Design of a Wireless Sensor Node for Vibration Monitoring of Industrial Machinery

Alaa Abdulhady Jaber*, Robert Bicker**
* School of Mechanical and Systems Engineering, Newcastle University, UK
* Mechanical Engineering Department, Unversity of Technology, Iraq
** School of Mechanical and Systems Engineering, Newcastle University, UK

Article Info

Article history:

Received Nov 2, 2015
Revised Dec 11, 2015
Accepted Jan 6, 2016

Keyword:

Arduino microcontroller
Condition monitoring
Fault detection
Vibration analysis
Wireless sensor node

ABSTRACT

Machine healthy monitoring is a type of maintenance inspection technique by which an operational asset is monitored and the data obtained is analysed to detect signs of degradation, diagnose the causes of faults and thus reducing the maintenance costs. Vibration signal analysis was extensively used for machines fault detection and diagnosis in various industrial applications, as it responds immediately to manifest itself if any change is appeared in the monitored machine. However, recent developments in electronics and computing have opened new horizons in the area of condition monitoring and have shown their practicality in fault detection and diagnosis processes. The main aim of using wireless systems is to allow data analysis to be carried out locally at field level and transmitting the results wirelessly to the base station, which as a result will help to overcome the need for wiringand provides an easy and cost-effective sensing technique to detect faults in machines.So, the main focuses of this research is to design and develop an online condition monitoring system based on wireless technology that can be used to detect and diagnose the most common faults in the transmission systems (gears and bearings) of an industrial robot joints using vibration signal analysis.

Corresponding Author:

Alaa Abdulhady Jaber,
School of Mechanical and Systems Engineering,
Newcastle University,
Stephenson Building, NE1 7RU, Newcastle upon Tyne, UK
Email: a.jaber@ncl.ac.uk

1. INTRODUCTION

Based on the concept of wireless sensor networks (WSNs), which comprise of a number of battery-powered (or take advantage of nearby power supply if available) sensor nodes, each of which contains different (or the same) sensors types to monitor different variables and transmit the data wirelessly, embedded systems have been extensively used for building different health monitoring systems. The typical sensor node should be small size, low power consumption and low cost. WSN solutions are being increasingly employed in monitoring applications, for example in vehicle fault diagnosis [1]. The system comprises a large number of sensors able to communicate with each other through a wireless network, able to get live data from the vehicle, such as oil temperature, wheel balance, and fuel level. The embedded microprocessors gather the data and send them to an external monitoring entity. Another paper has suggested an intelligent diagnosis system combining WSN with a multi-agent system (MAS) [2]; to satisfy the needs for high sampling rates, high precision, high speed and large amounts of data transmitted from mechanical equipment. The efficiency of the system for a coal preparation plant was investigated, and its practicability was demonstrated. Other applications of WSN systemsare found in structural health monitoring. Rad and Shafai [3] utilized wireless embedded sensors as a successful alternative to fiber optics sensors to assess the state of the infrastructure of bridges in North America.WSNs have also shown sufficient potential in data collection when they have been applied to monitor wind turbine blades [4]. Here piezotronic accelerometers were used to pick up the signals from blades in both healthy and damaged states, and the sensors were fixed at different locations on the blades and wireless data acquisition utilized.Micro-electro-mechanical-sensors (MEMS) have also been used for condition monitoring. For example, a tiny and very light weight MEMS accelerometer has been mounted on a rotor shaft to monitor its dynamic behavior [5]. The accelerometer was connected to a wireless sensor node for the wireless transmission of vibration signals. Without any added imbalance and at different rotating speeds, vibration measurements such as acceleration values were taken with acceptable performance. It was reported that this technique assisted in reducing the number of sensors needed to monitor the rotating parts. A wireless embedded system has been applied to helicopter gearbox monitoring [6], with the aim of designinga sensor node fixed to the planetary gears' carrier in order to gather vibration signals to an external receiver through the antenna which extends into the gearbox. The acquired signal was analyzed using signal processing methods. An experimental system consisting of a set of planetary gears built using one sun gear and four planetary gears was constructed, and four wireless sensor nodes were installed in the space between each two neighboring gears.

A research study has implemented the envelope analysis algorithm for wireless bearing health monitoring based on vibration signals measurement, however, to overcome the limitations of memory size and restricted computational capabilities in the commercially available wireless nodes, the authors have used a 32-bit microcontroller type TM4C1233H6PM from Texas Instruments along with Zigbee wireless module. Interestingly, the rapid developments in smartphones and portable devices have changed the traditional way of using them. Researchers have developed a scalable android application based on a smartphone to diagnose some types of fault in an industrial air compressor [8]. They mentioned that the developed system is very reliable. Another paper has presented a remote monitoring system for a rotating machine which can be run based on smartphone or PDA (personal digital assistant) [9]. In this paper the developers put the capability of informing the concerned user if a fault appears in the remotely monitored machine. Similarly, a paper proposes a real-time method to perform the monitoring of temperature, humidity, air quality and vibrations of operating machinery in a factory zone using smart phones [10]. The developed system was able to instantly provide numerical results, depending on the received and analysed data, to the smart phones of the factory manager. However, in this paper the aim is to design a low coast wireless sensor node that can be utilized for healthy monitoring of different machines via vibration signal analysis. Vibration analysis of an industrial robot has been used to confirem the performance of the designed node in signal capturing and transimiting.

2. DESIGN OF THE SENSOR NODE

The sensor node, as depicted in Figure 1, is an embedded system usually outfitted with a microcontroller unit (MCU) (or digital signal processor (DSP)), radio frequency (RF) transceiver, power supply, and memory along with various sensors and actuator depending on the application. The microcontroller with memory to store the data acts as the central computing and controlling device of the sensor node. The radio transceiver module, which allows two-way radio communication between several nodes in order to distribute the information, represents the communication subsystem of the node. The power supply subsystem is in charge of powering the whole sensor node, and is normally composed of batteries, which offer an easily available, low cost, and high capacity source of power, and have become companion with sensor nodes.In this work, one node will be fixed on the robot arm and will be in charge of performing the vibration signals capturing and then sending the information to the base station, which is connected to a PC for data visualization and robot health assessement. So, the node has to be of light weight and small size, easily attached and does not add extra loading on the robot arm. The node has to be of a reasonable price and easy to produce, for future expansion of the wireless systems. In this section the selection of the accelerometer, microcontroller, transceiver and the power supply source of the node is discussed. However, the proposed design methodology is not limited for industrial robots and could be utilized for health monitoring of different machines.

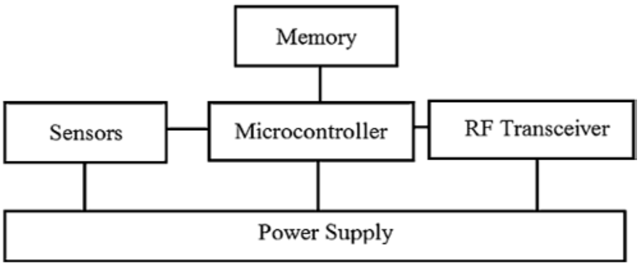


Figure 1. Wireless sensor node hardware architecture [11]

2.1 Accelerometer Selection

Accelerometers are electromechanical devices that convert the mechanical signals, such as vibration and force, to electrical signals, and are an extensively used for fault detection in many machines because of their accuracy, robustness and sensitivity. Selecting an appropriate accelerometer and the way it is mounted on a machine are significant factors in determining the success of any condition monitoring program. Misleading data can be produced if an unsuitable accelerometer is selected for the machine under study or an appropriate one is mounted in an incorrect location on the machine. The major parameters which need to be considered for accelerometer selection are [12]: sensitivity, range, bandwidth, frequency resolution, reliability, accuracy, operation environment and cost. MEMS (Micro-electro-mechanical-sensors) accelerometers compared with the conventional piezoelectric accelerometers, consume less power, are small in size, light weight, low cost and can achieve good performance [13]. Consequently, they are more suitable for wireless systems design, and will be used here. However, a variety of MEMS accelerometers are commercially available. After taking the all-important technical requirements in the consideration, the ADXL001 MEMS accelerometer has been chosen.

The ADXL001 is a MEMS based, single-axis accelerometer developed by Analog Devices. It provides a high performance, wide bandwidth of 22 kHz, and is small in size. The ADXL001 operates on a 3.3V or 5V supply, and outputs an analogue voltage, which allows the direct connection of the accelerometer output to the analogue input pins on a data acquisition device (DAQ) or a microcontroller. The accelerometer is available in 3 full-scale dynamic ranges of ±70, ±250, and ±500 g. For this work the ±70 g range was deemed appropriate. An evaluation board is specifically designed by Analog device for this accelerometer making it easier to use. To prevent anti-aliasing, the evaluation board provides a user configurable low-pass filter on the accelerometer output. Moreover, due to stress, over acceleration or fabrication errors MEMS accelerometer can develop flaws; and fortunately, ADXL001 accelerometer like other MEMS accelerometer has a build in self-test pin, which can be used to test both the electrical circuit and the mechanical structure of the accelerometer. Furthermore, using just one accelerometer for robot health monitoring was not considered sufficient, since when the robot is performing a general task its joints are rotating around different axes, therefore, the accelerometer may be more sensitive to specific joints than others yielding the monitoring system unreliable. For this reason, it was decided to fix three ADXL001 accelerometers in an orthogonal configuration to measure the vibration in X, Y, and Z directions using a purpose designed aluminium adapter, as shown in Figure 2. To fix this adapter correctly on the robot for accurate pick-up of vibration signals, adhesive mounting using superglue was adopted here.

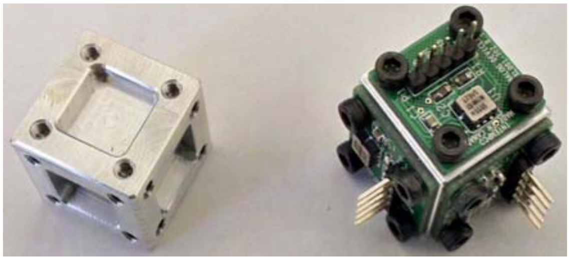


Figure 2. Designed 3 axis accelerometer adapter

2.2 Microcontroller Selection

A microcontroller unit (MCU) is a programmable device that combines electronics components, include central processing unit (CPU), memory and peripheral devices, integrated into a single chip [14]. There are two families of microcontrollers: CISC (Complex Instruction Set Computer) and RISC (Reduced Instruction Set Computer). CISC is characterized by a single bus between the CPU and memory, which means a single memory for both data and instructions, and is commonly known as Von Neuman architecture. Because of this, it involves more than one machine cycle for theexecution of each instruction; this constitutes the disadvantage of this family. RISC microcontrollers, on the other hand, are the Harvard architecture, which provides the program and data memory physically separated from each other, two separate buses between the CPU and data-memory, and between the CPU and program memory. The microcontroller to be utilized in a sensor node is responsible for the acquisition, processing, compression, recording and storage of data. There are a considerable number of microcontroller boards produced by different companies available, and key requirements that need to be taken into a countwhen selecting a microcontroller for constructing a sensing node are energy consumption, cost, processing speed, memory size, physical size and support for peripherals. The selected microcontroller must have a fast, high resolution analogue to digital converter (ADC) with at least three input channels, one for each of the three single axis MEMS accelerometers. The sampling frequency of the ADC has to be fastenough to cover the highest sampling frequency that could be used for capturing the vibration signals from the accelerometers. So, in order to select an appropriate microcontroller to be used in the sensor node, a comprehensive evaluation of the available products has been performed. Several microcontroller families from, for example, Microchip, Arduino, and Texas Instruments companies have been assessed. The Arduino was found to be well suited for this project, as it meets the requirements and also has a large community support which makes the development process simpler.

Arduino offer several microcontroller models with different characteristics. The main differences between these models are the type of processor, the number of inputs and outputs ports, and the capacity of memory. Arduino controllers are relatively efficient, consume less power, cheap, and suitable for use in a harsh environment [15]. To reduce the number of available Arduino options it was decided to look at Arduino boards that have high specification and meet the desired requirements, such as DUE and Intel Galileo, which is a new board designed to be compatible with Arduino hardware and software and is based on Intel architecture.The Intel Galileo board was thoroughly tested, but despite the high functionality it was established that the signal capturing process takes very long time in order to capture 4096 samples, as it is the default number of samples, from one analogue input channel; as a result was rejected. The Arduino DUE board was tested and found to be more than capable for this work in terms of signal capturing speed, processing speed, memory size, power consumption and, of course, cost.The Arduino DUE (Figure 3) is an open-source, single-board microcontroller based on a 32-bit, RISC, Atmel SAM3X8E ARM Cortex-M3 processor. It offers a relatively small size form, measuring 101.6 mm x 53.3 mm, and compatibility with most of the standard Arduino shields. The Arduino DUE board has an 84 MHz clock frequency, USB connection, four UARTs (Universal asynchronous receiver/transmitter) serial ports, a power jack, a reset button, and an erase button. There is a 12-bit resolution analogue to digital convertor (ADC) built in inside the processor with 1 MSPS (mega samples per second) sampling frequency and 12 input channels. The board comes with 512 KB flash memory, and 96 KB of SRAM.Whilst other Arduino boards accept up to 5V on I/O pins, the Arduino Due board is based on 3.3V on the I/O pins, fortunately, the selected accelerometer can work using 3.3V or 5V which makes the connection of the accelerometer to the board does not require any conditioning circuit.

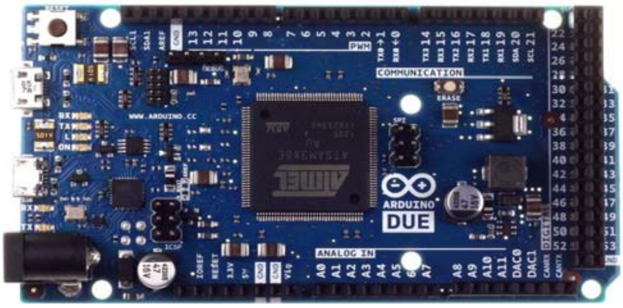


Figure 3. Arduino DUE microcontroller board [https://www.arduino.cc]

2.3 Wireless Technology Selection

Communication devices are used to exchange data between the nodes of wireless networks. The advances in wireless applications have led companies to develop different types of wireless standard. These standards are usually classified by their capabilities and properties, and designed to suit different applications, such as fault detection or human health monitoring. In this section, three widely used wireless network technologies are discussed, in order to investigate their pros and cons and also to select the appropriate technology for the work here. These technologies are [16, 17]:

The IEEE802.11x family of standards is meant for wireless local area network (WLAN), which is also known as Wireless Fidelity (Wi-Fi). There are four generations of Wi-Fi products available which are IEEE 802.11a/b/g/n, operate in high frequency, unlicensed Industrial, Scientific, and Medical (ISM) radio bands ranging from 2.4 GHz to 5 GHz. Typically, it is adapted for relatively high bandwidth and high data transfer rate, ranges from as low as 1 MBPS (megabyte per second) to over 50 MBPS, and commonly used for mobile computing devices, such as laptops. With the use of a standard antenna the transmission range can be up to 300 feet, and it can be significantly improved by utilizing a directional high gain antenna. Although the data transmission and rate ranges are enough for wireless sensor network application, the power requirement generally limited its usage in wireless sensor application. Bluetooth (IEEE802.15.1) is another standard that has lower power than IEEE802.11, an operating frequency 2.4 GHz within the ISM band, and represents a personal area network (PAN) standard. It is specifically aimed to serve applications that require short range communication, such as data transfer between computers and other peripheral devices like keyboards or cell phone to replace wire connectivity. Bluetooth supports star network topology, and can enable up to seven remote nodes to communicate with a single base station. However, in addition to its short range application and scalability problem (low number of nodes per network), the other disadvantages of Bluetooth are high power consumption and nodes that need a long time to be synchronized with the network when returning from sleep mode, and which increases the average system power.

The IEEE802.15.4 standard was designed for low data rate transmissions, low cost, and low power consumption wireless personal area networks (WPAN). In terms of communication range, this standard can be considered as a middle ground solution between IEE802.11 (Wi-Fi) and IEEE802.15.1 (Bluetooth), and supports multiple transmission frequencies, multiple data rates, and two topologies, star and point-to-point (pair topology), which makes it a flexible standard. It operates in the unlicensed ISM bands at 868 MHz in Europe, 915 MHz in the USA and 2.4 GHz worldwide, with data rates 20 Kbps (kilo bit per second), 40 Kbps and 250 Kbps, respectively. ZigBee is a standard designed by the ZigBee Alliance, which is an association of companies working together to enable reliable, cost-effective, and low-power wireless network [18]. It is based on IEEE802.15.4 standard which means that ZigBee can take full advantage of this standard. In addition it can accommodate multiple networks topologies like star, point-to-point, and mesh networks. Also, a ZigBee network can have at most 65000 nodes, making it a very scalable standard. Because of the aforementioned features, the ZigBee standard has been adopted in many wireless sensor network applications, and will be the best candidate for the work described in this thesis. Additionally, ZigBee modules are nowadays available in the market in small size with very affordable prices.

2.4 ZigBee Module Selection

There are many parameters need to be considered when selecting a ZigBee module. From these parameters, but not limited to them, are power consumption, operating frequency, flexibility, coverage range, the module and testing costs, and the compatibility with the microcontroller unit. However, many certified semiconductor companies, such as Texas Instruments, Freescal, Digi International, are providing successful design for ZigBee products. After revising the available products the XBee module from Digi International has been selected, as it meets the above mentioned requirements, is compatible with the selected Arduino board, and includes other factor, i.e. previous successful experience, popularity, and available development resources.XBee is a trade name from Digi international, and it is a wireless module designed for applications that require reduced data communication while having long range capabilities with less power consumption. XBee modules come in different formats for different kinds of application. There are two types of XBee modules, series 1 (s1) and series 2 (s2). As shown in Table 1, XBee s2 consumes slightly less power and has a better range than the s1. The XBee s1 and s2 are pin-for-pin compatible, but based on different chipsets and running different protocols. The s1 module uses the IEEE 802.15.4 standard protocol, while the s2 module relies on ZigBee protocol. Furthermore, both series come in two different transmission powers, regular and Pro. The regular version is simply called XBee, and is less expensive than Pro version; the Pro version uses more power and is slightly larger than the regular version, but, on the other hand, has a longer communication range. The XBee s2 (Figure 4) has been selected for this research, as it provides good in-door range, good data rate, low-energy consumption, better receiver sensitivity and it supports mesh and tree networks. This makes the system scalable and reliable for future development. XBee modules can be attached to DUE using a wireless shield (Figure 3) designed for easy connection to the XBee. The shield acts as a daughter board and is attached on top of the DUE. It has an optional on-board micro SD-card connection capability for serving data on a card over the network. In the shield there is an on-board switch labelled Serial Select, which determines how the wireless shield's serial communication connects to the serial communication between the microcontroller and the USB-to-serial chip on the Arduino board. The switch allows two settings which are Micro and USB; in Micro mode, the XBee module will communicate with the microcontroller and the sent data from the microcontroller will be transmitted to the computer through USB as well as being sent wirelessly by the wireless module, but the microcontroller will not be programmable via USB in this mode. In USB mode, the microcontroller on the board is bypassed and the module can communicate directly with the computer, and helps in utilizing the Arduino’s USB-to-Serial connection to configure the XBee modules.

Table 1. Comparison between XBee s2 and s2

Specifications	XBee s1	XBee s2
Indoor range (m)	30	40
Outdoor range (m)	100	120
Frequency band (GHz)	2.4	2.4
Transmit power (mW)	1	2
Supply voltage (V)	2.8-3.4	2.8-3.6
Data rate (Kbps) (kilo bit per second	250	250
Transmit current (mA)	45	40
Receive current (mA)	50	40
Receiver sensitivity (dbm)	-92	-96

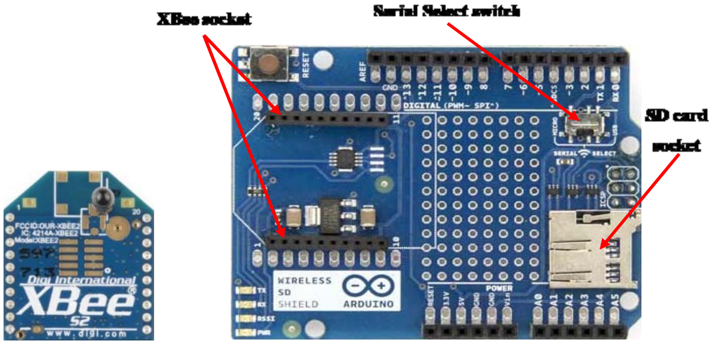


Figure 4. XBee module and the wireless shield for Arduino [https://www.sparkfun.com;https://www.arduino.cc]

2.5 Antenna Type Selection

XBee modules must have an antenna to send and receive signals, of which there are four different types offered on XBee s2, namely whip, chip, U.FL, and RPSMA, as illustrated Figure 5[19]. The wire (or whip) and chip antennas come pre-connected to the XBee modules, while the U.FL and RPSMA are derivatives of the connector types, offering chip with connectors on the board. The wire antenna is a single piece of wire that protrudes from the XBee and provides an omnidirectional radiation, which implies the maximum transmission distance in all directions is the same if the antenna is oriented in upright direction perpendicular on the module. The chip antenna is a flat ceramic chip that is mounted on the XBee. The form of signal radiation by this antenna is heart-shaped or cardioid, which means that the signal will be attenuated in many directions. However, because the chip antenna is nearly flush, that makes it a suitable choice for any sensor that needs to be located in a small space. It is also robust compared to the whip antenna which is subjected to mechanical stress. In some application the XBee module needs to be fitted inside a metal box; in which case a U.FL antenna has to be used. This type of antennas has a very small connector on the module and to achieve the physical connection to the external antenna an adapter cable is used. The advantage of this option is the XBee module can be enclosed in a casing and the antenna mounted on the outside of the case. Similarly, the RPSMA connector is a different type of socket from the U.FL connector. It is bigger and mounted directly to the XBee without a connecting cable. These last two options are more expensive. Generally, an external antenna allows better signal transmission and reception, and a larger rage. The indoor range of the selected XBee module is only 40 meters and sometimes noise in the medium distort the signals and thus shortening the range. Therefore, it was decided to utilize an external antenna type RPSMA.

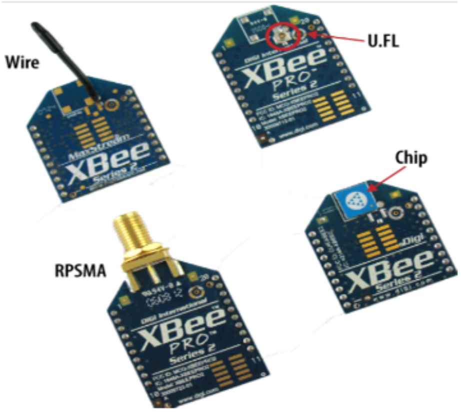


Figure 5. Types of antenna [19]

2.6 Power Source Selection

Rechargeable batteries are commonly employed as a power source for the wireless sensor nodes. Several characteristics should be considered when choosing a battery for a wireless sensor node. The most important of these attributes are energy density, charge/discharge cycle, size, self-discharge rate and cost. There is no any battery technology that meets all these criteria, so a compromise must be made. Each battery type has its advantages and disadvantages with a wireless sensor node. The most common rechargeable batteries types are: lead acid, nickel-cadmium (Ni-Cd), nickel-metal hydride (Ni-Mh), lithium-ion (Li-Ion), and lithium-ion polymer (Li-Po). The comparison of the different types of batteries can be seen in Table 1[20]. Lithium-based batteries are technically more advanced, most widely used and fastest growing energy sources. Comparing to the other three battery types, Li-Ion and Li-Po have higher specific energy densities and offer a lower self-discharge rate than both Ni-Cd and Ni-Mh, with only lead acid having a lower rate. Lithium-Po batteries are similar to Lithium-Ion however with a different type of electrolyte used, and furthermore offer the advantage of being very thin and light weight, thus allowing them to be easily included in sensor nodes while occupying very little space.In this regard, a Turnigy Li-Po battery (Figure 6) has been chosen as a power source for the wireless node. The battery has an energy capacity at 7.4 V/1000 mAh (1 hour continuously working at 7.4 V with 1 A discharge current) with weight 62 g and size 74 x 35 x 13mm, which can fit easily in the node enclosure as shown later.

Table 2. Comparison among rechargeable batteries

Specifications	Lead Acid	Nickel-Cadmium	Nickel-Metal hydride	Lithium-Ion	Lithium-Ion polymer
Energy density (Wh/kg)	30-50	45-80	60-120	110-160	100-130
Charge/Discharge cycle	200-300	1500	300-500	500-1000	300-500
Self-discharge/Month	5%	20%	30%	10%	~10%

Figure 6. Lithium-Ion polymer (Li-Po) battery

3. XBEE MODULES CONFIGURATION

In WSN must bethere is a node configured to work as a coordinator (or base station) [19] and the others as sensor nodes. The coordinator represents the network controller and is responsible for setting up and maintaining the network, which should be connected to a reliable, uninterrupted power source, and requires to be interfaced to a PC for further data processing and visualization. In order to allow the transceiver modules on the sensor node and base station to talk to each other, they have to be correctly configured individually, by utilizing specific software, before using them in a network. The configuration includes various aspects such as classification of the node (coordinator, router or end device), network ID, destination address and so on. X-CTU (XBee configuration and testing utility) is windows-based application software developed by Digi and represents the official configuration software for XBee modules [19]. Many versions of firmware can be selected and written into aZigBee module via RS232 or USB port, depending on the used interface board. For this purpose, the XBIB-U-DEV development kit (Figure 7) from Digi International was employed, and will be utilized later for the coverage range test. Figure 8 shows the layout of X-CTU software and its four main tabs. The PC settings tab is used to find the port through which the XBee module has been connected and also to help the user to select the XBee module for configuration from a range of plugged in devices. The range test tab is utilized to perform a wireless communication coverage range test, as will be explained later. Terminal tab is used to open the X-CTU terminal window, which can be used to read the data being received by the connected module. The modem configuration tab allows changing the firmware version, writing the firmware setting to the module, as well as setting the module as a coordinator, router or an end device by using its three sub-tabs drop-down menus namedmodem XBee and function set, respectively. The Firmware is instructions programmed in the module’s memory which controls the device and provides several instructions on how the devices can communicate with other hardware. The configuration progress includes two stages, coordinator configuration andend device (sensor node) configuration. In practise, when these modules are configured, the coordinator automatically scans to select a communication channel and it always listening to the sensor node and receive the incoming data.

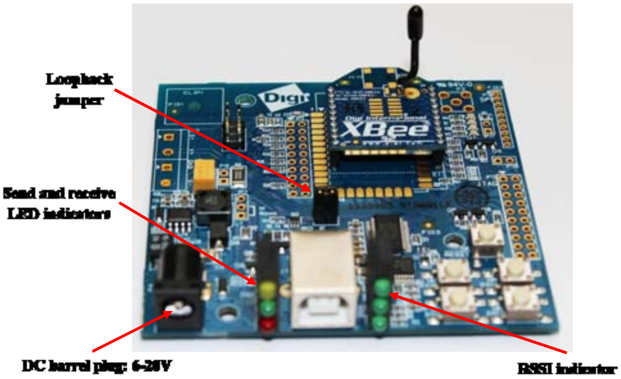
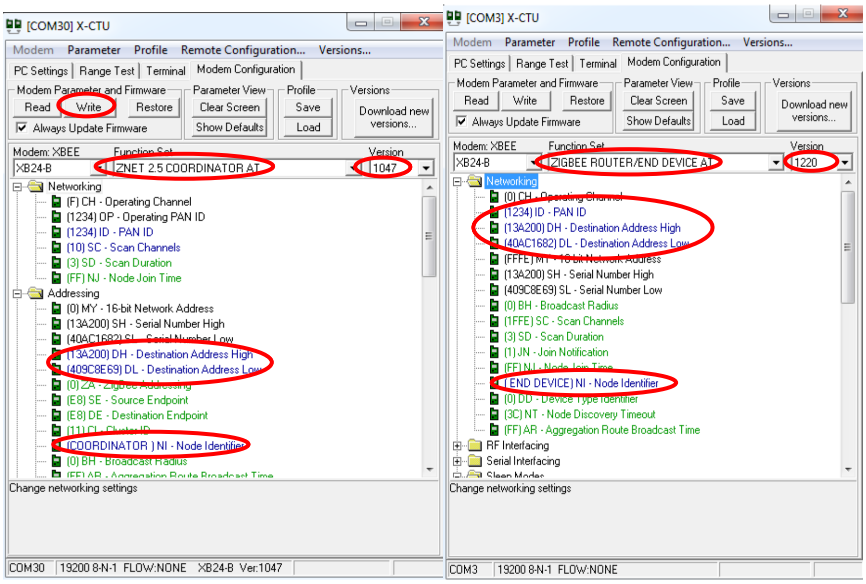


Figure 7. XBIB-U-DEV adapter for XBee configuration

The firmware type XB24-B, which supports the full functionality of ZigBee protocol, is selected from modem sub-tab in the modem configuration tab for both coordinator and end device, and the latest version of this firmware is required to be downloaded on the modules. The firmware supports coordinator, router and end device configurations. The configured parameters are listed in Table 3 and the screenshots showing the configurationof both modules are as shown in Figure 8. The function set sub-tab was used to configure the functions that the XBee modules provide, which are as a coordinator and an end device. However, all used XBee modules (here just two) must have the same personal area network (PAN) ID and baud rate e.g. 1234 and 19200. The destination address high (DH) and low (DL) for the coordinator should be the same as the serial high (SH) and serial low (SL) of the end device and vice versa, and which can be found written on the back side of the XBee modules. The node identifier is a user configurable text name that can be set to easily identify a module. Accordingly, by clicking the write button in the modem configuration tab, these settings will be downloaded on the connected module.

Table 3. The configured parameters in the XBee modules

Parameters to be configured	Setting	
	Coordinator	End device
Firmware	ZNET 2.5 COORDINATOR AT	ZNET 2.5 ROUTER/END DEVICE AT
Firmware Version	1047	1220
PAN ID	1234	1234
Destination Address High (DH)	13A200	13A200
Destination Address Low (DL)	409C8E69	40AC1682
Node Identifier	Coordinator	End device
Baud Rate	19200	19200
Other Parameters	Default	Default


[COM30] X-CTU
Modem Parameter Profile Remote Configuration... Versions...
PC Settings Range Test Terminal Modem Configuration
Modem Parameter and Firmware
Read Write Restore
Parameter View
Clear Screen Show Defaults
Profile
Save Load
Versions
Download new versions...
Networking
Addressing
Change networking settings
[COM3] X-CTU
RF Interfacing
Serial Interfacing
Sleep Modes

☑ Always Update Firmware

Modem: XBEE XB24-B Function Set ZNET 2.5 COORDINATOR AT Version 1047

(F) CH - Operating Channel
(1234) OP - Operating PAN ID
(1234) ID - PAN ID
(10) SC - Scan Channels
(3) SD - Scan Duration
(FF) NJ - Node Join Time
(0) MY - 16-bit Network Address
(13A200) SH - Serial Number High
(40AC1682) SL - Serial Number Low
(13A200) DH - Destination Address High
(409C8E69) DL - Destination Address Low
(0) ZA - ZigBee Addressing
(E8) SE - Source Endpoint
(E8) DE - Destination Endpoint
(11) CL - Cluster ID
(COORDINATOR) NI - Node Identifier
(0) BH - Broadcast Radius
(FF) AR - Aggregation Route Broadcast Time

COM30 19200 8-N-1 FLOW:NONE XB24-B Ver:1047

☑ Always Update Firmware

Modem: XBEE XB24-B Function Set ZIGBEE ROUTER/END DEVICE AT Version 1220

(0) CH - Operating Channel
(1234) ID - PAN ID
(13A200) DH - Destination Address High
(40AC1682) DL - Destination Address Low
(FFFE) MY - 16-bit Network Address
(13A200) SH - Serial Number High
(409C8E69) SL - Serial Number Low
(0) BH - Broadcast Radius
(1FFE) SC - Scan Channels
(3) SD - Scan Duration
(1) JN - Join Notification
(FF) NJ - Node Join Time
(END DEVICE) NI - Node Identifier
(0) DD - Device Type Identifier
(3C) NT - Node Discovery Timeout
(FF) AR - Aggregation Route Broadcast Time

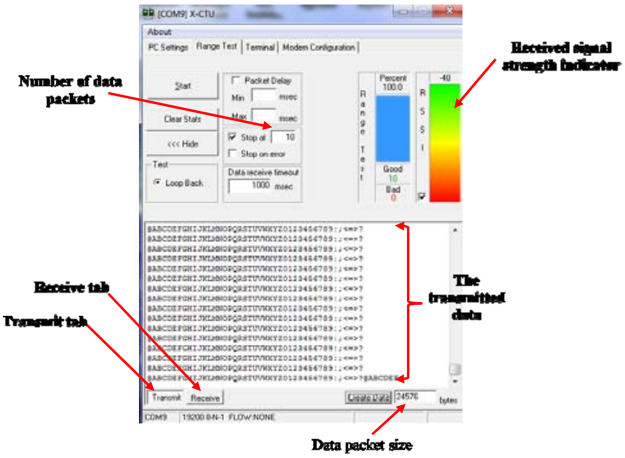
COM3 19200 8-N-1 FLOW:NONE

a) Coordinator configuration b) End device configuration

Figure 8. XBee modules configuration using X-CTU

4. COVERAGE RANGE TEST

The purpose of this test is to study the effects of XBee transmit power on the actual coverage range and communication quality between the base station and the sensor nodeunder real working conditions. The optimum configuration parameters will be the outcome of this study,and the received signal strength indicator (RSSI), which is defined as the signal strength level of a wireless device measured in (dBm) of the last received packet [21], is used. A loop-back test using X-CTU software is performed to investigate the relationship between RSSI and the distance for a point-to-point communication when different configurations are introduced. Figure 9 shows the screenshots of range test in X-CTU, it provides the RSSI indication bar in dBm where -40 dBm represents the strongest signal received by the module and -104 dBm is the weakest. The range test procedure was carried out as follows. The coordinator module is connected to PC and sends a packet of data to a remote module. Each of the coordinator and remote module are installed on a XBIB-U-DEV kit, as it has a feature to perform a loop-back test as shown beforehand in Figure 7. The remote module will send the received packet directly back to the coordinator module and the X-CTU estimates the value of the RSSIbased on the last received packet by the coordinator module, and shows it in dBm [22]. “Packet delay”, “Data packets number” and “Data received timeout” are parameters by which a wide range of scenarios can be simulated, and also, the data size that is needed to be sent during the experiments can be controlled using “Create data” tab.



☑ Stop at 10
□ Stop on error
Data receive timeout 1000 msec
Percent 100.0
-40
Good 10
Bad 0
24576 bytes
COM9 19200 8-N-1 FLOW:NONE

```
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?
$ABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789:;<=>?$ABCDE
```

Figure 9. Range test screenshot

4.1 Test Scenario

There are many parameters that affect the power consumed by the XBee modules. Operation modes, such as transmitting and receiving, draw different amount of currents, and therefore their power consumption is different. However, there are five power levels at which the XBee module can transmit the information, and they correspond to -8 dbm (the lowest), -4 dbm, -2 dbm, 0 dbm and 2 dbm (the highest) respectively, and by increasing the power level the consumed current will increase. There is no problem with power consumption in the base station node as it will be powered from PC in the practical conditions, but the consumed power in sensor node needs to be optimised. The point here is to carry out a test to measure the RSSI considering the effect of transmit power levels with existing of RPSMA antenna on both the sensor node and base station. This experimentwas carried out in an indoor environment and the distance between the two modules starts at 5m and is then increased in 5m steps until the wireless connectivity was lost. Throughout the experiments, a 24 KB data size (typical size of the measured vibration signals) was created by the X-CTU software and transmitted by the coordinator to the sensor node ten times; every time the RSSI was measured and then the averaged RSSI value computed and used for comparison. The power level can be set through the configuration modem tab of the X-CTU, and the nodes can be configured to communicate at different baud rates ranging from the lowest 1200 bps to the highest 230400 bps. In this test the modules were set to communicate at 19200 bps baud rate, since experimentally it was established that it is the highest one that can be used with the specified data size without loss of information.

The results are presented in Figure 10, and it can be observed that the measured RSSI values decreased as the distance is increased, due to the depletion of the wave energy as it propagates longer. The fluctuations in the RSSI values can be correlated with the presence of reflection and multipath phenomena because of the walls and interference from Wi-Fi routers located in the building [22]. Also, it is apparent that increasing the transmitting power has improvedthe transmit performance, but that will increase the power consumption level. Fortunately, it was indicated that all transmitted packets of data were received back by the coordinator when the sensor node transmit power was in the medium level; and consequently the device will be set to send the data at this power level. As stated in the XBee s2 datasheet and shown previously in Table 3 that the indoor range is approximately 40m, which wasfound to be the case according to the results obtained from the experiment. The transmitted data packets were fully received up to 40m with RSPMA antenna and medium power level, whereas the RSSI and data packets drop significantly when the distance is increased beyond thislimit.

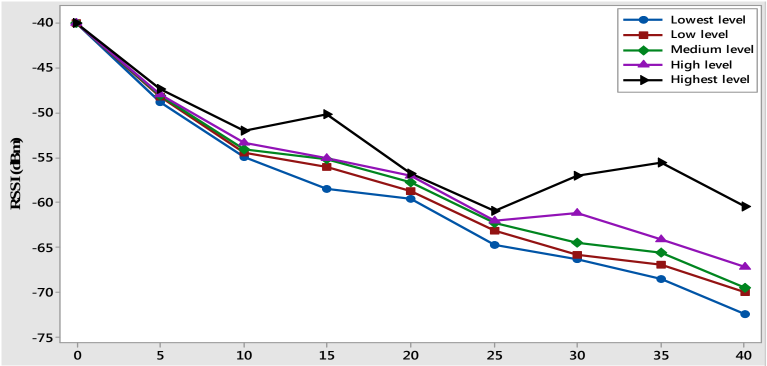


Figure 10. Measured RSSI values versus distance at different transmit power levels

5. SYSTEM SET-UP AND TESTING

To protect the wireless node and base station from dust and other contaminants, enclosures have to be used. Available enclosure for the Arduino DUE is designed to enclose just the boards without the other connected peripherals such as the sensors, battery, XBee shield and the antenna; thus, it was necessary to look for alternatives. Metal or plastic boxes are commonly used for creating custom-built enclosures for embedded equipment. For this purpose, two plastic boxes of size 179 x 86 x 51 mm, for the sensor node and base station were purchased and modified. Bolts were used to fix the Arduino microcontrollers to the boxes. Small rectangle and circular cuts were made on the base and right side of the sensor node box, to allow attaching the sensors adapter to the robot using super glue, and the antenna, via the wireless shield, and power supply port, from the battery, to the microcontroller. A double sided adhesive tape at the bottom of the battery was utilized to affix it on the box. The base station should be located next to a host PC, for reporting the analysis results and also to permit control of the entire monitoring system through the user interface. The sensor node was attached to the robot using two cable ties. The sensornode along with the final experimental are depicted in Figure 11.

Prior to deploying the designed system for robot condition monitoring, system accuracywas tested and validated. The testing was performed by setting up the system for real-time robot vibration analysis and the captured signalswere compared with their peer that were acquiredusing NI 6009 DAQ device (from National Instruments). Significant differences between signals are not preferred, as it could lead to increase rate of false fault detection. The result of the healthy signals comparisons for each of the three axes of measurements is presented in the Figure 12. Signals from both NI DAQ and the sensor node are compared with the robot executing a pick and place task, using the same sampling frequency (383 Hz); the time-domain signals are associated with their probability distributions. It can be seen that the wireless system signals are identical with NI DAQ signals, and normally distributed. It can be observed that the signals from the designed system are having slightly higher amplitudes, also the other statistical features, i.e. the standard deviation (STD), root mean square (RMS) and maximum and minimum amplitudes, as shown in Table 4. This could be attributed to attaching the sensor node to the robot structure, which cannot be avoided, making the system capturing extra vibration from the robot, while this is not the case when the NI DAQ. Nevertheless, it was concluded that the signals captured using the Arduino DUE board have very reasonable accuracy, and thus it is expected that by employing the DUE board the desired credibility level of the designed robot health monitoring system can be achieved.

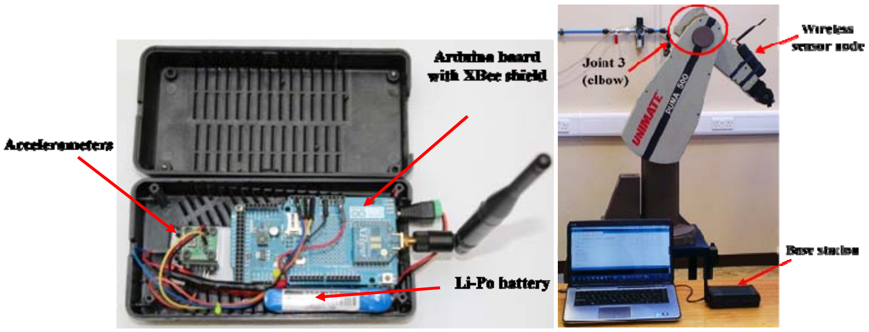
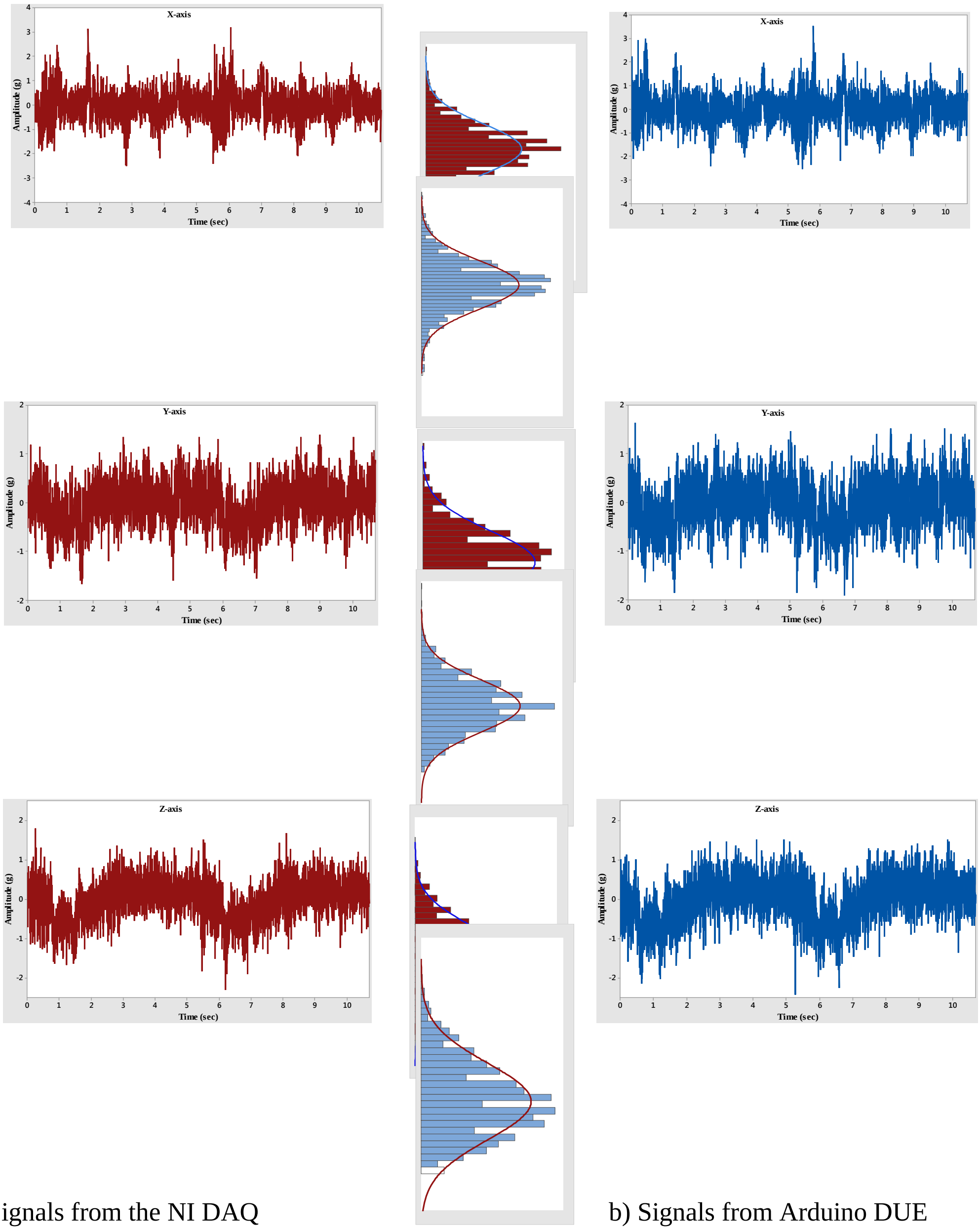


Figure 11. Wireless sensor node hardware and the experimental set-up



a) Signals from the NI DAQ

b) Signals from Arduino DUE

Figure 12. Time-domain signals from the X, Y and Z accelerometers captured using NI 6009 DAQ and Arduino DUE board

Table 4. Statistical features extracted from time-domain signals captured using NI 6009 DAQ and Arduino DUE board

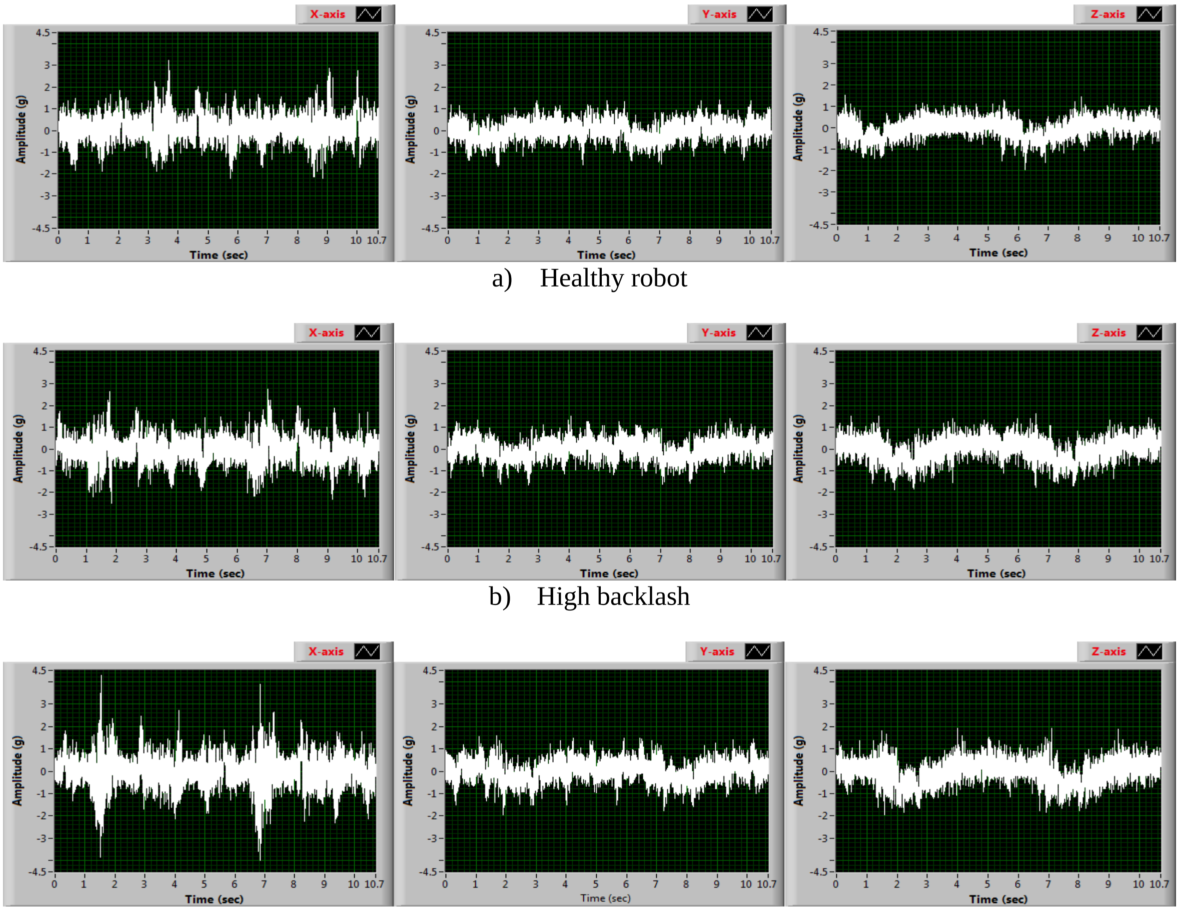
Axis of measurement	Used signal acquisition system	STD (g)	RMS (g)	Maximum (g)	Minimum (g)
X	**NI DAQ**	0.62613	0.626	3.18387	-2.50729
	Arduino DUE	0.6636	0.664	3.5530	-2.5378
Y	**NI DAQ**	0.45126	0.448	1.39296	-1.67155
	Arduino DUE	0.50545	0.506	1.63550	-1.91748
Z	**NI DAQ**	0.52054	0.521	1.79090	-2.30827
	Arduino DUE	0.57752	0.578	1.52271	-2.42505

6. RESULTS AND ANALYSIS

For the purpose of fault detection on the robot, backlashin the robot's gears has been introduced to simulate afault in the robot. Joint 3 (Figure 11), which is defined as the elbow joint, was chosen to study the backlash fault detection. The transmission systemof joint 3 is composed of two stage gearbox. And to introduce backlash in this joint, there are number of screws should be rotated to adjust the pitch between the gearspairs. After adjusting the backlash, the robot was programmed to mimic a standard robot task by undertaking a simple pick and place sequence. In the first instant, the robot was tested when backlash level fine-tuned to be as minimum as possible (healthy robot), and then increased and decreased excessively to simulate the fault conditions on the robot. The vibration signals from the robot with the backlash faults introduced are shown in Figure 13. The figure presents the time-domain vibration signals from the robot, when it is healthy and with the two backlash levels (high and interference). However, always vibration of multi-stage gearboxes, as in the robotic system, is very complex and composed of different frequencies, including gear mesh, bearings, and running-speed frequencies. The vibration severity is also connected with the excited resonance frequencies of the robot. The high speed of the robot gears causes cyclic excitation of these resonance, leading to some periodic fluctuations with amplitude proportional to the fault severity. This can clearly be seen in X-axis signals (the first column in Figure 13), which were influenced the most by backlash, while signals in the other two axes were less affected. The high amplitude components present in the signals were observed when the robot changes the rotating direction of the joints. This is leading to developing number of impact between the mating pairs, due to the contact force between teeth. In the interference case the backlash was completely removed, so the impact effect is eliminated between mating gears but the gears are overloaded on the other hand.

It was anticipated that by increasing the backlash the vibration level will increase in the robot, but the opposite has been found. This can obviously be noted by comparing the vibration signals produced with different backlash levels to the healthy robot condition specially X-axis signals. It can be seen, for instance, in the case of higher backlash the amplitudes of vibration signals are lower than that of the healthy case; this also corresponds with the results of a research was published byBicker, et al. [23]. The amplitudes of vibration are increased when the backlash level is reduced, until the increase becomes more significant in the interference case. Although the presence of backlash in the transmission system can cause a transient impact at the reversal of motion, which will lead to an undesirable level of vibration, the high backlash between gears will allow more lubricating grease to enter between mating teeth leading to damping the vibration. Whereas, the tight gear mesh leads to squeeze the lubricant out of the mating teeth and heat the system up, due to friction between teeth, resulting in increasing the vibration level.

However, the investigated faults were mainly related to the elbow joint transmission. Other categories of faults related to this joint motor, such as motor bearings, stator and rotor faults, or faults in the other joints of the robot form a logical direction to further extend the capabilities of the designed embedded-based health monitoring system. Also, the examined faults scenarios were simple, as only one fault is seeded at a time; however complex scenarios could be studied by combining multiple fault types at the same time. Moreover, in the present study, accelerated fault types were seeded in the joint gearbox. For future research, the algorithm could be extended to monitor the progressive development of faults in the robot transmission. This can be achieved by allowing the robot to operate for a considerable time period and the vibration level is monitored. Otherwise, a control device that can be connected to the backlash adjustment grub-screws and allow the progressive increase or decrease of the gear backlash while the robot is running could be developed.



a) Healthy robot

b) High backlash

c) Interference backlash

Figure 13. Vibration signals from the accelerometers at different backlash level in the robot gearbox

7. CONCLUSION

This paper has detailed the design approach and hardware selection requirements for developing a wireless vibration monitoring system that can be utilized for machines health monitoring. The Arduino DUE was selected since it was appropriate as a core of the wireless sensor unit. The wireless capability was put on the DUE board by using Arduino wireless shield. A comparison between the existing wireless technologies like Wi-Fi, Bluetooth and ZigBee was presented and it was established that the ZigBee wireless transmission consumes less power and is very reliable when it operates within the recommended range. The XBee wireless module, which is based on ZigBee protocol, was employed to establish the communication between the sensor node and the base station. Configuration of the XBee transceivers for both sensor node and base station was crucial in the implementation of wireless networks. The configuration method of the XBee transceivers in addition to several experiments to test the transmission range and to select the optimum transmitting power on the sensor node were carried out. Enclosures for the sensor node and base station hardware were developed relying on commercially available boxes, and then the final system installation and testing illustrated.In order to demonstrate the validation of the developed wireless embedded system, captured vibration data were compared to their peers that were captured from a healthy robot using NI 6009 DAQ. Then, the utilized robot was programed to execute a handling task that mimics one of its real tasks and with different backlash faults seeded inside. Vibration signals from the three axes of measurements were captured from the robot while it is performing this task. The captured signals have showed very good sensitivity to the small changes in the backlash level, and thus the system can be effectively used for fault detection via vibration signal analysis. However, the designed system needs more development that could make it more intelligent. Also, only one robot was monitored throughout this study, therefore, the hardware and software could be developed to accommodate monitoring of more than a robot. This can be achievable by building and using more than one wireless sensor node and updating the designed embedded software to be able to communicate with more than one sensor node.

REFERENCES

[1] A.P. Shukla, H. Garg, G. Varshneya, and A.K. Srivastava, "Real time acquisition of vehicle diagnostic data using wireless sensor network," 2009, pp. 89-93.

[2] B. Wu, J. Lin, and X. Xiong, "Design and implementation of intelligent monitoring and diagnosis system based on WSN and MAS," vol. 238 CCIS, pp. 290-297, 2011.

[3] M.F. Rad and L. Shafai, "A wireless embedded sensor for structural health monitoring applications," in *Antenna Technology and Applied Electromagnetics and the Canadian Radio Science Meeting, 2009. ANTEM/URSI 2009. 13th International Symposium on*, 2009, pp. 1-4.

[4] S.G. Taylor, K.M. Farinholt, G. Park, C.R. Farrar, and M.D. Todd, "Application of a wireless sensor node to health monitoring of operational wind turbine blades," 2011, pp. 45-53.

[5] M.E. Elnady, J.K. Sinha, and S. O. Oyadiji, "On-Shaft Wireless Vibration Measurement for Condition Monitoring of Rotating Machine," presented at the International Conference on Vibration Problems Prague, 2011.

[6] G. Qin and N. Hu, "Design of embedded wireless sensor and its soft encapsulation for embedded monitoring of helicopter planetary gear set," *Journal of Physics: Conference Series*, vol. 364, 2012.

[7] G. J. Feng, J. Gu, D. Zhen, M. Aliwan, F.S. Gu, and A.D. Ball, "Implementation of envelope analysis on a wireless condition monitoring system for bearing fault diagnosis," *International Journal of Automation and Computing*, vol. 12, pp. 14-24, 2015.

[8] N.K. Verma, S. Sarkar, S. Dixit, R.K. Sevakula, and A. Salour, "Android app for intelligent CBM," in *IEEE International Symposium on Industrial Electronics*, 2013.

[9] W. Wanbin and P.W. Tse, "Remote machine monitoring through mobile phone, smartphone or pda," in *Proceedings of the 1st World Congress on Engineering Asset Management, WCEAM 2006*, 2006, pp. 309-315.

[10] K.Y. Lian, S.J. Hsiao, and W.T. Sung, "Mobile monitoring and embedded control system for factory environment," *Sensors (Switzerland)*, vol. 13, pp. 17379-17413, 2013.

[11] S.H. Yang. (2014). *Wireless Sensor Networks: Principles, Design and Applications.*

[12] J.K. Sinha, *Vibration Analysis,Instruments, and Signal Processing*, First Edition ed.: CRC Press, 2014.

[13] A. Albarbar, S. Mekid, A. Starr, and R. Pietruszkiewicz, "Suitability of MEMS accelerometers for condition monitoring: An experimental study," *Sensors*, vol. 8, pp. 784-799, 2008.

[14] M.D.P. Emilio, *Embedded Systems Design for High-Speed Data Acquisition and Control*: Springer 2015.

[15] H.H. Yan and Y. Rahayu, "Design and Development of Gas Leakage Monitoring System Using Arduino and ZigBee," in *Proceeding of International Conference on Electrical Engineering, Computer Science and Informatics (EECSI 2014)*, Yogyakarta, Indonesia, 2014, pp. 207-212.

[16] C. Townsend, S. Arms, and MicroStrain, "Wireless Sensor Networks:Principles and Applications," in *Sensor Technology Handbook*. vol. 1, J.S. Wilson, Ed., ed: Elsevier 2005.

[17] S. Giannoulis, C. Koulamas, C. Emmanouilidis, P. Pistofidis, and D. Karampatzakis, "Wireless sensor network technologies for condition monitoring of industrial assets," in *IFIP Advances in Information and Communication Technology* vol. 398, ed, 2013, pp. 33-40.

[18] V. Bhandari and P. Abrol, "Field Monitoring of Treated Industrial Waste Water," *International Journal of Electrical and Computer Engineering*, vol. 4, pp. 237-242, 2014.

[19] R. Faludi, *Building Wireless Sensor Networks*: O'Reilly Media, Inc, 2011.

[20] I. Buchmann. (2003). *What's the Best Battery?*

[21] W.W. Dargie and C. Poellabauer, *Fundamentals of wireless sensor networks: theory and practice*: John Wiley & Sons, 2010.

[22] R. Piyare and S.R. Lee, "Performance Analysis of XBee ZB Module Based Wireless Sensor Networks," *International Journal of Scientific & Engineering Research*, vol. 4, pp. 1615-1621, 2013.

[23] R. Bicker, A. Daadbin, and J. Rosinski, "The monitoring of vibration in industrial robots," in *ASME 12th Biennial Conference on Mechanical Vibration and Noise*, 1989.