Integration of Wireless Sensor Networks into Industrial Control Systems

T. Raza[1], W. Lang[1], R. Jedermann[1]

[1] Institute for Microsensors, -actors and –systems (IMSAS), University of Bremen, Germany
toqeer.iiee@gmail.com, wlang@imsas.uni-bremen.de, rjedermann@imsas.uni-bremen.de

Abstract In this paper, a prototype is developed, which can easily integrate a wireless sensor with programmable logic controllers (PLC) using ProfiNet. The low-cost embedded system Raspberry Pi was used to connect a wireless gateway with 'ProfiNet'. A Wizziboard wireless sensor node, running the Dash7 communication protocol, periodically sends temperature and humidity data. This data is received at the UART of Raspberry Pi. A modified Snap7 library is used for communication between Raspberry Pi and SIMATIC Manager. The data is transferred using the Ethernet bus. The prototype is tested on SIMATIC Manager with a PLCSIM simulator installed on a PC. The Siemens HMI software Wincc flexible is used to monitor the state of PLC systems. Here, Wincc flexible is connected to the PLC over the TCP/IP protocol. SIMATIC Manager and wincc flexible have been tested on a windows PC. Data logging is also implemented in the HMI software.

Keywords Wireless sensor, Profinet, Raspberry Pi gateway

Introduction

Logistic processes, e.g. production plants and warehouse / storage systems, need sensor information to improve their performance. Modern control systems have to react to dynamic changes of process parameters, e.g. different types or qualities of materials. Real time feedback enables a fast reaction to faults and an optimization of control parameters.

Control systems are mostly implemented by programmable logic controllers (PLC). Commercial PLCs, such as STEP 7, provide various solutions for wired sensors by different bus protocols.

Industrial automation has many communication standards, which include Profibus, EtherCat and Profinet. Profinet is the latest innovative protocol for the industrial Ethernet. Industrial automation is for production or electrical drives. Profinet is a much more efficient protocol used nowadays. In a harsh industrial environment, it is necessary that communication should be reliable and in real time. Due to the increase of the number of sensors and field devices, nowadays the cost for wiring is also increasing. There are also some industrial places where wireless sensors are rapidly being used. Wireless sensors or a wireless sensor network (WSN) can be placed far from the place where the actual phenomenon is occurring. In this type of sensor, the network position of the sensor and the communication topology are carefully implemented.

Wireless sensor networks are used to measure a variety of ambient conditions, such as temperature, humidity, pressure, noise level and lightning. Nowadays, wireless sensors are widely used in industrial environments. They can be used for the control of robots in a manufacturing system. There are several other places where wireless sensor networks can be efficiently implemented, such as the process control of the automation industry and the instrumentation of any industry. There are several factors, which can be affected by the design of a wireless sensor network. These factors may include the level of fault tolerance, what the minimum cost will be, what the network topology involved will be and, last but not the least, the transmission media, which transmit data. Generally, WSN consists of a sensor node and the sensor node gateway. This gateway can be used to store data using different protocols (Cayirci et al. 2001).

Wireless sensor networks provide a promising solution for an increasing number of logistical systems in which control cannot be implemented by wired sensors alone. Although they have been in the focus of research for more than 10 years now, they have found little attention in industrial control. Also, hardly any researcher of wireless sensors knows about industrial control requirements and protocols.

Wireless sensors can be easily implemented by embedding them into a product itself. This can improve the accuracy of measurement. There are different places where wireless sensors can play an important role instead of wired sensors, for example, warehouses for the storage of vegetables and fruits. These types of refrigerated warehouses need a low temperature and ventilation to keep the vegetables and fruits fresh. Proper ventilation is done using ventilation units.

Authors Version, see http://dx.doi.org/10.1007/978-3-319-45117-6_19 for official version. Please cite as:

Raza, T.; Lang, W.; Jedermann, R.: Integration of Wireless Sensor Networks into Industrial Control Systems. In: Freitag, M.; Kotzab, H.; Pannek, J.,(eds.): Dynamics in Logistics: Proceedings of the 5th International Conference LDIC, 2016 Bremen, Germany, Springer International Publishing, Cham, 2017, pp. 209-218. DOI: 10.1007/978-3-319-45117-6_19

At the moment, there is no proper sensor information, which can give the actual air-flow rate inside the boxes. Due to this, we cannot control the speed of the ventilator fans. They always run at the full speed, and a large amount of energy is wasted. To overcome this energy wastage, proper feedback control can be implemented using a wireless sensor. When we have proper information about the airflow, we can control the speed of the ventilator fans.

Wireless sensors can help to measure the temperature and other parameters at multiple locations as compared to wired sensors. In contrast to wired sensors, which are fixed and have a limited access, wireless sensors can be implemented in the boxes, which are temporarily placed in the warehouses.

Our paper shows how the number of applied sensors in industrial control can be increased by providing adequate protocol translators and interfaces to wireless technologies. A gateway for wireless sensor networks can be connected to industrial PLC systems. This prototype is currently compatible with Siemens PLC.

The paper is organized in the following way. The second section will explain about the basic block diagram and structure of the implementation. The third section will explain about the Raspberry Pi, which is used as a converter in this project. The fourth section will explain about the sensor module. The fifth section will explain the software implementation. The sixth section will be about the limitation of wireless sensors, and the last section will be the summary and conclusion.

Overall Structure

Due to the complexity of the Profinet protocol, it was not feasible to directly connect the wireless sensors with a PLC. A separate protocol translator was used instead, which also provided more flexibility. Different types of wireless sensors can be connected with only minor software changes. The complete chain of required hardware interfaces, units and protocols is displayed in Figure 1.

The temperature and humidity sensors are integrated on the WizziKit node. Wizzikit works on the Dash7 protocol. Dash7 stands for “Developer Alliance for standard harmonization of ISO 18000-7” and is used for low power consumption wireless devices. The Dash7 protocol uses a frequency range 433–434.97 Mhz in Europe [2]. The data wirelessly transmitted through the WizziMote is received at the WizziKit base, which can be called the Dash7 gateway because it works on the Dash7 protocol. Details about sensors and WizziKit will be discussed in the sensor module section of the paper.

The base station or gateway has a serial output. The serial output of the gateway is connected with the UART of the Raspberry Pi. The Raspberry Pi, which is used here, can be called a gateway, which translates one protocol to another protocol. The input protocol of the Raspberry Pi is a serial protocol, which can be seen from the block diagram, while the output protocol of the Raspberry Pi is Ethernet. This Ethernet is used as a profinet to communicate with the Siemens PLC simulator.

The data, which is received through the serial port of the Raspberry Pi, is then transmitted to the PLC simulator using Ethernet. The communication between Raspberry Pi and the PLC software is done using an open source library named Snap7 library.

This data is then displayed on the HMI (Human Machine Interface) using Winccflexible. Winccflexible is the HMI software for Siemens PLC. This software is used to programme the HMI panels. In this project, there is no real HMI panel used, so this software is used here to display the sensor values. The communication between the PLC simulator and Winccflexible is also done using a TCP/IP protocol.

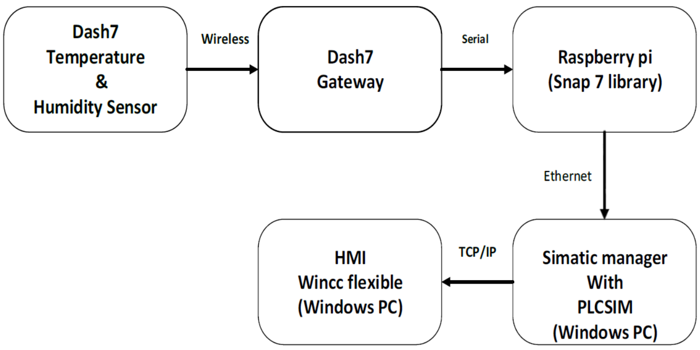

Fig 1. Block diagram of the overall structure

Raspberry Pi as an Industrial Gateway

Raspberry Pi is a credit card size computer with lots of built in functionalities. It is fully programmable and very flexible. There are different versions of Raspberry Pi available on the market. The Raspberry Pi, which is used here, belongs to the category of the ‘B’ model and has different properties. The processor of Raspberry Pi consists of 32- bit ARM. This CPU is based on the ARM11 architecture with 512 MB of RAM (Maksimovic and Vujovic 2014) for the model ‘B’ series.

One RJ45 connector is available, which is used to connect the Raspberry Pi with the Internet using an Ethernet cable; it can also be used for communication with other devices (Norair 2009).

Communication Protocol

Raspberry Pi has different communication protocols available, such as UART, I^2C and SPI. Every protocol has different aspects according to the requirements. UART was used to receive the data from wireless sensors. The detailed description about UART is given below

UART

UART is also called universal an asynchronous receiver/transmitter for serial communication. Serial communication is a low level method of transferring data between Raspberry Pi and a computer. There are two pins, which are used for the UART communication; these pins are called TXD and RXD. The TXD pin is used for data transmission while the RXD pin is used for receiving data.

To connect the Raspberry Pi with other devices, it should be noticed that the TXD pin of the Raspberry Pi should be connected with the RXD pin of the other device, and, similarly, the RXD pin of the Raspberry Pi should be connected with the TXD pin. Raspberry Pi also has a voltage level difference, which makes it impossible to connect with RS232. Raspberry Pi has a voltage level of 0 to 3.3V; it is necessary to use a voltage level converter to connect it with RS232 (Embedded Linux Forum 2015).

Sensor Module

A wireless sensor usually consists of a wireless sensor node and a base or gateway. A sensor node basically consists of four components that may include a processing unit, a sensing unit, a transceiver unit and a power unit. There are different applications for which additional components are included in the sensor node, such as a location finding system and power generator.

In the sensor node there is as processing unit, which is usually used to process data for communication with other nodes. The sensor node has a very important unit called a power unit. The power unit is used to provide the required power for the sensor node. A transceiver unit is used to connect with the network (WizziLab 2015).

The WizziKit, which is used, usually consists of two parts; these include

- WizziMote
- WizziBase

WizziMote

WizziMote has a built in microcontroller of Texas instrument TI CC430F5137. There is also an antenna fixed on WizziMote for the 433MHZ frequency (WizziLab 2015).

WizziBase

WizziBase is used to give a base board for WizziMote; there are different methods to connect WizzibBase with other devices. JTAG and a serial connector are available on the WizziBase. The WizziBase provides power to WizziMote with the help of JTAG, FTDI or an external power plug (WizziLab 2015).

Sensors

There are three sensors mounted on the WizziMote node. The following three data types were received using Raspberry Pi.

- Tmp 100 temperature sensor
- SHT25 temperature sensor
- SHT25 humidity sensor

Software Design

The software design has three parts, which are explained in the following three sections.

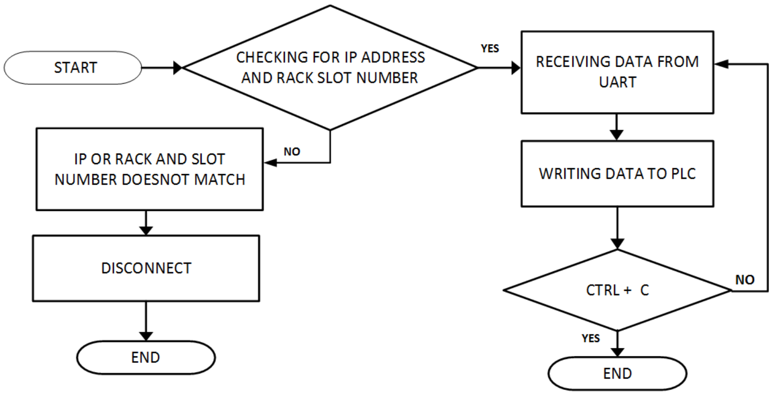

Figure 2. Flow chart describing the software module

Raspberry Pi

The software module used for receiving data from the UART of the Raspberry Pi and transmitting it through the Ethernet port of Raspberry Pi can be explained by a flow chart. An open source library named as the Snap7 library was used to communicate Raspberry Pi with the Siemens PLC simulator using Profinet.

The software module first checks the IP address and the rack and slot of the PLC simulator. If it does not match the required IP address or the rack and slot of the PLC it will terminate the program. If the requirement is fulfilled it will start receiving data from the Raspberry Pi and transmitting it to PLC using Profinet. ctrl+C is used to quit the main program. This program continuously reads data from UART, so this command is used to quit the program at any time.

PLC (Programmable Logic Controller)

The Siemens software named SIMATIC Manager was used to program the PLC and PLCSIM simulator used instead of a real PLC. The data, which was transmitted through the Ethernet port of the Raspberry Pi, was received in the shared data block.

There are two different types of data blocks in the PLC. One data block is called the instant data block while the other data block is called the shared data block. Data blocks are the memory area, which can be easily defined for different types of memory, such as bits, bytes, words and even our data types. In SIMATIC Manager, OB1, also called the organization block, is the main block where all the function blocks and functions are. It is similar to the main function in other programming languages.

The instant data block is the data block, which is associated with the function block. The instant data block must be created before using the function block to make possible a call for the function block in the OB1. In this data block, different values are stored, which are used by the function block.

The second data block is called the shared data block. If the system does not have enough bits to store the data, then we can use the shared data block to store the data. The shared data block has different aspects, as compared to the instant data block. The instant data block can share its data only with the function block for which it is created while, on the other hand, the shared data block data can be shared anywhere in the system (Siemens AG 2006).

The shared data block has an ability to share data. In this project, the shared data block was used to receive data from the Raspberry Pi. The reason behind using the data block in this project was that the data, which was transmitted by Raspberry Pi, was in a byte format. The data blocks can have any data type depending on the requirement of the user.

HMI (Human Machine Interface)

The software design section's last part consists of HMI software programming named Winccflexible. Although this software is used to program the HMI panel, in this project it was used to display the sensor values.

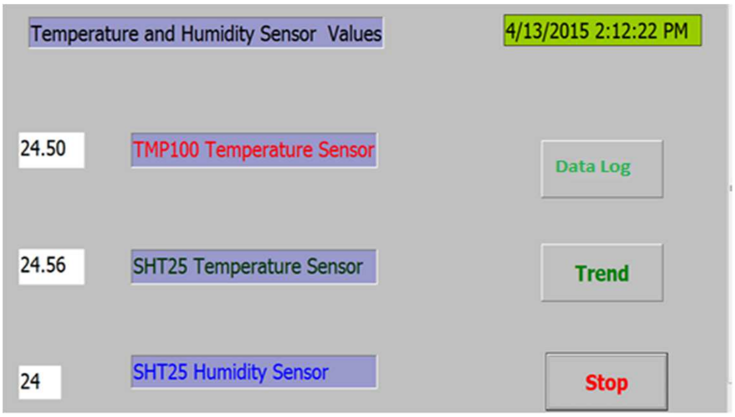

Figure 3. HMI for sensor values

Basically, HMI has three parts, which can be classified as follows.

- Temperature and Humidity sensor values
- Data logging
- Sensor values trends

Temperature and Humidity Sensor Values

In this part of HMI, we can see that there are three boxes, which have some values. These three boxes correspond to each sensor's real time value. These real time values are calculated in the SIMATIC Manager and then transferred into Winccflexible.

Data Logging

Data logging was also done using this HMI. In this HMI, there is a special feature, which is defined by the box on the right hand side. This box is named "Data Log". Due to this box, data logging can be started and controlled. Data logging was done in the Excel sheet format. All three sensors' values are connected with this single button. When we press this button, data logging will start automatically, and, when we press this button, again data logging will stop again. When the data logging starts, it will generate three Excel files at the back end as defined by the path of those files.

Sensor Value Trends

The third part of the HMI describes the trends among all three sensor values. There is a button on the right hand side of the HMI named trend. When we press this button, it will start generating the trend between the three sensor values. The trends give a better overview of any process. The trends, which are generated, are displayed as a bar graph.

Limitations of Wireless Control Applications

Industrial control systems have the highest requirements in terms of the reliability of the system components, whereas as wireless system always has to take communication failures into account. So, wireless communication should be made as reliable as possible, e.g. by using safe protocols with acknowledgements and automated re-transmission of lost packets.

Depending on the application, process control consists of tasks with different reliability requirements. Whereas wireless sensors might not be suitable for some critical security tasks, they can still be applied to other tasks, such as the optimization of process parameters. In case of lost sensor data, optimization can be skipped for one circle, but process control can continue without interruption, only with slightly less optimized parameters.

Wireless reading of product item parameters is another safe application in industrial control systems. The transmission of sensory information, such as recorded statistics of use and aging effects is not time critical. If necessary, the start of processing can be delayed until all required data have been transmitted.

Summary and Conclusions

In this paper, an idea of integrating a wireless sensor into industrial control systems was successfully implemented. The idea was not only successfully implemented, but an economical solution for the industrial gateway was also provided in the shape of Raspberry Pi. The idea was to integrate the wireless sensor using Profinet and to give the wireless sensor a serial base station. So it was necessary to convert the UART protocol into the Profinet protocol. Industrial gateways, which translate one protocol into another industrial protocol, are very expensive as compared to the solution provide here, which is only a $35 Raspberry Pi. This solution is also very flexible. In future, it can also be used for wireless sensors having a base station with different protocols such as I2C or SPI. The software is designed in such manner that this can be done easily with simple modification.

During the implementation, certain considerations were taken such that there was no real PLC available, so it would behave as if it were implemented in a real control system. For this reason, it was tested on a simulator. Open source software named Snap7 was used to make this idea into reality. This library was designed and implemented for real Siemens systems. So this was implemented the first time for a simulator system.

HMI was designed not only for a better understanding of the real time values but also to keep the values saved for further processing. For this implementation, data logging was performed for the all three sensor values. The real time values were displayed in HMI, and trends were also shown using HMI.

In future, there are some additional features, which can be added. This idea can be used currently for Siemens PLC. But, in future, this can be enhanced and made vendor independent. On Raspberry Pi, we can make further enhancement such that Raspberry Pi can be used as a plug and play device. There is also a possibility to make a GUI on Raspberry Pi to make adjustments in the program as per requirements from the GUI itself.

Acknowledgments The project COOL is supported by the German Federal Ministry for Economic Affairs and Energy on the basis of a decision by the German Bundestag.

References

Cayirci E, Akyildiz IF et al. (2001) Wireless sensor networks: a survey. Computer Networks International Journal of Computer and Telecommunications Networking. doi:10.1016/s1389-1286(01)00302-4

Embedded Linux forum (2015) RPi low-level peripherals. http://elinux.org/RPi_Low-level_peripherals Accessed 9 December 2015

Maksimovic M, Vujovic V (2014) Raspberry Pi as a wireless sensor node: performances and constraints. Information and Communication Technology, Electronics and Microelectronics (MIPRO) 37th International Convention. Doi:10.1109/MIPRO.2014.6859717

Norair JP (2009) Introduction to Dash7 technologies. RFID Journal

Siemens AG (2006) Automation and drives manual for STEP 7. http://www.fer.unizg.hr/download/repository/STEP7.pdf Accessed 9 December 2015

WizziLab (2015) WizziMote product page. http://www.wizzimote.com/ Accessed 9 December 2015