www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 4 Issue 3 March 2015, Page No. 10755-10759

RFID-Based Tracking System

Mr.G. Manikandan [1], Mr.N.Mathavan [2] , Mr.M.Paramasivan[3], T.Ashly [4]
[1,2,3] Assistant Professor, Department of Electronics and Communication Engineering,
[1]Kodaikanal Institute of Technology, Tamilnadu, India
[2]Nadar Saraswathi College of Engineering and Technology ,Tamilnadu, India
[3]Aksheyaa college of engineering ,Tamilnadu, India
[4] UG Student, Department of Information Technology,
Kodaikanal Instiute of Technology, Tamilnadu, India
e-mail : mrg.manikandan@gmail.com[1]

Abstract This paper describes the design of a radio frequency identification system that we called Trees RFID Tracking System (TRTS). This suggested study develops a system that would enable the detection and identification of trees illegal logging cases and hence preventing risks of species distinction and deforestation threats. The TRTS consists of RFID passive tags (static tags) fitted in trees and serving as unique identity for each tree; handheld readers (moving devices) with a suitable readable range and embedded circular polarization antenna. These readers would be held by forest officers and the data read from the tags would be accessible through the readers thanks to a visualization software that would analyze and process the data received. The database that saves all the readings and user interface and enables access to that data is located at the server side of the system. Communication between the tag readers and the server side is done through 3G connectivity enabled at the handheld reader device. An example of this suggested study practicality is the forests in Ifrane region of Atlas Mountains which are well known for the cedar species that are constantly subject to illegal extracting and thus are threatened by extinction. Moreover there has been no suggested method to improve their management process. Here is where our RFID system comes to play

Keywords- Space Carrier detect (CD); Clear to sound(CTS); Digital Interface Receiver (DIR); Data Terminal Ready (DTR).

I. INTRODUCTION

A Forest environmental monitoring system based on a wireless sensor network is proposed. It consists of three parts: data monitoring nodes, data base station and remote monitoring center. This system is suitable for the complex and large-scale forest environment monitoring, such as for trees. This paper is devoted to the explanation and illustration for our new forest environment monitoring system design. The system had successfully accomplished the online auto-monitoring of the deforestation. Deforestation is clearing Earth's forests on a massive scale, often resulting in damage to the quality of the land. Forests still cover about 30 percent of the world’s land area, but swaths the size of Panama is lost each and every year. The world’s rain forests could completely vanish in a hundred years at the current rate of deforestation. Forests are cut down for many reasons, but most of them are related to money or to people’s need to provide for their families. The biggest driver of deforestation is agriculture. Farmers cut forests to provide more room for planting crops or grazing livestock. Often many small farmers will each clear a few acres to feed their families by cutting down trees and burning them in a process known as “slash and burn” agriculture. Logging operations, which provide the world’s wood and paper products, also cut countless trees each year. Loggers, some of them acting illegally, also build roads to access more and more remote forests—which leads to further deforestation. Forests are also cut as a result of growing urban sprawl. Not all deforestation is intentional. Some is caused by a combination of human and natural factors like wildfires and subsequent overgrazing, which may prevent the growth of young trees.

Serial Communication

Serial communication is basically the transmission or reception of data one bit at a time. Today's computers generally address data in bytes or some multiple thereof. A byte contains 8 bits. A bit is basically either a logical 1 or zero. Every character on this page is actually expressed internally as one byte. The serial port is used to convert each byte to a stream of ones and zeroes as well as to convert a stream of ones and zeroes to bytes. The serial port contains a electronic chip called a Universal Asynchronous Receiver/Transmitter (UART) that actually does the conversion.

The serial port has many pins. We will discuss the transmit and receive pin first. Electrically speaking, whenever the serial port sends a logical one (1) a negative voltage is effected on the transmit pin. Whenever the serial port sends a logical zero (0) a positive voltage is affected. When no data is being sent, the serial port's transmit pin's voltage is negative (1) and is said to be in a MARK state. Note that the serial port can also be forced to keep the transmit pin at a positive voltage (0) and is said to be the SPACE or BREAK state. (The terms MARK and SPACE are also used to simply denote a negative voltage (1) or a positive voltage (0) at the transmit pin respectively).

When transmitting a byte, the UART (serial port) first sends a START BIT which is a positive voltage (0), followed by the data (general 8 bits, but could be 5, 6, 7, or 8 bits) followed by one or two STOP Bits which is a negative(1) voltage. The sequence is repeated for each byte sent. Figure 1 shows a diagram of what a byte transmission would look like.

At this point you may want to know what the duration of a bit is. In other words, how long does the signal stay in a particular state to define a bit. The answer is simple. It is dependent on the baud rate. The baud rate is the number of times the signal can switch states in one second. Therefore, if the line is operating at 9600 baud, the line can switch

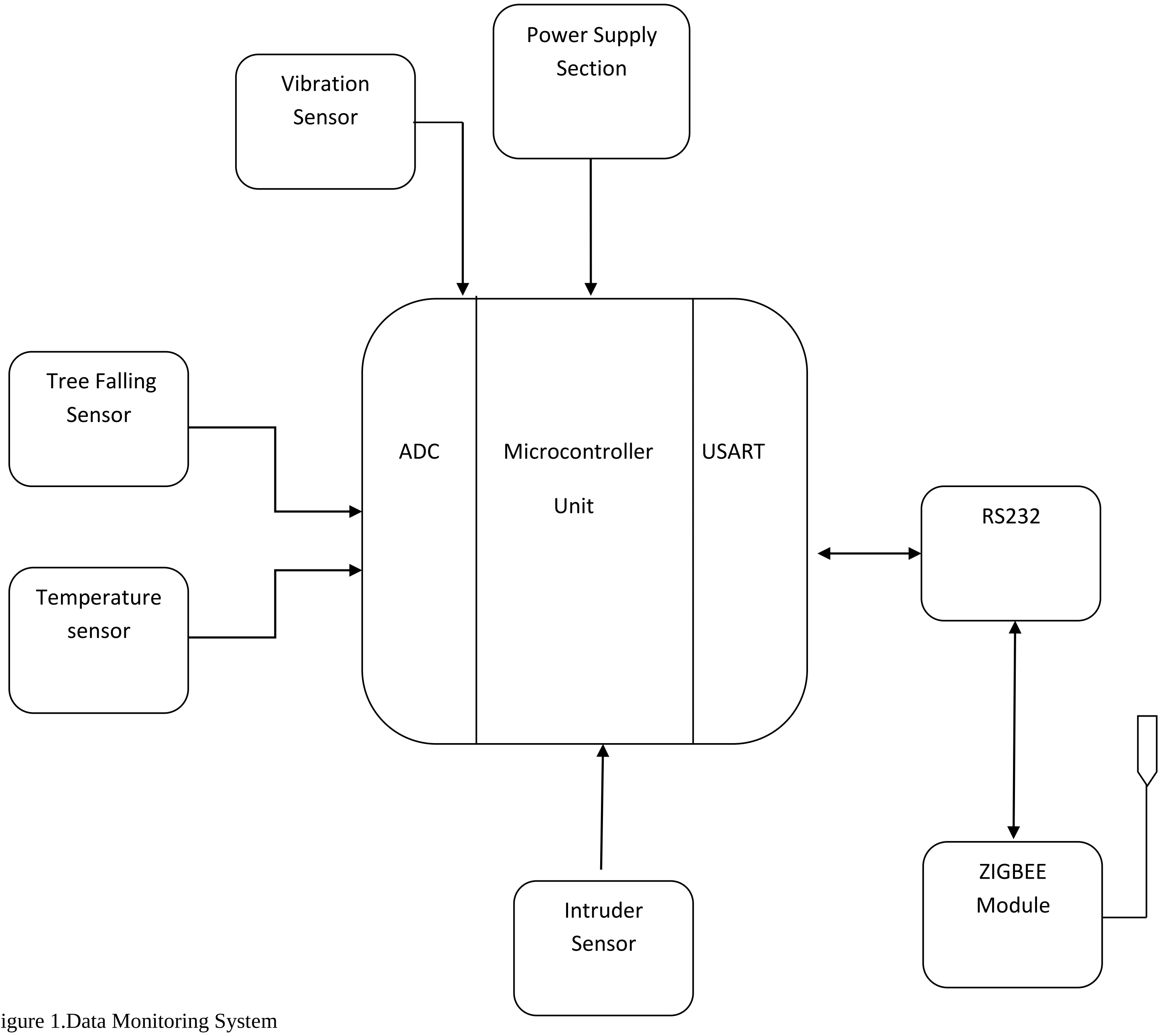


Figure 1.Data Monitoring System

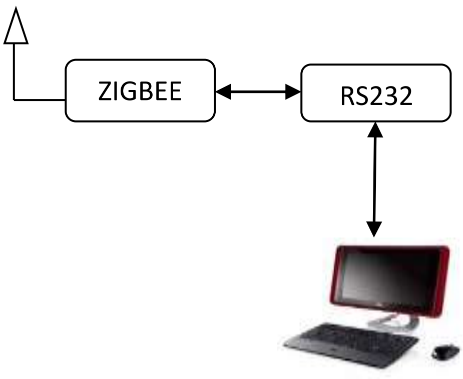


Figure 2. Data Base Station

states 9,600 times per second. This means each bit has the duration of 1/9600 of a second or about 100µsec.

when transmitting a character there are other characteristics other than the baud rate that must be known or that must be setup. These characteristics define the entire interpretation of the data stream. The first characteristic is the length of the byte that will be transmitted. This length in general can be anywhere from 5 to 8 bits.

The second characteristic is parity. The parity characteristic can be even, odd, mark, space, or none. If even parity, then the last data bit transmitted will be a logical 1 if the data transmitted had an even amount of 0 bits. If odd parity, then the last data bit transmitted will be a logical 1 if the data transmitted had an odd amount of 0 bits. If MARK parity, then the last transmitted data bit will always be a logical 1. If SPACE parity, then the last transmitted data bit will always be a logical 0. If no parity then there is no parity bit transmitted.

The third characteristic is the amount of stop bits. This value in general is 1 or 2. Assume we want to send the letter 'A' over the serial port. The binary representation of the letter 'A' is 01000001. Remembering that bits are transmitted from least significant bit (LSB) to most significant bit (MSB), the bit stream transmitted would be as follows for the line characteristics 8 bits, no parity, 1 stop bit and 9600 baud.
LSB (0 1 0 0 0 0 0 1 0 1) MSB

The above represents (Start Bit) (Data Bits) (Stop Bit). To calculate the actual byte transfer rate simply divide the baud rate by the number of bits that must be transferred for each byte of data. In the case of the above example, each character requires 10 bits to be transmitted for each character. As such, at 9600 baud, up to 960 bytes can be transferred in one second.

The above discussion was concerned with the "electrical/logical" characteristics of the data stream. We will expand the discussion to line protocol. Serial communication can be half duplex or full duplex. Full duplex communication means that a device can receive and transmit data at the same time. Half duplex means that the device cannot send and receive at the same time. It can do them both, but not at the same time. Half duplex communication is all but outdated except for a very small focused set of applications.

Half duplex serial communication needs at a minimum two wires, signal ground and the data line. Full duplex serial communication needs at a minimum three wires, signal ground, transmit data line, and receive data line. The RS232 specification governs the physical and electrical characteristics of serial communications. This specification defines several additional signals that are asserted (set to logical 1) for information and control beyond the data signal

These signals are the Carrier Detect Signal (CD), asserted by modems to signal a successful connection to another modem, Ring Indicator (RI), asserted by modems to signal the phone ringing, Data Set Ready (DSR), asserted by modems to show their presence, Clear To Send (CTS), asserted by modems if they can receive data, Data Terminal Ready (DTR), asserted by terminals to show their presence, Request To Send (RTS), asserted by terminals if they can receive data. The section RS232 Cabling describes these signals and how they are connected.

The above paragraph alluded to hardware flow control. Hardware flow control is a method that two connected devices use to tell each other electronically when to send or when not to send data. A modem in general drops (logical 0) its CTS line when it can no longer receive characters. It re-asserts it when it can receive again. A terminal does the same thing instead with the RTS signal. Another method of hardware flow control in practice is to perform the same procedure in the previous paragraph except that the DSR and DTR signals.

Note that hardware flow control requires the use of additional wires. The benefit to this however is crisp and reliable flow control. Another method of flow control used is known as software flow control. This method requires a simple 3 wire serial communication link, transmit data, receive data, and signal ground. If using this method, when a device can no longer receive, it will transmit a character that the two devices agreed on. This character is known as the XOFF character. This character is generally a hexadecimal 13. When a device can receive again it transmits an XON character that both devices agreed to. This character is generally a hexadecimal 11.

II.NULL MODEM

Serial communications with RS232. One of the oldest and most widely spread communication methods in computer world. The way this type of communication can be performed is pretty well defined in standards. I.e. with one exception. The standards show the use of **DTE/DCE** communication, the way a computer should communicate with a peripheral device like a modem. For your information, **DTE** means *Data Terminal Equipment* (computers etc.) where **DCE** is the abbreviation of *Data Communication Equipment* (modems).

One of the main uses of serial communication today where no modem is involved—a *serial null modem* configuration with **DTE/DTE** communication—is not so well defined, especially when it comes to flow control. The terminology *null modem* for the situation where two computers communicate directly is so often used nowadays, that most people don't realize anymore the origin of the phrase and that a null modem connection is an exception, not the rule.

In history, practical solutions were developed to let two computers talk with each other using a null modem serial communication line. In most situations, the original modem signal lines are reused to perform some sort of handshaking. Handshaking can increase the maximum allowed communication speed because it gives the computers the ability to control the flow of information. A high amount of incoming data is allowed if the computer is capable to handle it, but not if it is busy performing other tasks. If no flow control is implemented in the null modem connection, communication is only possible at speeds at which it is sure the receiving side can handle the amount information even under worst case conditions.

III. RS232

When we look at the connector pin out of the RS232 port, we see two pins which are certainly used for flow control. These two pins are **RTS**, request to send and **CTS**, clear to send. With **DTE/DCE** communication (i.e. a computer communicating with a modem device) **RTS** is an output on the **DTE** and input on the **DCE. CTS** are the answering signal coming from the **DCE.**

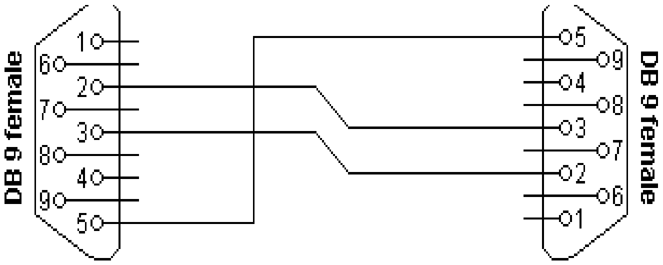
Before sending a character, the **DTE** asks permission by setting its **RTS** output. No information will be sent until the **DCE** grants permission by using the **CTS** line. If the **DCE** cannot handle new requests, the **CTS** signal will go low. A simple but useful mechanism allowing flow control in one direction. The assumption is that the **DTE** can always handle incoming information faster than the **DCE** can send it. In the past, this was true. Modem speeds of 300 baud were common and 1200 baud was seen as a high speed connection.

For further control of the information flow, both devices have the ability to signal their status to the other side. For this purpose, the **DTR** data terminal ready and **DSR** data set ready signals are present. The **DTE** uses the **DTR** signal to signal that it is ready to accept information, whereas the **DCE** uses the **DSR** signal for the same purpose. Using these signals involves not a small protocol of requesting and answering as with the **RTS/CTS** handshaking. These signals are in one direction only.

The last flow control signal present in **DTE/DCE** communication is the **CD** carrier detect. It is not used directly for flow control, but mainly an indication of the ability of the modem device to communicate with its counterpart. This signal indicates the existence of a communication link between two modem devices.

IV NULL MODEM WITHOUT HANDSHEKING

How to use the handshaking lines in a null modem configuration? The simplest way is to don't use them at all. In that situation, only the data lines and signal ground are cross connected in the null modem communication cable. All other pins have no connection. An example of such a null modem cable without handshaking can be seen in the figure below.



Connector 1	Connector 2	Function
2	3	Rx ← TX
3	2	TX → Rx
5	5	Signal ground

Figure 3. Simple null modem without handshaking

V. ISSUES

If you read about null modems, this three wire null modem cable is often talked about. Yes, it is simple but can we use it in all circumstances? There is a problem, if either of the two devices checks the **DSR** or **CD** inputs. These signals normally define the ability of Other side to communicate. As they are not connected, their signal level will never go high. This might cause a problem.

The same holds for the **RTS/CTS** handshaking sequence. If the software on both sides is well structured, the **RTS** output is set high and then a waiting cycle is started until a ready signal is received on the **CTS** line. This causes the software to hang because no physical connection is present to either **CTS** line to make this possible. The only type of communication which is allowed on such a null modem line is data-only traffic on the cross connected **Rx/TX** lines.

This does however not mean that this null modem cable is useless. Communication links like present in the Norton Commander program can use this null modem cable. This null modem cable can also be used when communicating with devices which do not have modem control signals like electronic measuring equipment etc.

As you can imagine, with this simple null modem cable no hardware flow control can be implemented. The only way to perform flow control is with software flow control using the **XOFF** and **XON** characters.

VI. ZIGBEE MODULE

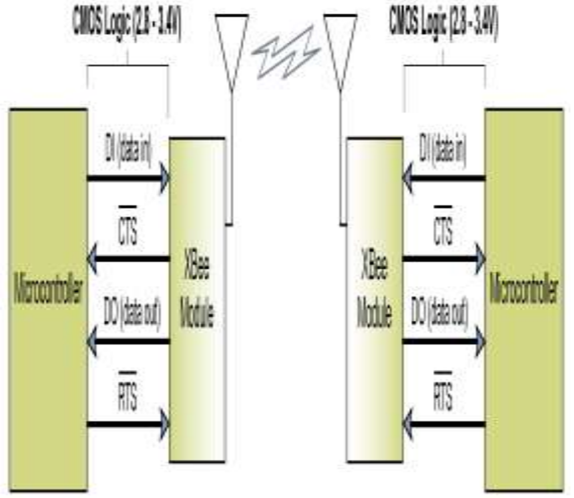


Figure 4. Zigbee module

- The XBee and XBee-PRO OEM RF Modules were engineered to meet IEEE 802.15.4 standards and support the unique needs of low-cost, low-power wireless sensor networks. The modules require minimal power and provide reliable delivery of data between devices.
- The modules operate within the ISM 2.4 GHz frequency band and are pin-for-pin compatible with each other
- The XBee®/XBee-PRO OEM RF Modules interface to a host device through a logic-level asynchronous serial port. Through its serial port, the module can communicate with any logic and voltage compatible UART; or through a level translator to any serial device (For example: Through a Digit proprietary RS-232 or USB interface board).

VII. CONCLUSION

In this paper the main focus is to make an intelligent decision using a fuzzy logic system in wireless sensor networks for forest fire detection. With means of fuzzy logic toolbox in MATLAB we are able to find the probability of fire and further deeper analysis can be done to observe effects of each input on the entire system. In addition, it can be easily implemented and modified during the process.

[1] A. Burger, "Helveta: Supply Chain Software Firm Tackles Deforestation by digitizing trees, timber."

[2] D. Pritchard, "Reducing emissions from deforestation and forest degradation in developing countries (REDD)." presented for Foundation for International Environmental Law and Development, March 2009.

[3] M. Bhuptani, and S. Moradpour, RFID Field Guide - Deploying Radio Frequency Identification Systems. Prentice Hall, New Jersey, 2005.

[4] L. Bolotnyy, and G. Robins, "The case for multi-tag rfid systems." International Conference on Wireless Algorithms, Systems and Applications (WASA), pp. 174-186, August 2007.

[5] G. Qiu, "RFID-enabled automation in support of factory integration," Robotics and Computer-Integrated Manufacturing, Vol. 23, No. 6, pp. 677-68, Dec 2007.

[6] Y. Sangkeun, L. Junseob, K. Yongwoon, and K. Hyungjun, "An integrated mobile RFID service architecture between B2B and B2C networks," 9th IEEE International Conference on Advanced Communication Technology, Vol. 1, pp. 90–93, Feb. 2007.

[7] P. Zeydin and N. Inanc, "Smart parking applications using RFID Technology," 1st Annual RFID Eurasia, pp. 1–3, Sept. 2007.

[8] Z. Min, L. Wenfeng, W. Zhongyun , L. Bin and R. Xia, "A RFID based material tracking information system," IEEE International conference on Automation and Logistics, pp. 2922- 2926, August 2007.

[9] F.A. Aloul, N.R. Aji, A.A. Al-Zarouni, and N.H. Fakhro, "Mobile RFID tracking system," 3rd International Conference on Information and Communication Technologies: From Theory to Applications, ICTTA, pp. 1- 4, April 2008.

[10] K. Y. Wu, H. Roy Campbell and H. Nitin, "RFID trees: A distributed RFID tag storage infrastructure to backtrack hikers in a forest," Sensor Mesh and Ad How Communication and Networks conference, pp. 1-8, June 2010.

[11] S. Hoyt, D. S. John, D. Wilson, and L. Bushnell, "A treetour with radio frequency identification (RFID) and a personal digital assistant (PDA)," in Proc. 2003 Annual Conference of the IEEE Industrial Electronics Society, vol. 3, Roanoke, VA, pp. 2035– 2040, Nov. 2003.

.