 Hop

Zigbee uses mesh networking techniques so it's interesting to know how much the delay will increase if a packet has to traverse several routes before getting to it's destination. These measurements where taken by running the ping.pl program on a mobile platform (laptop) and first moving out of range of the XBee Coordinator, then placing a new XBee Router node just within range of the XBee Coordinator. The next step is to move in the same direction until the signal from the XBee Router node is also lost. Then we place a second XBee Router node just within range of the first XBee Router.

To verify which route a packet took trough the network, we use the Route Record Indicator frames received by the coordinator. These frames contain all the routers a packet has to traverse to get to it's destination.[2] This type of frame is only sent on a set interval (See section 7.3.1 for a description of the AR register that controls the route aggregation broadcast time.) so after a

test run we have to compare timestamps of the round trip time packet with the timestamps of Route Record Indicator.

7.5.3 Encrypted Link vs. Unencrypted Link

For a wireless sensor network to be secure one may need to use encryption. It's interesting to see how this effects the delay of packets traversing the network. This test was done using the same tools as in Section 7.5.1. First a XBee network with two nodes was created without encryption enabled. The round trip time over different signal levels where measured. Then the network was configured to use encryption and the round trip time tests where done again.

Figure 10: Picture taken of Vigeland Metal Refinery taken from the other side of the river Ora.

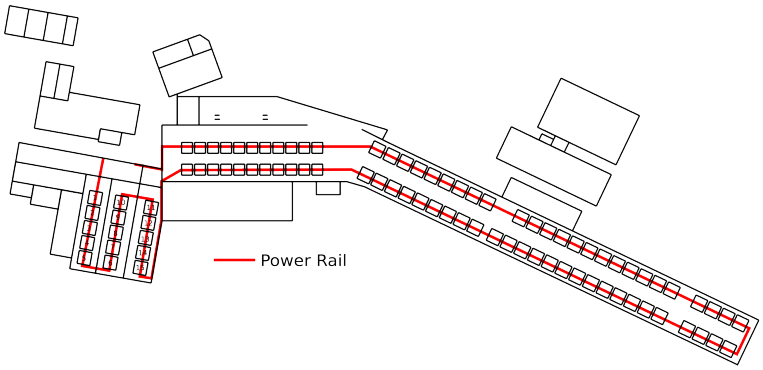

Figure 11: Schematic drawing of Power Rail path at Vigeland Metal Refinery AS

Figure 12: Picture taken of factory Hall C facing south.

7.6 Tests in industrial environment

This section covers how we tested the XBee devices in an industrial environment.

7.6.1 Environment Description

We tested the XBee sensor on the factory floor at Vigeland Metal Refinery A/S. This environment has some interesting obstacles for the sensors to overcome. Especially magnetic fields generated by the current running trough furnaces and power rails. While testing the current running trough the furnaces and the rails was 36000 ampere. This current generates significant magnetic fields in the whole factory. The magnetic fields are known to block signals to mobile phones. XBee sensors run on a different frequency than mobile phones but may be affected in a similar way. Other obstacles are concrete walls separating different parts of the factory and large metallic bodies (furnaces and other equipment) that may stand in the way of the signal.

Figure 11 shows the path the power rail takes through the factory. The current runs through every furnace from the bottom up through the liquid aluminum and out of the furnace via electrodes. The most significant magnetic fields are close to the power rail.

7.6.2 Magnetic field effect test

The effect of the magnetic field on the XBee devices was tested by first measuring the delay and signal strength at a location with lesser magnetic fields. Then we moved the XBee device into a location with a stronger magnetic field (Close to power rails, or between furnaces) but still at the same length away from the Coordinator or router.

7.6.3 Coverage Test

The coverage was tested by using a mobile platform with a XBee and running a ping program to see if the signal comes trough and how strong the signal is. Measurements where then taken in key locations round the factory at about one meter height over the factory floor. New router nodes where added to cover areas not covered by other nodes to find how many nodes would be needed to cover the whole factory.

For this test we used two of the standalone XBee Routers described in Section 7.1.1.

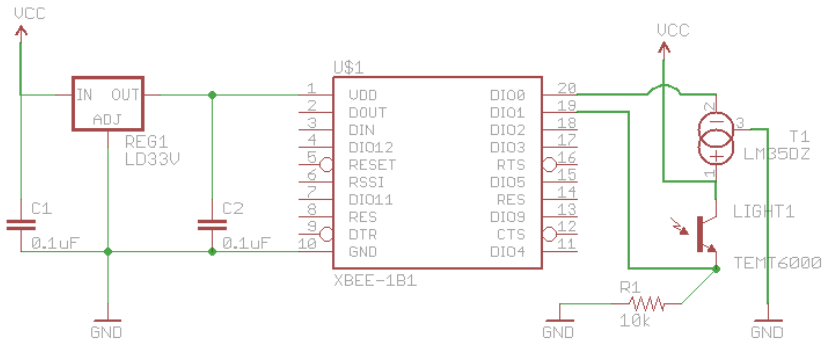

Figure 13: Schematic over test setup with XBee, Power supply, LM35DZ temperature sensor and TEMP6000 light sensors

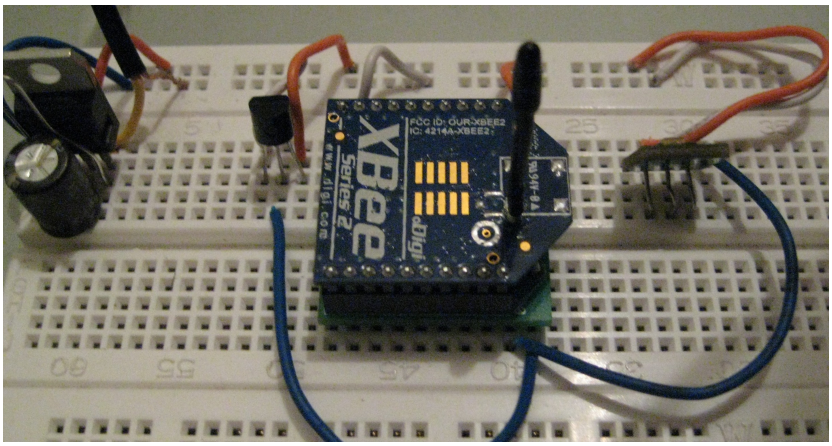

Figure 14: Prototype of the schematic in Figure 13 on a breadboard.

7.7 Implementing Test Sensor

For testing the periodic sample taking functionality of the XBee devices it is necessary to create a circuit with some analog sensors. Figure 13 shows the circuit of the remote sensor. It has two analog sensors, one for temperature and one for light, described in detail in the next section.

The reasons we want to test this functionality is that if the XBee can be used for reading analog sensors one can cut down on the number OS devices needed to take measurements, reducing cost and complexity.

7.7.1 LM35DZ Centigrade Temperature Sensor

The LM35DZ is a analog centigrade temperature sensor. This sensor output 10mV (Milli-volt) on pin two for each degree centigrade over 0 degrees centigrade. For example a output of 205mV is 20,5 degrees. Pin two of the temperature sensor is connected to Pin 20 (DIO0) of the XBee. The XBee can then read the mV value from the temperature sensor.

7.7.2 TEMT6000 Light Sensor

The TEMT6000 Light sensor combined with a 10kΩ pull-down resistor gives a varying volt output depending on the light intensity hitting the sensor. Figure 13 shows how the light sensor is connected to Pin19 (DIO1) of the XBee. The XBee can then read the mV value and get a light level reading.

7.7.3 Power supply

The power supply is a voltage regulator named REG1/LD33V in Figure 13. This part of the circuit provides 3.3 V for the rest of the circuit. C1 and C2 are capacitors to keep the voltage output of the linear regulator stable, and to provide a stable feedback loop to the regulator.

7.7.4 XBee Setup

The XBee is set up to take measurements of DIO0 and DIO1 analog pins every 30 seconds and send it to a Coordinator. This is done by setting the sampling rate register (IR) to 30000 milliseconds and D0 and D1 to 4 (Analog Input). The coordinator then receives the samples and stores the samples using the analog.pl program described in Section 7.2.2.

8 Test Results

8.1 XBee Power Usage

The power usage of the XBee devices depend on the mode it's running in. Below we cover what kind of power usage one should expect from Coordinator nodes, Router nodes and End Nodes.

8.1.1 XBee Coordinator

Our tests show that the XBee Coordinator node draws a constant current of about 40mA. According to the general product specification the XBee should draw 40mA when the radio is turned on. This means that the coordinator has the radio turned on constantly. The coordinator needs to keep it's radio on constantly to ensure minimal delay and retransmissions. In this mode the XBee routers draws 0.136 watts excluding all power supply circuits.

8.1.2 XBee Router

The router nodes has the same power usage profile as the coordinator. The router nodes draw a constant 40mA, this means the router nodes also keeps the radio on constantly. The reason both the Coordinator and Routers keep the radio on is that they need to listen for requests from all other devices within range. This means that Routers and Coordinators are not designed to run off a battery. In this mode the XBee routers draws 0.136 watts excluding all power supply circuits.

8.1.3 XBee End Device

End devices, in contrast to Coordinators and Routers have the option of sleeping and going into idle and are designed to run off a battery. Our tests show that End devices in idle mode use 8-10mA, this is close to the specification. In sleep mode the power draw is below the threshold of our measurement equipment. But the specification tells us that the power down current is $0.5\mu A$. Figure 15 shows a XBee configured as an End Device in a 5s sleep cycle. This means that the node sleeps for 5 seconds, wakes and takes a analog measurement of one Analog pin and sends a packet to the coordinator with the measured value. Here one can see that the device first sleeps near 0mV wakes up to idle at 100mV (or about 10mA). The peaks is up to 400mV (about 40mA) are measured when

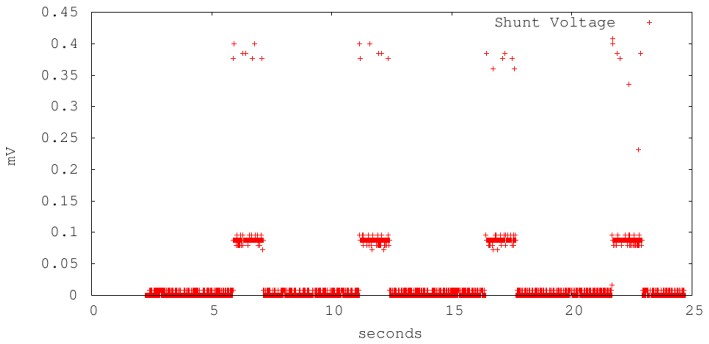

Figure 15: 10Ω shunt voltage. XBee End Device. Shows a XBee End device in 5s sleep cycle mode, where it wakes up every 5 seconds and sends an IO sample.

the device turns on the radio to send or receive data. Figure 16 shows a XBee end device transitioning from idle to sending/receiving (radio on) then back to idle. This is a closeup of one of the peeks in Figure 15. The radio is on about 5 milliseconds at a time in this case, and draws about 40mA. As the XBee draws this current at 3.3 volt the peak instantaneous power usage is 0.132 watts for 5 milliseconds at a time.

XBee End devices can sleep for extended periods thus saving battery in applications where the end device only needs to send samples on a cycle. If the end device needs to receive data it needs to wake up to check if the router it's joined to has data waiting. The router will not keep this data waiting indefinitely, thus the end device needs to wake up every SP (Sleep Period) seconds to query the router for data. The Sleep Period should be the same for all end devices and routers to ensure that Routers keep the data in the waiting queue long enough for the end device to wake up and get the data.

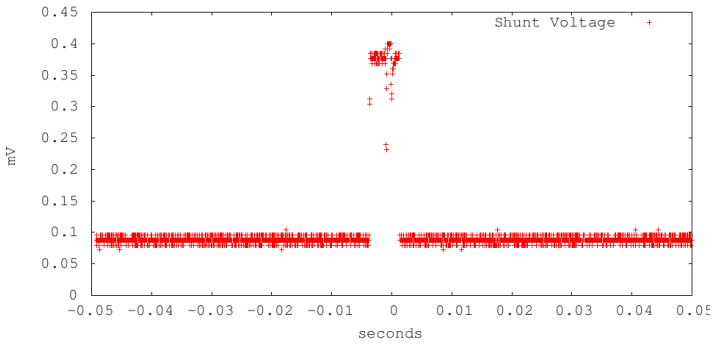

Figure 16: 10Ω shunt voltage. XBee End Device. Shows a XBee End device going from idle to sending/receiving mode back to idle. The peek lasts about 5 milliseconds.

8.2 Delay measurements

8.2.1 1 Hop delay

Figure 17 shows a scatter plot over the round trip time versus the signal level. One can see that the round trip time is usually between 50-70 milliseconds. Once the signal degrades the delay gets more unpredictable. This is due to packets not getting through and needs to be resent.

8.2.2 1-3 Hop delay

Figure18 shows the delay for 1 hop, 2 hops and 3 hops. The Figure shows that for each hop the delay increases but not by much.

XBee seems to prefer linking to routers that are fewer hops away from the coordinator, this made measuring the delay of 3 hops difficult. We could see the mobile XBee relinking with the first router even if the signal strength was much worse than the signal level for the second router. The specification doesn't say anything about this behavior but we assume that this is intentional as packets that has to traverse fewer hops creates less traffic on the network.

8.2.3 1 Hop delay on encrypted versus unencrypted link

Figure 19 shows a averaged scatter plot over delays for both encrypted and encrypted links. The measurements where taken with a XBee router node sending packets to a coordinator that sends that same packet back. First without any encryption options set and then with encryption enabled. Here the delay for each signal level has been averaged to show more clearly that the encrypted link has a slightly higher average delay. The difference is small 10-15milliseconds but 10 milliseconds here means the delay is over 10% higher when using encrypted link.

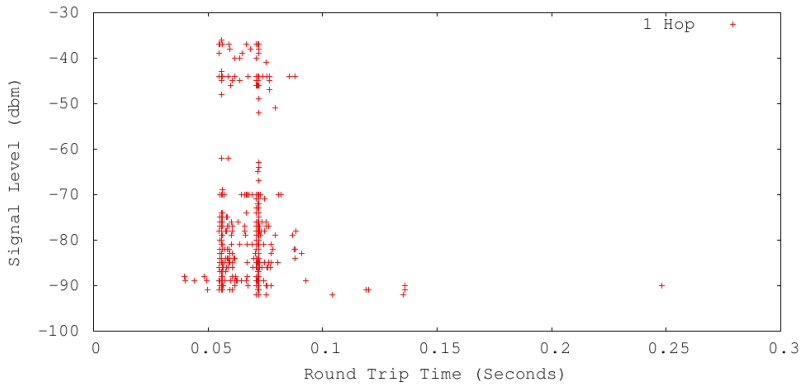

Figure 17: Scatter plot over round trip time versus signal level.

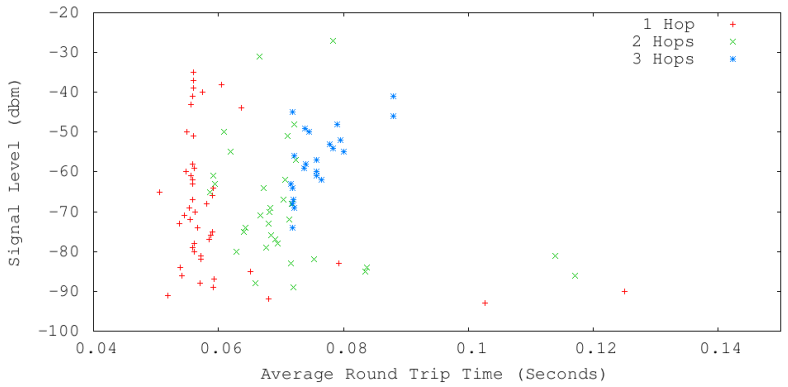

Figure 18: Averaged scatter plot over delay for 1 hop, 2 hops and 3 hops.

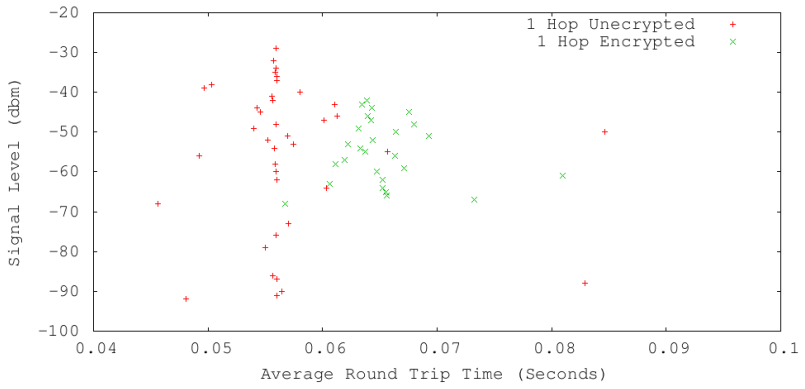

Figure 19: Averaged scatter plot of round trip times versus signal level for unencrypted link and encrypted link.

8.3 Results from our tests at Vigeland Metal refinery

8.3.1 Effects of the general environment

The effect of thick concrete walls is significant, the signal strength drops off when the mobile XBee moves out of the line of sight of the coordinator or router, and comes behind concrete walls. The effect of other objects in the signals path like (Furnaces, steel drums, etc) also has a negative effect on the signal but much less then the negative effect of solid concrete walls.

8.3.2 Effect of Magnetic fields on XBee

We could not discern any effect of magnetic fields on XBee devices, both on delay and signal strength. Even very close to the power rails (Where the magnetic field is the strongest) there seems to be no significant effect on the signal strength or packet delay. We also tried testing XBee devices in the 1 meter gap between furnaces where the magnetic field is strong, but we could still not see any significant drop in the signal strength.

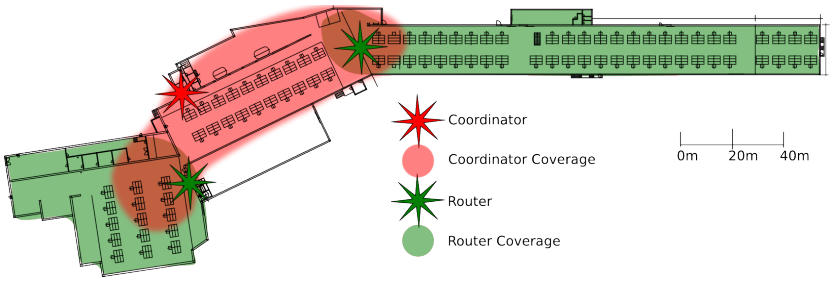

Figure 20: Coverage Map

8.3.3 Coverage test

Figure 20 shows the approximate coverage of the Coordinator and two Router nodes. The whole factory is covered with 3 nodes placed in key locations. We needed three is because parts of the factory halls are cut off by concrete walls interrupting the signal giving higher and unstable round trip times. By placing two routers within the line of sight of the coordinator they cover the whole factory with high enough signal strength.

At the end of hall D (Right in Figure 20) the signal strength starts to drop off but is still strong enough to give short and stable round trip times. The signal has to travel approximately 150 meter from the router to the end off hall D, the path the signal can take is mostly free air but even if the signal has to travel trough several furnaces the signal strength does not drop enough to affect the round trip times.

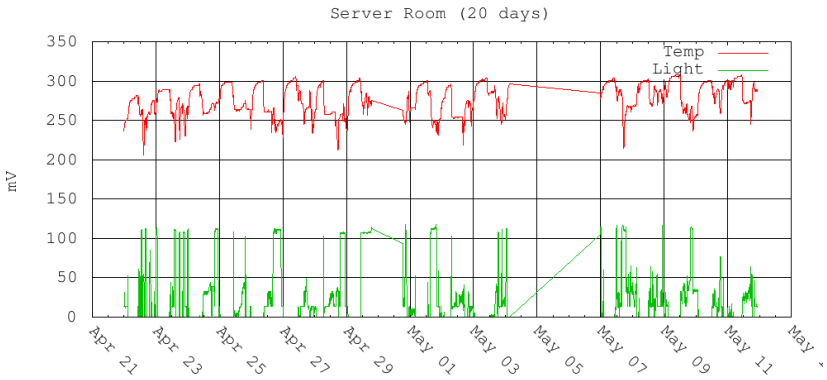

Figure 21: Sample of data captured with the sensor circuit described in Section 7.7.

8.4 Sensor test results

Using the XBee device as a standalone device with two analog sensors attached works as expected. Figure 21 shows an example of the data captured with the sensor circuit described in section 7.7. The data was captured in a server room over 20 days. There where two outtakes where samples where not gathered, Apr 30 - May 01 and May 04 - May 07. Both outtakes happened because the machine that captured the samples was not running and had nothing to do with the performance of the XBee devices.

The y-axis is in mV because the sensors attached to the XBee are analog. The temperature sensor outputs 10mV per centigrade above 0. A reading of 300mV means 30 centigrade.

9 Discussion

9.1 Choosing a network topology

ZigBee supports star topology as well as peer-to-peer networks such as wireless mesh.[13] A company should choose a topology depending on the physical location of the equipment. If all the equipment is close enough together a star topology is the best choice. In a star topology all the devices can only communicate with the ZigBee coordinator making it only one hop to reduce latency. The ZigBee coordinator can also use GTS to ensure that all the devices gets to send their information.

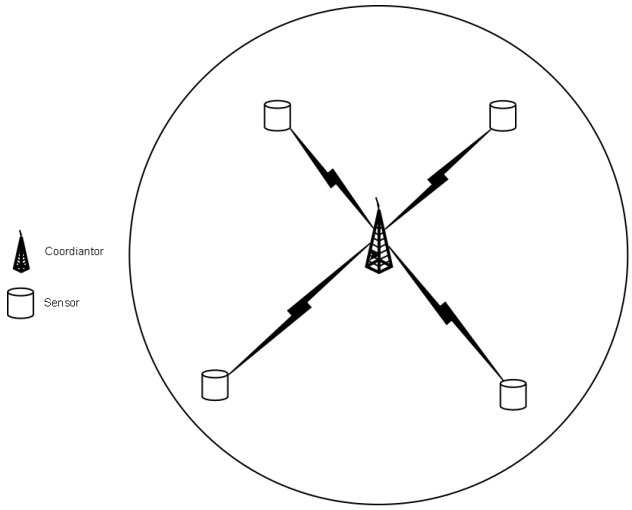

Figure 22: Star topology network

When the equipments are further away and the ZigBee coordinator can not reach all of the end devices, designated routers have to be set up. In this case a wireless mesh network can be used. A wireless mesh network is self-healing and self-organizing. ZigBee networks consists of two types of devices: Full Function Devices (Routers, Coordinators) and Reduced Function Devices (End devices). An end device can only communicate with ZigBee routers that are their parents. Messages meant for end devices is stored by the parent router until end device receives the data when it polls the router, hence a network should be set up in such a way that a there is a parent for every end device.

To expand the network with equipment that are further away you can use a cluster tree network. In a cluster tree network only PAN coordinators can com-

municate with each other. Each cluster can be set up with either star topology or wireless mesh. Additionally only end-devices should run on batteries, while routers and coordinators should not. Routers and coordinators is required to be always on, and depending on the use the battery could be drained in a matter of days.

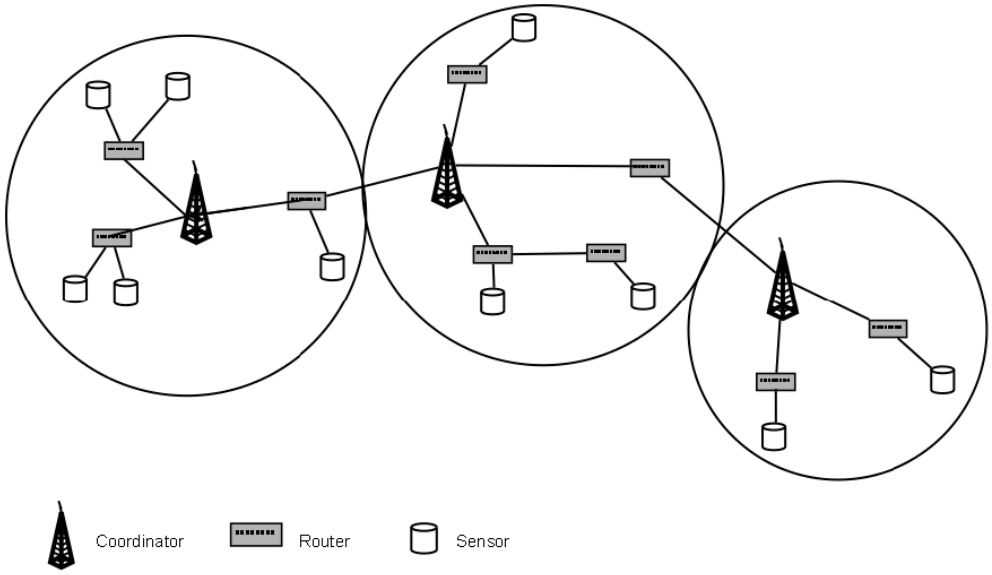

Figure 23: Cluster tree network with each cluster in a wireless mesh

9.2 Discussion of Reliability

The ZigBee protocol like most other protocols features acknowledgments and retransmissions in case a packet is lost. Before every transmission the source device will check if a route to the destination device is present. In a CBM scenario the end devices always sends messages to the network coordinator. However in scenario A where the sensor only sends messages when the equipment is in critical condition there is a need for some sort of mechanism to know that the sensor is still operational. For example the network coordinator can ping the end devices using a schedule. In scenario B where the sensors sends data periodically we can easily know if the sensor is operational, the end program that gathers information should have a function that alerts an user if data from a certain device has not been received in a given amount of time.

ZigBee uses CSMA/CA to access the radio channel, this in theory can make end devices wait before being able to send the data if the radio channel is busy. This can cause a problem if the end devices cannot send the data when an equipment is in a critical condition. This only applies to scenario A, because in scenario B the sensor sends data in a schedule and can be coordinated by using GTS. However this should not cause a significant problem for scenario B either, because it is highly unlikely that several equipment breaks down at the same time and cause so much traffic that one sensor gets denied access to the radio channel.

Our tests in an industrial environment shows that Wireless Sensor Networks, here represented by the ZigBee protocol and XBee devices, can be reliable even in the presence of some environmental obstacles that are common in industrial environments. Our tests show that Magnetic fields have no significant effect even very close to the source. Concrete walls do however pose a problem and increases the need for routers to extend the range of the network. The use of extra routers may increase the delay somewhat, but our tests show that the extra delay is no more than 10ms for each extra hop packets need to travel trough.

During our tests we discovered that encryption also cause more transmission delay, around 10ms. Also when a transmission is encrypted using the network key, every hop will decrypt and authenticate the packet and re-encrypt it, this adds to the transmission delay.

As long as the signal level does not fall to low the delay should be within 50ms for one hop.

9.3 Discussion of Security Requirements

ZigBee has an extensive set of security features, but does it fulfill the requirements of confidentiality, integrity and availability requirements needed for a CBM solution? The security requirements are different depending on which scenarios used in the CBM solution. For both confidentiality and integrity ZigBee and WirelessHART is very similar so we will only discuss the differences for the availability requirement.

Confidentiality

Confidentiality is different depending on the two scenarios. In Scenario A the sensor decides the condition of the equipment, and only an alert is sent. The alert can contain information about the equipment like why it was sent.

In Scenario B the sensor does not decide the condition of the equipment, but is used for monitoring. The sensors sends data frequently with new updates containing information about the equipment.

The frequency of data transmission in Scenario A makes it hard for an outsider to gather information about the equipment. The message only contains an alert, additional information such as which machine, what monitor information(temperature, vibration, etc)can be added as well. However there is no need for the additional information. The equipment in critical condition can be derived from the ID of the sensor, the monitor information can be decided by having several alert levels. In the case of a furnace Level 1 can mean it starts to overheat, Level 2 can mean its over the recommended level, Level 3 can be that its in critical condition and needs maintenance right away and so on. All this information can be in a central database of the company. This way an outsider gains no real information even if he intercepts the message. No confidentiality protection is needed by using this method. However the central database needs to be secure, but this issue is outside the scope of this thesis.

In Scenario B the messages sent from the sensor nodes contains information about the equipment. Since the sensor sends more frequently and has more information concerning the equipment, the company might want some sort of confidentiality protection. This can be accomplished by using encryption.

Integrity

Data integrity is equally important for both scenarios. Earlier we mentioned three concept when it comes integrity: data integrity, system integrity and authenticity.

Data integrity is vital in CBM. If there is no integrity check and outsider can replay a message and cause unnecessary maintenance or send a remote command to the sensor to send wrong information or not send information at all causing equipment to break down. Data integrity is achieved by using Message Integrity Code. A MIC is appended at the end of every frame containing a hash of the message. The destination device then recalculate the hash when received and compares it with the original hash. The packet is discarded if the hash doesn't match. The sensor nodes will only accept commands coming from a trusted source like the ZigBee coordinator or Trust Center, preserving data integrity.

To protect system integrity some sort of authorization method needs to be implemented. The devices has no such method. If an outsider is able to gain physical access to a ZigBee device he can easily reprogram the device. The only way is to have good physical security and only allow trusted employees access to the devices.

Authenticity of a message is achieved through entity authentication in high security mode. Entity authentication is based on a challenge-response mechanism based on a pre-shared secret. In this case a master or network key.

Both ZigBee and WirelessHART uses AES-128 in CCM mode. This allows data to be confidential and authenticated by devices with the correct keys.

Availability

The worse threat to availability in a Wireless Sensor Networks is jamming. ZigBee uses DSSS for modulation. DSSS only uses one frequency and is therefore more vulnerable to jamming compared to WirelessHART that uses FHSS. ZigBee only scans frequency and channel during startup, so during jamming the frequency and channel have to be manually set.

When an attacker compromises a node or is able to join a malicious node in the network there are attacks that specifically targets availability of the network. Such as routing loops and selective forwarding. These attacks increases the latency and affects the availability. As with integrity protection entity authentication can be used to avoid these attacks.

9.4 Securing the network

ZigBee features an extensive amount of security features that can be used to secure the network. In this section we will discuss what features we recommend for setting up a secure ZigBee network for use in an industrial environment.[6] Note that not all vendors supports every security feature specified.

Protect the network using a Network Key ZigBee features three security keys, the Master, Link and Network keys. While the use of master and link keys are optional, the network should at least make use of the network key. The network key is shared among all the nodes within the network and can be used to all messages within the network. Nodes without a valid Network key cannot join or send messages in the network.

Use address filtering at the MAC layer If the ZigBee devices supports this feature, it should be used so the ZigBee devices only accepts nodes from authorized sources.

Employ encryption

As mentioned earlier, ZigBee provides data confidentiality by utilizing AES-128 encryption. Encryption should be used to protect transmitted data.

Employ high security mode If possible use high security mode within the network. High security mode mandates the use of key establishment using SKKE protocol and entity authentication. High security mode permits the use of Master, Link and Network keys, while Standard security mode only permits the use of Link and Network keys.

Designate a ZigBee coordinator ZigBee supports wireless mesh networks that is self-healing and self-organizing, one node however must function as the coordinator and initiate formation of the network as well as perform other essential functions as sending beacon transmissions and setting the security level of the network. Any Full Function Device can be the coordinator, but from a security perspective this flexibility is not desirable.

It is recommended to set one designated node to be the coordinator and disable all other FFD devices from being able to take over the role as coordi-

nator. It is also a good idea to have a backup coordinator in case the primary coordinator fails.

Pre-assign a PAN identifier and restrict node connectivity It is especially important to pre-assign a node with PAN identifier if the ZigBee network is using a cluster tree network. The node is only permitted to join the cluster with the same pre-assigned PAN identifier.

Secure join procedures Some ZigBee units can acquire the key through a physical connection. This is the optimal choice for getting the initial Master or Network key. Only devices with the correct Master or Network is allowed to join the network. This will result in no keys is sent over the air and secrecy of the keys are protected at all times. If it is not possible to acquire the keys this way the second alternative is send the initial key over the air, but using a different channel leading to a brief moment of vulnerability, but less risk compared to using the primary channel. If possible pre-load the Trust Center address along with the initial Master or Network Key.

Additionally the coordinator can turn off the option for new devices to join the network. So even if an outsider knows the security keys they can't make a malicious node join the network.

10 Conclusion

One of the reasons Wireless Sensor Networks has yet to break through in the industry is the security and reliability challenges in a wireless network. Some companies cannot risk to have information leak out to outsiders and therefore uses a wired connection instead of a wireless network. While wired connections are safer in many ways, wireless networks is more cost effective. However wired connections does not always work in all applications, one such application is Condition Based Maintenance. The equipment or parts that needs to be monitored can be moving or not easily accessed with a cable.

In a CBM solution the information sent over the air is not so sensitive that loss of these information would lead to company losses. Depending on how the sensors are used encryption of messages might not be necessary at all. However based on our tests the transmission delay and power consumption in an encrypted message is not significant higher so using encryption does not shorten the life span of the battery significantly and while it might not be necessary to have encryption a company network should always be encrypted. Both ZigBee and WirelessHART offers encryption by using AES-128 in CCM mode, and while you can use brute-force to gain access to the content of the message the amount of time and contents might not be worth it.

More importantly than confidentiality is integrity. While a loss of confidentiality might not cause any company losses, a loss of integrity can cause losses and unnecessary maintenance. ZigBee offers integrity protection by using a Message Integrity Code. The MIC can be set to 32, 64, or 128-bit. The bit-length of the MIC determines the probability that a random guess of the MIC would be correct.

Availability is major concern when it comes to Wireless Sensor Networks, some method to detect when sensors are down needs to be implemented. Especially since ZigBee is more vulnerable to jamming compared to WirelessHART.

Based on our basic tests the sensor network seems to be reliable enough for use in CBM. The end devices can last up to years depending on the frequency of data transmissions on a battery. The delay and reliability mechanisms ensures that messages get through, and the range of the ZigBee devices could cover a relatively large area if there is no obstacles.

Wireless Sensor Networks seems to be able to fulfill the requirements needed for a CBM solution, but before deploying a WSN some considerations should be needed. Such as the basic security design. ZigBee has some unique challenges

when it comes to security, and coordination between the end program and the network is needed to be able to cover the weaknesses in ZigBee. Such as device detection to ensure a node is up and running, or policies for allowing a new device to join as most attacks revolves around compromising a node or be able to let a malicious node join.

From what we have read about WirelessHART it does seem that it's more robust than ZigBee with functions as FHSS and redundant paths. However based on our tests we believe that ZigBee can deliver a reliable service in most circumstances as long as there is procedures followed when the network might be exposed for jamming.

As security in ZigBee networks is not mandatory, some research before purchasing ZigBee units need to be done. To be able to maximize security in the network make sure the ZigBee units is compatible with high security mode.

The technology in this thesis are in constant development, and in the near future even stronger security or reliability will be implemented.

Further work

In this thesis we only had a network consisting of small number of nodes. A larger scale network would provide more information and could be able to run simulations of a CBM solution and tests the different scenarios specified in this thesis.

Experiments was only conducted on nodes using the ZigBee standard, and while we theoretically compared it to WirelessHART we did not have any real hands-on experience with WirelessHART. Tests using the WirelessHART standard would prove if our assumptions are correct.

According to the ZigBee specification, the memory required for the Trust Center grows with the number of devices in the network. This could lead to limitations on the size of the network and would require a cluster tree network. Test of memory requirements would be needed for a larger network.

Other areas of study is using public-private key encryption in embedded devices. Using public key encryption in these kinds of networks would greatly increase the security and privacy. Of special interest is TinyECC which is a software package that provides Elliptic Curve Cryptology (ECC) based Public Key Cryptology (PKC) for embedded devices like sensors in a wireless sensor network. TinyECC provides a digital signature scheme, a key exchange protocol, and a public key encryption scheme. Using TinyECC it is already possible to

use PKC on many embedded devices with TinyECC, but it takes significant resources (CPU, Power consumption) to do[11].

References

[1] Analog Devices. Voltage regulators for power management. `http://www.analog.com/library/analogDialogue/archives/30-4/Voltage.html`.

[2] Digi International. *XBee / XBee-PRO ZB RF Modules*, 2009.

[3] C.P. Fleeger. *Security in computing, 3rd edition*. Prentice-Hall Inc. NJ., 2003.

[4] IEEE Computer Society. *Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks(WPANs)*, 2007.

[5] Tomas Lennvall, Stefan Svensson, and Fredrik Hekland. A comparison of wirelesshart and zigbee for industrial applications. *Factory Communication Systems, 2008. WFCS 2008. IEEE International Workshop on*, pages 85–88, May 2008.

[6] Ken Masica. Securing zigbee wireless networks in process control system evironments. Technical report, Lawrence Livermore National Laboratory, Apr. 2007.

[7] Maxim-IC. Linear regulators in portable applications, Jul 2002. `http://www.maxim-ic.com/appnotes.cfm/an_pk/751/`.

[8] Modbus-IDA. *Modbus Specification*, 2006.

[9] William Stallings and Lawrie Brown. *Computer Security: Principles and Practice*. Pearson Education, Inc., 2008.

[10] STMicroelectronics. *LD1117 Series linear regulators data sheet*, 2005.

[11] North Carolina State University. Tinyecc: A configurable library for elliptic curve cryptography in wireless sensor network, Feb 2007. `http://discovery.csc.ncsu.edu/software/TinyECC/`.

[12] T. Zia and A. Zomaya. Security issues in wireless sensor networks. *Systems and Networks Communications, 2006. ICSNC '06. International Conference on*, pages 40–40, Oct. 2006.

[13] Zigbee Alliance. *ZigBee Specification*, 2008.