Wireless Sensor Networks for In situ Energy Efficiency Monitoring

Joe Cowlyn, Xuewu Dai, Ghanim Putrus
Faculty of Engineering and Environment, Northumbria University, Newcastle upon Tyne, NE3 5JL, UK
joe.cowlyn@northumbria.ac.uk

Abstract— **Energy efficiency in manufacturing is critical for a reduction of overall energy consumption and also better demand response (DR) to offset non-essential peak energy against rising cost of energy. Motivated by our industrial partner's demands on precise and timely information about the energy usage, this paper presents the design and testing of a wireless sensor network (WSN) for collecting electrical energy consumption and power quality monitoring within an industrial environment. The proposed network includes a set of in situ wireless nodes each connected to a production line and a data sink located in a control room for data collection, data visualization and user interaction. The features of this system include the ability to real time monitor energy consumption, low cost and cable-free deployment of multiple monitor nodes to cause no additional hazards in the production environment. This system enables that production line energy consumption can be monitored effectively using inexpensive sensors and data can be acquired, without disruption of production, in order to determine energy usage between different stages and lines of production and more accurately grade customer pricing per part produced.**

Keywords—wireless sensor network; smart meter; power quality; industrial; data collection

I. Introduction

Power usage and efficiency is important to consider in manufacturing. A large quantity of power is required daily in order to produce parts at rate suitable for large scale production and how this power is used and distributed between different manufacturing lines and the cost per part calculated is of large importance to part producers and customers. The rising cost of energy is also an important factor in desiring the ability to know exactly how much is being used at each production stage and by each piece of equipment during production. This can be used in order to reduce the amount of energy consumed, and therefore cost, as much as possible. Power quality (PQ) is also important due to the rapid growth in the use of equipment that generates PQ disturbances and also an increase of equipment that are sensitive to these disturbances. This may cause damage to equipment over long term exposure to fluctuations in power quality and even may cause halting of production due to power outages or requirements to replace damaged equipment. These concerns on PQ have increased the demand for advanced PQ monitoring systems. Power quality monitoring is to detect unwanted disturbances of the electricity supply by monitoring the voltage/current waveform and analyse its features. The information provided by the energy monitoring system at each individual production line is vital to be able to determine the energy usage at different stages of production and will be used to improve the way machines are managed or operated during a production cycle and attempt to reduce non-essential peak energy use and monitor power quality.

Wireless sensor network technology has demonstrated a great potential for consumer and commercial applications. It is an attractive prospect for industrial applications due to its cable-free deployment and maintenance. The 802.15.4 standard [1] is mainly used to allow for the creation of wireless personal area networks (WPAN), which are used to convey information over short distances. This is the best solution as it allows for small, power efficient and inexpensive networks of sensors to be implemented and maintained easily.

This area of research has been previously carried out in a home environment as described in [2] so energy use of electrical appliances could be monitored for users to adapt their energy usage habits and reduce energy costs. This is tailored to a home environment and does not really detail how it could be applied to other areas such as industrial.

The application of WSN in an industrial environment was explored in [3], where wireless sensors are tested for gas turbine engine testing with real-time and safety requirements.

The co-existence of IEEE 802.15.4 with other 2.4GHz wireless devices (e.g. IEEE 802.11) was investigated in [4] and some solutions were proposed, such as channel choices and setup distance from existing wireless routers, when setting up the proposed sensor network.

A similar energy monitoring project is described in [5] which with focus on collecting power quality data from the Turkish Electricity Transmission System. It defines a set of power quality events that can be reported to a central database and a data model for storing the events for future evaluation. It also presents a Graphical User Interface (GUI) for visualisation of the power quality data.

In order to apply the idea of a sensor network to an industrial manufacturing environment this paper proposes a wireless sensor network (WSN). The reason why a wireless network is best for an industrial environment is it allows for easy deployment due to the low cost of each node and ability to put nodes inside an existing device or junction box without external modification. There is a reduction of required cabling between sensors and data collection devices when using a WSN, which also reduces safety risks associated with additional cables over a workshop. The network will also allow for real-time monitoring of production as it will continually report power and part count data to the base station as it is collected, which can be displayed instantly. Another feature is the ability to connect a large number of nodes to the network as they do not need to be physically connected to each other in order to attach to an existing network and in-built features such as Personal Area Networks (PANs) can allow for more than one network to exist simultaneously. This can be used to collect the same data in a different node group without interference or for allowing other networks collecting different data to also coincide.

The following discussion will be organised into five parts: Section 2 will elaborate on the proposed network architecture, Section 3 will define the hardware used in setting up the system, Section 4 will detail the process of the software used to run the network and data acquisition, Section 5 will describe tests carried out on an implementation of the network and present an evaluation of results and Section 6 will contain a summary of conclusions and suggested future work.

II. Network Achitecture

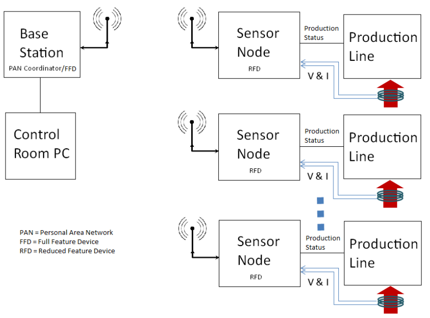

Fig. 1. Diagram of wireless sensor network

The proposed wireless sensor network (WSN), shown in Figure 1, consists of a set of wireless sensor nodes each installed at a different production line in the workshop, a base station and a computer located in a remote control room. The wireless sensor node not only monitors the power consumption by measuring the three-phase voltages and currents supplied to the production line, but also takes the digital information provided from the production line indicating its status (e.g. the number of products that have been produced). An IEEE 802.15.4 compatible radio module is also integrated into the wireless node for wireless communication. The base station consists of an embedded wireless device and microprocessor attached via USB COM port to a control room computer. This computer runs software to trigger reading of the sensor nodes at a set interval and record the results. It also includes a web server to access the recorded data in a webpage graphical user interface (GUI).

The WSN uses the IEEE 802.15.4 MAC protocol [1] in order to communicate between nodes. The base station acts as the PAN coordinator and each sensor node can either operate as a Full Featured Device (FFD) and act as a coordinator if required or as a Reduced Feature Device (RFD) and just send and receive its own data. Each 127 byte IEEE 802.15.4 frame must be correctly formatted with a 2 byte Frame Control Field which defines what data is included in the frame. For each frame sent by a sensor node the Frame Control Field must define the frame as being a Data frame, and choose the correct addressing mode for the current requirement, usually 2 byte short addresses. The PAN address must also be set in order for the address filter to correctly filter only frames for the intended recipient to be read by the node. Each frame also includes a 2 byte Frame Check Sequence which can be set and checked automatically in the AT86RF233 hardware [6] by setting register bit TX_AUTO_CRC_ON of the TRX_CTRL_1 register to 1 and checking register bit RX_CRC_VALID of the PHY_RSSI register is 1 upon receipt of a frame. The header of the MAC frame can range between 9 and 25 bytes depending on the frame control field settings which, including the FCS, leaves 100-116 bytes available for data transmission per frame.

IEEE 802.15.4 was chosen instead of a more featured network standard such as ZigBee [7] because it provides all the necessary features to both enable sending of data packets between nodes and allows for management of the network via MAC commands. ZigBee provides an application and network stack with a series of profiles that define what sort of data is to be sent over the network and has a large number of different versions of the standard with different supported features including ZigBee, ZigBee Pro and ZigBee Green Energy. This WSN is only intended to send power data and the format is known at all the nodes beforehand, this allows for a much less complicated and faster network to be implemented directly on top of the IEEE 802.15.4 standard.

At the base station, a graphical user interface (GUI) will be developed for user interaction and data visualization. All the data collected through the WSN will be saved in the database at the base station and presented to the end user in a graphic way for easy understanding, such as the trend of energy consumption, statistical charts and so on. Both live and history data can be retrieved. For remote access anywhere anytime, it is preferred that these function are provided through a secure webserver.

III. Hardware Development

The sensor nodes actually consist of two parts: an ARM microprocessor with wireless unit for data acquisition, processing and transmission, and a conditioning board (or existing smart meter) for voltage and current measurements. The production line status is provided by the devices to be monitored and connected to the ARM microprocessor via digital signals. Some digital and analog ports of the microprocessor are reserved for future use.

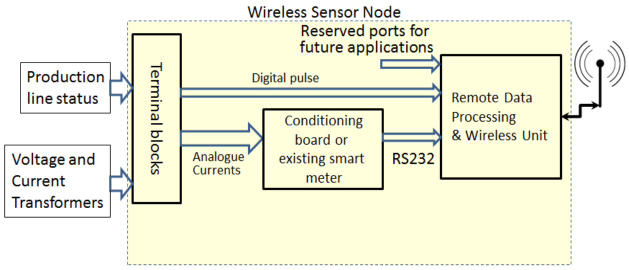

Fig. 2. System diagram of Power Monitor system.

Figure 2 above shows the layout of the intended system. It shows the two parts of the wireless sensor node and how they are connected to each other and the production line as previously described. Details about the hardware design involved in the system are described below.

A. Voltage/Current Measurement

As a prototype development, since our focus is the development of a wireless system to enable wireless data acquisition, a Mk10E smart meter is used to provide the voltage and current measurements. The Mk10E manufactured by EDMI is a 3-phase smart meter with in-built current transformer (CT), voltage range of 52-290V, frequency range of 45-65Hz, multiple communication ports (Optical, RS232, RJ45 Ethernet [8]) and a configurable LCD display. The meter can be configured using the EDMI EziView software as described in [9] via optical port to change any of the supported values, such as to allow for an external CT or to change the communication port protocols.

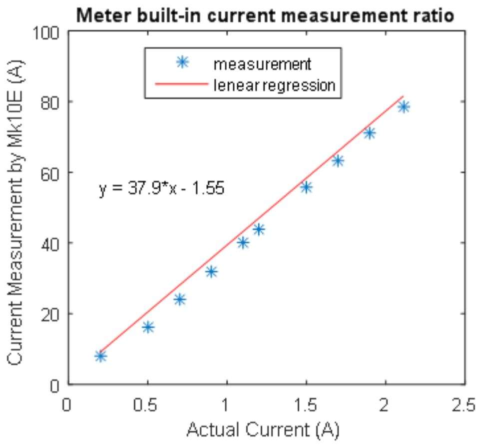

Fig. 3. Graph of meter built-in current measurement

It was not specified what the internal CT ratio was rated as so a test was conducted in order to accurately determine the actual ratio used. A series of measurements were taken by the smart meter and by a hand held current meter (ICM 33II) on the Phase A current between 0 and 2.5A and compared in Figure 3.

The x axis shows the actual current as detected by the current meter and the y axis shows the current value reported by the meter. Linear regression is applied to approximate the current conversion ratio and the result shows that the current conversion ratio is about 37.9 and has an offset of -1.55. The small non-zero offset could be caused by the measurement errors of the handheld current meter, so can be ignored.

The MK10E can use a range of protocols to access the internal registers such as IEC 62056-21 and UDP but the simplest means of accessing it is using the proprietary EDMI Command Line Protocol (CLP) [10]. This protocol defines a series of commands that can be issued to the smart meter in order to configure or extract data from it. Each command consists of a series of hexadecimal bytes that define the functionality. Commands are framed with two bytes starting with the value of 02 defined as <STX> and ends with the value 03 defined as <ETX>. Commands must be escaped against internal values of <STX>, <ETX> and of the escape values 10 and 13. This is done by placing a value of 10 before any value that must be escaped and then adding the value 40, therefore a value of 02 becomes 1042. The command must have the escape values returned to the original form before CRC checks are performed and commands processed. Each command also contains a CCITT 16 bit CRC (cyclic redundancy check) defined as <CRC> which can be used to check for data integrity. There are then a set of supported commands such as logging into the meter with the value of 4A (ASCII 'L') or reading a register with the value 52 (ASCII 'R'). The commands have their own data specified in [10] which define what needs to be sent with the command in order to perform it, such as a 2 byte address value required for a register read or write command. There is also an Extended Command Line feature that allows for a specified destination and source address to be defined and includes a frame number to match requests with responses, this is formed as a CLP command of value 45 (ASCII 'E'). An example command is shown below:

<STX>E<Destination Address><Source Address><Frame Number>R<Register Address><CRC><ETX>

This equates to a hexadecimal string of 02450CE472A742C064E8000052F530ACA203.

This command is to read the F530 special register which reports a collection of registers contained in the meter as a single access. All bytes in the commands are in big endian (most significant byte) order. The first two bytes are the frame start and extended command values, the next 4 bytes are the address of the meter that is to be queried. This is a useful feature as it could allow multiple meters to be attached to the same sensor node or to allow for network PAN controller features of the IEEE 802.15.4 specification to be used and know the destination sensor without additional addressing requirements. The next 4 bytes are the source address of the command which in the example uses the value used by all tested existing software. The next two bytes are the frame number of the command. This is then followed by a register read command byte and the address specified of F530. Finally there is a 2 byte CRC and the frame end byte.

The returned response is very similar to the command sent, with the register address followed by the requested data bytes which are of a size specific to the requested register as defined in [10] and ends with a CRC and end frame value as normal.

This command structure can be easily used over a WSN as it has a very small footprint, of a maximum of 207 bytes from the example command's response and a general size of around 20 bytes. The in-built CRC can also be used to check data integrity and request retransmission if required and the extended command line features can be used to better direct commands over the network.

The software used for tests in this paper only use extended command line with register read commands.

B. *Enhanced IEEE 802.15.4 radio board*

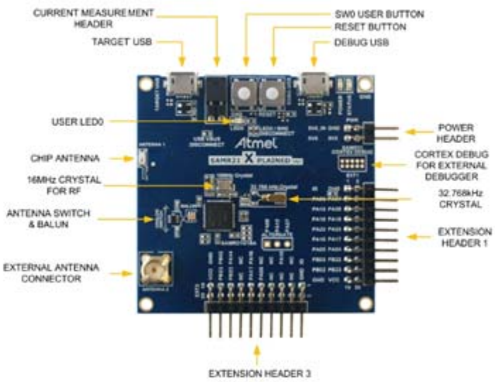

Fig. 4. SAM R21 Xplained Pro Evaluation Kit Overview [11]

The Atmel SAM R21 XPlained Pro microcontroller, shown in Figure 4, has a 32-bit ARM Cortex-M0+ processor, 32KB of SRAM, 256KB of flash memory and contains a built-in DSP. It includes an in-built SPI connected AT86RF233 module allowing for IEEE 802.15.4 communications with enhanced data rate of up to 2Mbps on the 2.4 GHz ISM band with a -99dBm RX sensitivity and maximum TX output power of +4dBm. In transmitting state, the radio board consumes 35.5 mW of power (or 26 mA of power at 1.8V) but less than 2 µW of power (or only 1.1µW of power at 1.8V) in power down (sleep) mode [6].

It has the ability to switch between its on-board ceramic antenna and an external antenna automatically depending on the best signal and can also be set manually if required. The RF module also includes a hardware accelerated automatic acknowledgement as an extended mode and an on-board AES encryption module and true random generator for IEEE 802.15.4 security requirements. There is also a watchdog timer available to ensure that the device does not become inactive due to software and will automatically reset the microcontroller if the timer is not reset at the required interval to ensure continuous operation.

Peripherals can be easily added to the R21 as there are 5 supported SERCOM ports which can be configured for UART, SPI and other communication protocols which are directly supported in hardware or using the 28 available general purpose I/O (GPIO) pins. Programs can be directly flashed to the microcontroller using the single wire debug (SWD) interface over the debug USB cable using the Atmel Studio software [12]. This also supports using the same USB cable for power and as a serial COM port interface with a PC.

This is a very good device for a WSN as the wireless data packet is compatible with the standard IEEE 802.15.4 data frame format. It also allows for addresses and PANs to be easily filtered with the support of the hardware accelerated address filter which makes setting up a WSN much less complicated.

IV. Software Development

This section describes the software architecture of the proposed embedded systems for both sensor nodes and base station. The proposed system is developed on the mbed internet of things software framework [13] and integrated with FreeRTOS [14] for time-sensitive data acquisition and processing. The features of the proposed system are: collection of power consumption data and production line status via wireless sensor node, aggregation of data in a database system and display of data in a web enabled graphical user interface.

A. *Software Architecture*

A decision was made to use the mbed API [13] for the R21 programs as it allows for fast setup of the microcontroller and easy access to peripherals, only requiring the pin values from the data sheet to configure them.

The mbed library consists of a series of layers as shown in Figure 5, which allows for abstraction of user code from the underlying hardware. The components mainly used in the program from the mbed library are the mbed API and CMSIS-Core for direct control of the microcontroller register values, for clock and antenna configuration.

The CMSIS-Core standardises the names of the underlying microcontroller registers allowing easier access using value names as defined in the datasheet instead of memory mapped RAM address values, which makes it a lot simpler to do low level configuration.

A RTOS is a real-time operating system designed for microcontrollers. FreeRTOS [14] was chosen as it is open source with very good hardware support for the chosen microcontroller. It is a very simple implementation requiring only 3 C source files and compiles to a very small binary image of 6-12Kb. The reason a RTOS is required for this project is the ability to allow for multiple "threads" of operation to occur seemingly simultaneously. This allows all the processing required, such as peripheral access and data processing, to happen without blocking the operation of another part of the program allowing for much more reliable software and time sensitive operations to be correctly managed.

B. Software Functionality

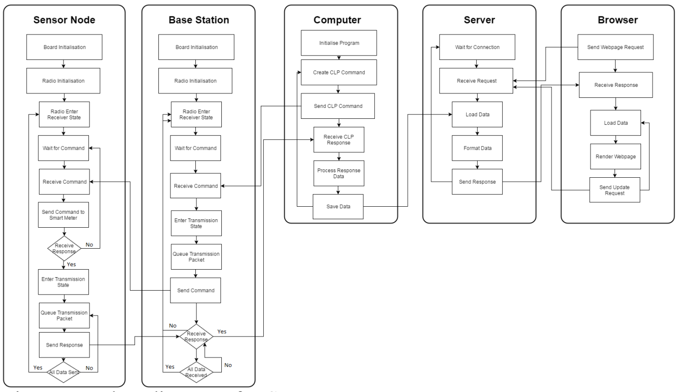

Fig. 5. Flow diagram of WSN programs

Figure 5 shows the control flow of each program in the system and how each interacts with the others. A data collection cycle is initiated by a program on the computer connected to the base station.

The program first constructs a CLP command to request a register value and then, at a set interval, it sends the command to the base station. Each time the base station receives a command over the USB serial connection, it will redirect this command to the radio by buffering it and placing it into an IEEE 802.15.4 frame [1]. The frame is then sent to the radio module over SPI and the transmission state enabled. Once the frame is sent the radio will be set back to receiving state. The sensor node will then receive the frame and check the CRC is correct (as discussed in Section 2). If it has been transmitted over the WSN correctly, the command will then be extracted from the frame payload and sent to the Mk10E smart meter via UART to the meter's optical port.

The meter will then process the command and if correct, will return a response packet. This response is then sent back to the sensor node (as before). This response is then put into an IEEE 802.15.4 frame. If the returned data is larger than the allotted frame payload size of 100-116 bytes, it will be divided up and send in separate packets. The same procedure is followed to send a packet from the sensor node as from the base station. Once the frame(s) arrive at the base station the response is buffered and if larger than a single packet will be reformed into a complete CLP response. This response is then returned to the computer over the USB cable. The program on the computer will then record the response data in a comma separated value (CSV) file.

This CSV file can then be read by any program that is able to understand it such as Excel or MatLab and the data retrieved. A user can then use a web browser to send an HTTP request to the web server hosted on the computer, to retrieve either live or historical data. This request will start a connection on the server, and the server will respond with a webpage containing the required data. An example of historical data displayed in a webpage as part of the proposed system is shown in Figure 6.

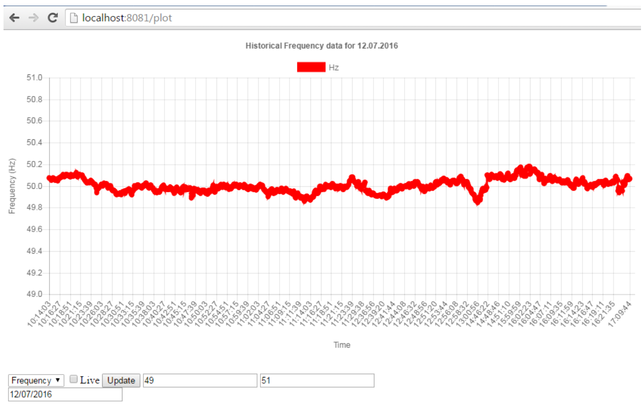

Fig 6. Historical meter data displayed in a webpage from csv data file

V. Results

In order to test the viability of the proposed network tests were carried out on two factors: A. the ability for the WSN to sample data continuously for a sustained period of time and B. the packet loss rate of the WSN.

A. Sustained Data Collection Test

A network meter data collection test was conducted between 07/07/16 18:19:46 and 08/07/16 12:02:02. The results show that the programs and hardware are capable of extended operation. The setup used an isolated current transformer, variable resistance, two Atmel R21 Xplained Pro microcontrollers running as data relays, an ICM 33II current meter, an EDMI Atlas Mk10E smart meter and a PC running data collection program (as described in Section 4).

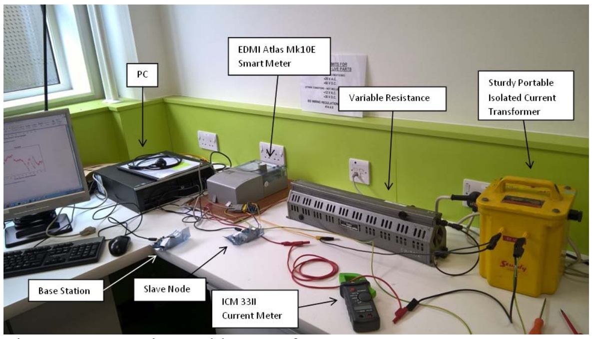

Fig 7. Experimental layout of meter test

The layout of these devices is shown in Figure 7. Only the phase C voltage, phase A current, frequency and applicable derived values, such as phase angle, were collected.

The meter register F530 was sampled every 4 seconds, with up to 3 retries if no response was received within 3 seconds for each retry.

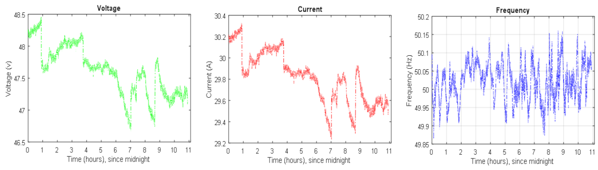

Fig 8. Graphs of Voltage, Current and Frequency data collected on 08/07/16

Figure 8 shows the collected results and that the test successfully demonstrated the WSN was capable of sustained operation over an 18 hour period.

B. Packet Loss Test

To test the packet loss rate of the WSN, a series of tests were conducted. Each test consisted of a laptop running a packet test program and two R21 nodes, one operating as base station connected to the laptop via USB and the other as sensor node attached to a power source (PC or battery pack) via USB.

The packet test program, once initiated, sent 100 packets of 100 bytes in 4 second intervals and recorded the bytes received back from the sensor node if received within 4 seconds. If no packet was received within 4 seconds, the packet was considered lost. The base station node relayed the packet from the laptop to the sensor node and the sensor node immediately sent it back upon reception.

Each test placed the base station in a defined position and measured a distance away from it to place the sensor node. The tests were performed at set distances of 10cm, 1m, 2m, 2.5m, 7m and 9m. Line of sight (LoS) was noted for each test. At each distance the power level was adjusted by setting the PHY_TX_PWR register of the AT86RF233 module on the base station node to 0, 3, 6, C and F for power levels of -17, -6, 0, +3 and +4 dBm respectively [6].

Packet loss rate is defined as the total number of packets lost over the total number of packets sent, in this case 100. The validity of returned data was also checked and any packets with incorrect bytes, checked against what was sent, were considered lost.

Figure 9 shows that there is an overall low packet loss rate of maximum 5% at the lowest power setting of -17dBm at a distance of 2.5m was recorded. This is possibly due to interference from a nearby 802.11 router [4] and the channel selected of 12 (as discussed in Section 1). The best recorded power level seems to be 0dBm as Figure 10 shows a packet loss of 0% for all distances tested.

Overall the test show that a close range with LoS and low interference from walls, routers or other factors is best, but operation with these factors included has an acceptable packet loss rate. This loss rate could also be further reduced by addition of acknowledgment and retries.

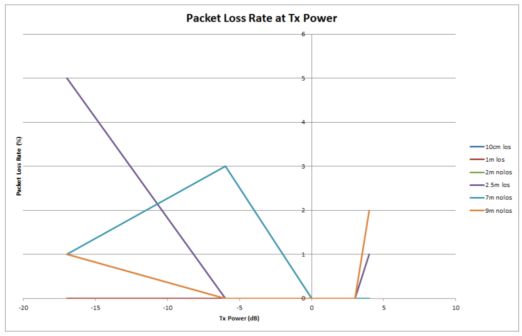

Fig 9. Graph of packet loss rate at defined Tx power ranges at set distances

VI. Conclusions and Future Work

This paper presents a wireless sensor network with reliable data acquisition and communication for electricity power consumption monitoring of a manufacturing line. The developed sensor node provides a low cost and easily deployable real time monitoring solution for power data collection and a base station system for storing and displaying the collected data. A series of proof of concept test have been done and results of sustained data collection and of packet loss measurement presented, which show that the system is viable for the intended purpose and sufficiently robust to allow for extended use and accurate data transmission.

In the future, a complete implementation of the IEEE 802.15.4 MAC layer could be explored in order to allow for use of a beacon enabled network and enhanced collision avoidance. This would allow for large numbers of nodes to be attached to a network which could then be tested with multiple production lines. Another future consideration would be to use a high resolution DSP instead of a smart meter for data collection with high accuracy timestamps, which would also require accurate time synchronisation techniques to be applied to the WSN.

Acknowledgment

The authors would like to thank Northumbria University, Òscar García Castellote, Baptiste Morand, Petr Dobias, Jiwen Zhu and the authors' families for their help and support.

References

[1] *Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)*, IEEE Standard 802.15.4, 2011 [Online]. Available: http://standards.ieee.org/getieee802/download/802.15.4-2011.pdf. Accessed: Nov. 4, 2015.

[2] E. Chobot, D. Newby, R. Chandler, N. Abu-Mulaweh, C. Chen, and C. Pomalaza-Raez, "Design and implementation of a wireless sensor and Actuator network for energy measurement and control at home," International Journal of Embedded Systems and Applications, vol. 3, no. 1, pp. 1–15, Mar. 2013.

[3] X. Dai *et al.*, "Development and validation of a simulator for wireless data acquisition in gas turbine engine testing," *IET Wireless Sensor Systems*, vol. 3, no. 3, pp. 183–192, Sep. 2013.

[4] P. Yi, A. Iwayemi, and C. Zhou, "Developing ZigBee deployment guideline under WiFi interference for smart grid applications," *IEEE Transactions on Smart Grid*, vol. 2, no. 1, pp. 110–120, Mar. 2011.

[5] D. Küçük *et al.*, "PQStream: A Data Stream Architecture for Electrical Power Quality" 2007. [Online]. Available: http://arxiv.org/ftp/arxiv/papers/1504/1504.04750.pdf. Accessed: Aug. 17, 2016.

[8] "Atlas Series Energy Meters Mk10 / Mk7 Hardware Reference Manual Revision M1" 2010. [Online]. Available: http://www.camax.co.uk/downloads/EDMI-Atlas-Hardware-Reference-Manual-Revision-M.pdf. Accessed: May 7, 2016.

[9] "Atlas Series Energy Meters Mk10 / Mk7 Software Reference Manual Revision M", 2009. [Online]. Available: http://www.smartbuildingservices.com.au/docs/default-source/ProductPdfs/edmi-atlas-software-reference-manual-revision-m.pdf?sfvrsn=0. Accessed: May 7, 2016.

[10] *Atlas 1 Register Manual*, 1910-E-00, EDMI, Regent Centre, Newcastle upon Tyne, 2015.

[6] "Atmel SAM R21E / SAM R21G SMART ARM-Based Wireless Microcontroller DATASHEET," in *www.atmel.com*, 2015. [Online]. Available: http://www.atmel.com/Images/Atmel-42223–SAM-R21_Datasheet.pdf. Accessed: Nov. 4, 2015.

[7] ZigBee Alliance, ZigBee Specification: ZigBee Document 053474r17 2008

[11] "SMART ARM-based Microcontrollers SAM R21 Xplained Pro USER GUIDE," in *www.atmel.com*, 2016. [Online]. Available: http://www.atmel.com/images/atmel-42243-samr21-xplained-pro_user-guide.pdf. Accessed: Apr. 22, 2016.

[12] Atmel, San Jose, CA. (2016). *Atmel Studio*, version 7 [Online]. Available: http://www.atmel.com/Microsite/atmel-studio/. Accessed: May 25, 2016.

[13] S. Ford, "Development platform for devices," in *mbed.org*, 2016. [Online]. Available: https://developer.mbed.org. Accessed: Aug. 17, 2016.

[14] "Market leading RTOS (real time operating system) for embedded systems with Internet of things extensions,". [Online]. Available: http://www.freertos.org/. Accessed: Aug. 19, 2016.