Wireless Control of an Automated Guided Vehicle

Rajeev K Piyare, *Member, IAENG,* and Ravinesh Singh

***Abstract*—This paper represents the design, implementation, and experimental results of a Radio Frequency (RF) based wireless control of a distributed Peripheral Interface Controller (PIC) microcontroller based Automated Guided Vehicle (AGV), which is known as ROVER II (Roaming Vehicle for Entity Relocation). ROVER II was designed in-house as a general purpose guide path following mobile platform for material handling and transportation within a manufacturing facility.**

***Index Terms*—Automated Guided Vehicle (AGV), communication, Peripheral Interface Controller (PIC), Radio Frequency (RF), ROVER II (Roaming Vehicle for Entity Relocation)**

I. Introduction

Wireless communications nowadays is becoming increasingly popular for factory and process control automation systems. Part of this growth is due to the emergence of very reliable radio frequency technologies capable of handling the extreme conditions present in industrial plants. The other factor driving growth is the realized benefits that wireless presents including reduced installation costs, elimination of phone line charges for remote sites, reduced mechanical wear to moving platforms (thus improving performance of material handling systems) and providing crucial information for production and maintenance workers wherever needed.

Industrial applications involving mobile systems can benefit from the use of wireless communication technologies. The localization and tracking of components, the coordination of autonomous transport vehicles and mobile robots, as well as applications involving distributed control are all areas in which wireless technologies could be used in an industrial environment [1].

It is known that AGV's are useful for the movement of material and for the productivity of automated systems. Fast AGV's with obstacle avoidance capabilities can maneuver themselves among manufacturing lines carrying components [2].

AGV control is usually done locally. An alternate implementation involves a remote control. This schema opens up a full set of new applications such as coordination of tasks in multi-AGV systems, providing at the end more flexible manufacturing systems. Based on the benefits provided by the remote control, an adequate architecture for AGV systems may consist in an external controller sending and receiving, through a wireless network, the control commands to the vehicle [3].

Due to financial limitations, the approach in this project is to use a low cost 8 bit RF transceiver that is capable of sending and receiving data bit over a fixed distance to control ROVER II. This approach, though, has limitations, it entails a simpler algorithm and it can send path data from a remote site computer to ROVER II.

The significance of the study is in the contribution in the field of wireless control of automated guided vehicle systems and robotics. Through this project, a simpler approach and algorithm is designed for using low cost and in house available RF transceiver and AGV with low processing requirements but with effective results.

II. Review of Related Literature

A number of projects are already available at different institutions in various countries with regard to the development of AGVs. This projects offer different complex approaches with different functionalities and features for the development and control of AGVs.

A. *Simple position sensor networks*

In this approach the position of a robot is gathered from infra-red sensor data and is then transmitted to other robots [4]. Data is transmitted via a two-way radio link. Robots are identified by a unique code which leads the transmitted position data. The received information is decoded and processed by the on-board microcontroller of the robot. The radio link uses an Ultra High Frequency transmitter/receiver and an encoding Integrated Circuit which detects if the address matches the robot's ID and then transfers the data directly to the microcontroller. The communication network as suggested is tailor made to communicate with very specific data between agents [4]. The system is therefore inflexible to any changes of the communication system.

B. *Inter-robot network (IRoN)*

IRoN is a robot communication network with modest cooperation and is flexible enough to support a range of tasks [5]. Cooperation between robots is achieved by exchanging state variables using implicit communication. In addition explicit communication routines are available. The

Manuscript received December 08, 2010. (Manuscript submitted for review on November 22, 2010.)

Rajeev K Piyare was with the University of the South Pacific and is now with the Department of Electrical & Electronics Engineering of Fiji National University, Suva, Fiji Islands. (phone: +679 9234591; e-mail: rajeev.piyare@fnu.ac.fj).

Ravinesh Singh is with the School of Engineering, University of The South Pacific, Suva, Fiji Islands. (e-mail: singh_rv@usp.ac.fj).

signal to noise ratio is increased at the expense of bandwidth by using a spread spectrum technique with 19,200 bits/second using a commercial radio modem, i.e. the energy of each single bit is spread over the entire frequency band.

IRoN uses a token ring communication to perform time division multiplexing which is relatively easy to implement. IRoN is a high performance robot network and is able to perform implicit and explicit communication. The drawback is that the state variables implicitly exchanged may not be suitable and/or sufficient for all robot applications, i.e. the complexity of the user processes increases while the Netserver module is kept simple [4]. Implicit communication is based on explicit communication, i.e. can be implemented on every system with explicit communication. IRoN can easily be modified to either maintain additional state variables and/or add new processes to handle the communication on each host.

In this paper a low cost RF communication architecture and control strategy of an AGV is proposed. The paper is organized as follows. The system architecture is explained in section III. Section IV describes the software design and experimental results are presented in section V. The Conclusion and recommendation are given in section VI.

III. SYSTEM ARCHITECTURE

ROVER II had been designed in-house as an automatic guided vehicle that uses path coordinates to guide it [6]. It is a six wheeled vehicle as shown in Fig.1.The four spherical roller castor wheels are used for steering the AGV and the two middle wheels are used for driving. The driving action is carried out two Tunglee DM08GN DC motors. The encoders are attached on the two driving wheels in order to measure the vehicle's displacement and then calculate its real time position and orientation.

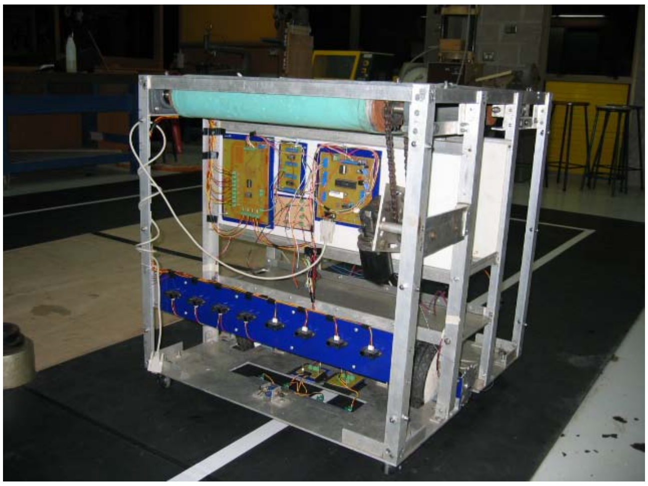


Fig. 1. ROVER II the mobile robotic platform

It consists of five processors in a star distributed architecture as illustrated in Fig.2.The Keypad and Liquid Crystal Display (LCD) controller; the Motor controller; the Sensor controller; the Power supply and the Master controller. Each controller is driven by a PIC16F877 microcontroller. The Master controller oversees and drives all the other co-controllers.

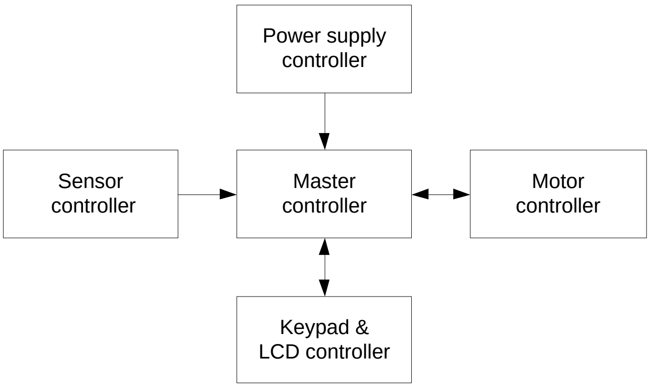


Fig. 2. ROVER II's star distributed microcontroller architecture

A. *Operation of the AGV*

The AGV (ROVER II) which has been developed is capable of following any predefined route of a complex guide path for material handling and transportation within a manufacturing system. The user firstly has to provide user authentication by the verification of password followed by specifying and uploading a particular path data using the keypad and LCD module, which is then transferred and stored on the master controller. Table 1 shows the respective two bit logic path data coordinates that needs to be appropriately entered by the user depending on the path actions. The coordinates entered by the user include a series of 1s and 0s whereas a different combination determines a different function. For instance, entering an x value of 1 and a y value of 1 would tell ROVER II to move forward whereas entering the value 0 and 1 would make ROVER II turn right as illustrated in Table 1.

TABLE 1
BINARY REPRESENTATION OF ROVER II'S PATH ACTIONS

Action	Path Data (Binary)
Stop	00
Turn Right	01
Turn Left	10
Go Forward	11

The Master controller uses this path data to control the Motor controller to follow the predefined route by appropriately varying the speeds of the two 12V, DM08GN direct current (DC) motors.

The left and right line tracing sensors controlled by the Sensor controller is used to keep the AGV on track and additional front and side infra-red (IR) ranging sensors are used to detect obstacles and stations respectively. This enhancement gives the AGV capability to navigate complex guide paths as shown in Fig.3 which can have any number of corners and junctions [6].

ISBN: 978-988-19251-2-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

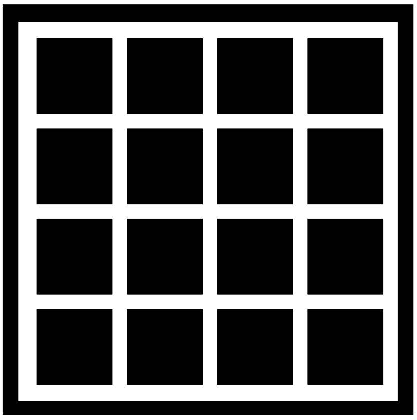


Fig. 3. Complex Guide Path

As the stations are detected the AGV stops for 5seconds to load/offload materials using the conveyor unit. However, if the obstacles are detected than the AGV stops until the obstacles are removed from the front of the sensors. In both these cases, whenever the AGV restarts after stopping, it recalls the operation which it was performing before being interrupted to stop and hence continues performing the same task which it was doing before stopping. As a result, ROVER II has memory capability to remember the task that it was performing before stopping and recalls this when it resumes operation. The Sensor controller reads in all these sensor inputs as hardware interrupt inputs and hence generates interrupt signals as either of these sensors is activated.

A power management system by the Power supply controller is also incorporated which automatically swaps the AGV's supply battery with a backup battery, once the voltage of the supply battery falls below a specified threshold voltage.

This paper focuses mainly on how the Master controller interfaces with the Keypad co-controller and with this knowledge, creates a wireless control mode that is compatible with the master program (i.e. the Master controller will read data as though it was from the Keypad controller yet it is from the transceiver).

The capability of sending data has been enhanced by introducing a means of wireless control in which a user would not need to approach the AGV to enter the necessary path coordinates for it to function. The user on the other hand, can send the data from a remote site computer. This is possible by introducing a low cost RF transceiver that is capable of sending and receiving data bits wirelessly.

However, the architecture as shown in Fig.2 has to be changed to accommodate the wireless controller. Fig.4. shows the proposed architecture that was considered to be the best option to make ROVER II wireless.

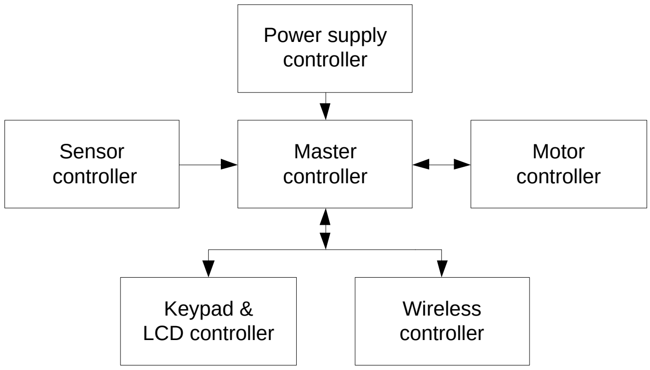


Fig. 4. ROVER II's proposed architecture

B. Overall system setup

The overall setup of the system is composed of a desktop computer control unit, a TLP434A RF signal transmitter, a RLP434A RF signal receiver, an AGV path, and a six-wheeled AGV. The RF signal transmitter is composed of a hardware transmitter connected to the PIC16F877 Boot loader board. The PIC16F877 Boot loader board is where the main program resides and is connected through a serial port (RS232) to the desktop computer. Tera Term program is used as a user friendly interface to send the command instructions composing of two bit path data signals to the PIC16F877 Boot loader board which transmits AGV path data through RF transmitter. The control strategy is showed by the simple block diagram in Fig.5.

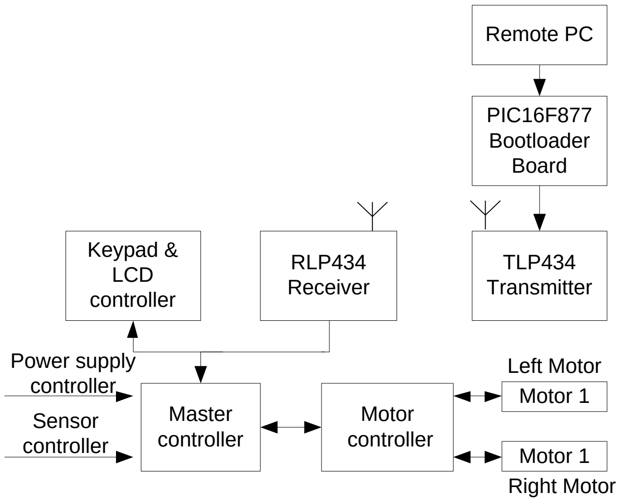


Fig. 5. Block diagram of wireless control

C. Wireless module

The data is transmitted half-duplex with speeds up to 8kb/s and up to 30m distance using a Laipac ASK Hybrid Transceiver as RF radio module. The data is transferred using an Amplitude Shift Keying modulation technique at a frequency of 433.92MHz and the radiated power from a quarter wave antennas is 0.25mW. The ASK modulation is obvious since the project requires transfer of digital data.

Since the receiver can handle a maximum of 6V supply, the entire circuit is operated at 5V, even though the encoder,

ISBN: 978-988-19251-2-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

decoder and transmitter can handle supplies of up to 12V. The address lines for both the encoder and decoder are grounded for this circuit since a valid transmission only occurs if address values are same on both circuits. In order to transmit, the data switches on the HT12E encoder are set to the required logic levels (closed switch to represent logic low and open switch to represent logic high) and then the transmit enable switch is closed to enable transmission of both the 8 bit address (all zero's in this case) and 4 bit data. The encoder outputs the 12 bit data serially using DOUT which goes to the TLP434A transmitter which transmits the encoded signal at 433.92MHz. This signal is received by the RLP434A receiver on ROVER II. The serial form of received data is passed on to the HT12D decoder which checks the addresses three times with its local addresses. Upon a match, the decoder latches the exact 4 bits data on data output pins while making the valid transmission pin high.

D. Interfacing Wireless controller to the Master Controller

Only 3 pins of the RF receiver are connected to the input PORTD, pins D4 and D5 and to PORTB, pin B5 respectively because there are only few control signals. The master-controller on the ROVER II reads this data from the input port and sends the appropriate 2-bit path data to the motor drivers. It can have 4 possible outputs as stated in Table 1 but the stop instruction can be done if the 2 bits are zero.

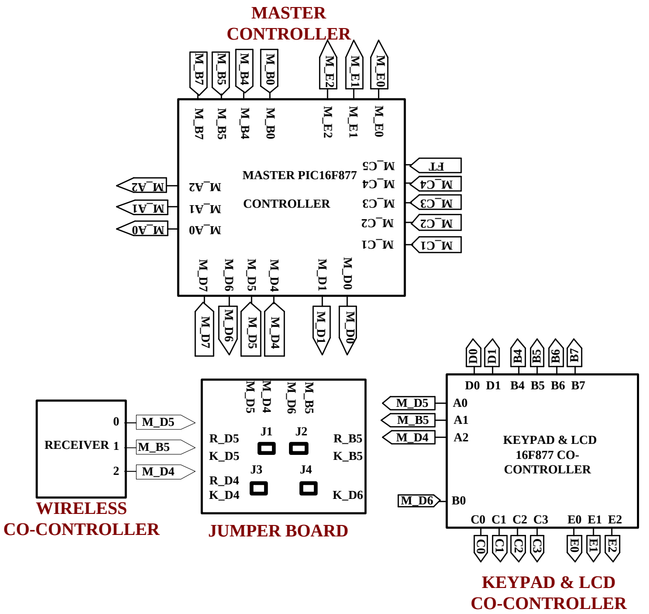


Fig. 6. Schematic layout of the Master controller with Wireless and Keypad Controller

To make it possible to have the option for wireless control or keypad control a jumper board was implemented as shown in Fig 6. The pins labeled with K_D4, K_D5 and K_B5 refers to connections from the Keypad and LCD controller. The pins labeled R_D4, R_D5 and R_B5 are for receiver outputs and the ones labeled M_D4, M_D5, M_D6 and M_B5 are for the Master controller. The user has to manually switch the jumpers in order to use wireless control or keypad control. When jumpers are connected to the left most pins, the Master controller is reading the path data through the wireless mode and when switched to the right most pins the path data are read from the Keypad and LCD controller.

IV. Software Design

The software was designed in such a way so as to enable a standard pattern of operation. The software coding for this project was done in PIC C compiler, a standard compiler used for PIC microcontroller programming at the School of Engineering, University of the South Pacific. Even when all the separate hardware is connected together, the unity among the components and the wholeness of the system is only achieved through software implementation [7].

The transceiver program used for sending and receiving data on the computer had to be user friendly as it is the means of communicating ROVER II with the user. The program itself was adapted from the original keypad and LCD program with some modifications and additions made to it.

The program starts off by welcoming the user and displays 2 options for user to choose from. Option 1 offers the user access and option 2 to exit the program as shown in Fig.7. If the user enters a '2' on the computer keyboard and finishes by pressing 'Enter', the computer displays a message declaring that it is safe now for user to exit from the program.

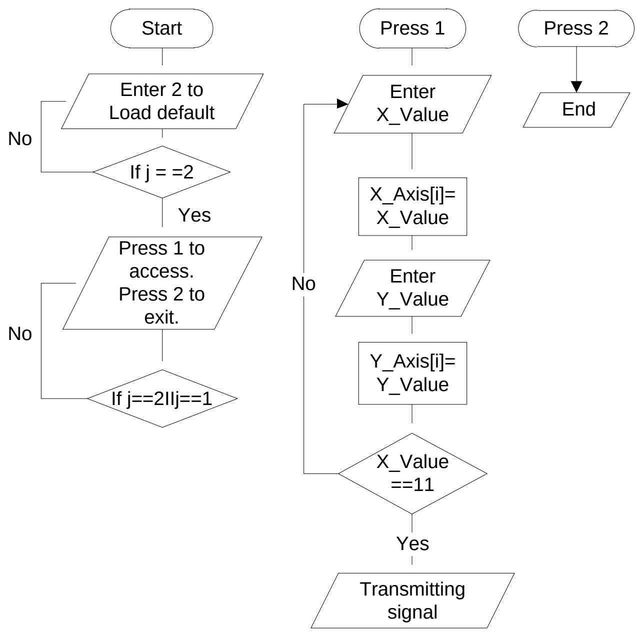


Fig. 7. Computer transceiver main program

However, if the user enters a '1', the program continues by generating a logic high control signal on PORTA pin A1 of the Wireless controller which interrupts the Master controller by activating its PORTB pin B5. All the PIC16F877 microcontrollers have an interrupt on change feature on pins B4 to B7 such that if interrupt RB (INT_RB) is enabled, then any logic changes on these pins would cause an interrupt. Hence, the logic high control signal generated

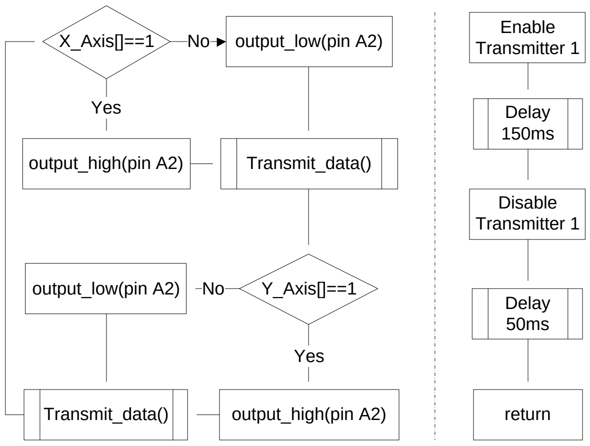
ISBN: 978-988-19251-2-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

on PORTA pin A1 which interrupts the Master controller by activating its PORTB pin B5 tells the Master controller that the Wireless controller is now activated.

Then the computer shows that access has been granted and prompts the user to enter the first X_Value. After entering the X_Value the computer checks if the value is valid (i.e. that it is either a 0 or 1). If the value is not valid, the computer asks the user to enter X_Value again. Until a valid X_Value is entered, the computer will keep asking the user to enter an X_Value. This also applies to entering the Y_Values. When the user enters a valid X_Value, the transceiver stores the value in its internal memory and asks the user to enter the corresponding Y_Value. This process continues on until the user enters the value '11' which tells the transceiver the end of data and it starts transmission from the computer transceiver to the ROVER II transceiver.

The two bit path data combinations as shown in Table 1 are stored in X_Data array (X_Data[ ]) and Y_Data array (Y_Data[ ]) respectively. The X_Data array holds the most significant bit combination. These combinations of path data are finally transmitted to the Master controller by the Communication_Link() subroutine function as indicated in Fig.8a.

In the Communication_Link() subroutine function the Wireless controller sends the two bit path data combinations to the Master controller. PORTA pin A0 is first made high which is sent to the Master controller PORTD pin D5. This pin remains high for as long as the data transfer takes place and pulled low at the conclusion of the Communication_Link() subroutine program. PORTA pin A1 at the start of this subroutine is low but once the correct data logic is available on PORTA pin A2, it is made high which generates an interrupt signal (INT_RB) on the Master controller at PORTB pin B5. This interrupt indicates to the Master controller that a valid data is available on PORTD pin D4 which has been sent by PORTA pin A2 of Wireless controller. The two bit path data patterns as mentioned in Table 1 are stored in the X_Data array (X_Data[ ]) and Y_Data array (Y_Data[]) on the Wireless controller and are read one at a time in a sequential manner and then sent to the Master controller on a single data line PORTA pin A2. When the data is sent through pin A2, it waits to be enabled which is done by the transmit_data () subroutine is shown in Fig.8b. The transmit_data () subroutine sends enables TE1 which sends the data while it delays it for 150ms and then disables it again for the next 50ms.



(a) Communication_Link() (b) transmit_data()

Fig. 8. Communication_Link() and transmit_data() subroutines

A 150ms delay is introduced between each data transfer that gives the master controller enough time to be able to correctly read in this data which it also stores in a similar array at the other end. This process of transmitting the two bit path data continues until there is no more data left which is denoted by variable *i* becoming zero and then the subroutine program returns to the main program by finally turning PORTA pin A0 and A1 low. When completion of data transmission occurs, the program loops back to the beginning of the subroutine where it offers the user two options, one for granting access and the other for exiting the program.

V. EXPERIMENTAL RESULTS

Command window illustrated in Fig.9 is used to control ROVER II's movement operation through PC in long distance area. There is a path action command (x,y position) in each axis such as Stop, Turn Right, Turn Left and Go Forward as illustrated in Table 1. In path action command, set of x, y coordinated path data pairs are sent to ROVER II with wireless communication channel. ROVER II receives the command of two bit path data to control the AGV with regards to predefined path.

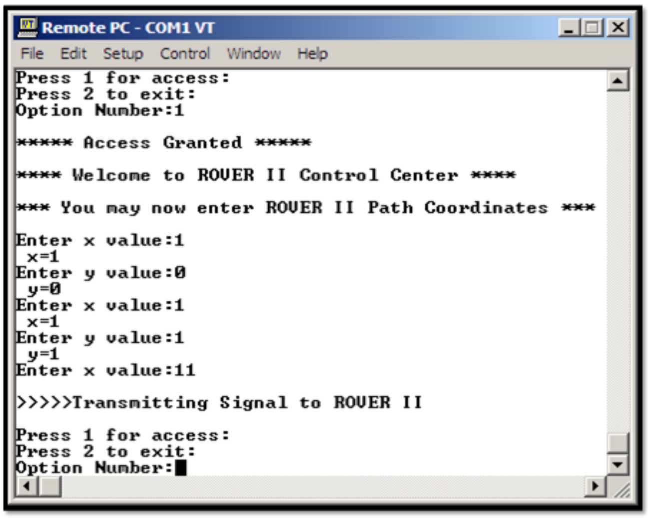


Fig. 9. ROVER II's command window

ISBN: 978-988-19251-2-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Experiments are conducted by sending different combinations of two bit path data to ROVER II wirelessly. Fig.10 shows the transmitted data and Fig.11 shows the data that was received by the Master controller using Tera Term program and both data are the same.

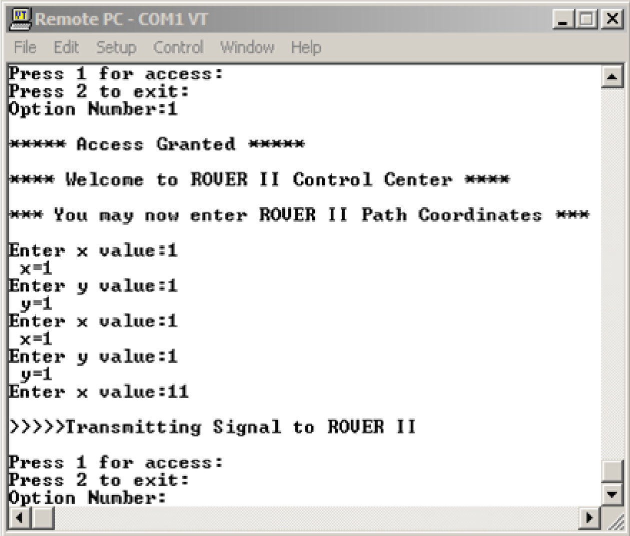


Fig. 10. Transmitted Data from Remote PC

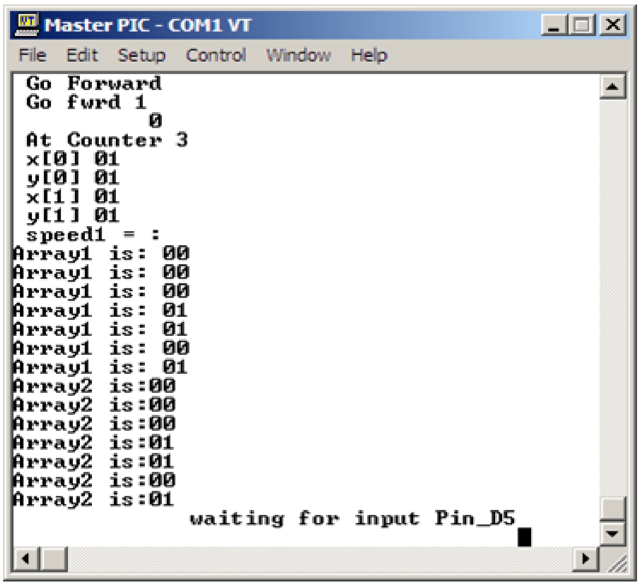


Fig. 11. Received Data on the Master controller

VI. Conclusion

The present paper presents the design, implementation and experimental results of Radio frequency (RF) based wireless control onto a distributed PIC microcontroller based AGV, known as ROVER II. One of the transceivers was used for sending the two bit path data from the remote site computer and the other was dedicated for receiving data on ROVER II and sending it to the Master controller. The physical communication port is a serial interface from the remote PC to the PIC16F877 Boot loader board. The command window enables remote AGV control and is capable of sending six x values and six y values which is used to guide ROVER II.

Future work is planned to increase the capability and the efficiency of ROVER II by creating a user friendly Graphical User Interface (GUI) and also incorporating neural network algorithm for path planning. Treatment of dynamic model and machine vision application of automated vehicle are also planned for the next step.

References

[1] A. Willig, K. Matheus and A.Wolisz, “Wireless Technology in Industrial Network,” *Proceedings of the IEEE*, Vol 93, Issue 6, pp. 1130-1151, June 2005.

[2] V.K.Kongezos and C.R.Allen,“Wireless communication between AGV’s (Autonomous Guided Vehicle) and the industrial network CAN (Controller Area Network),” *Proceedings of the 2002 IEEE*, International Conference on Robotics and Automation, Vol , pp. 434-437, May 2002.

[3] C. Lozoya, P. Marti, M. Velasco and J.M. Fuertes, “Effective Real-Time Wireless Control of an Autonomous Guided Vehicle,” *Proceedings of the IEEE*, pp. 2876-2881, 2007.

[4] C. Fung, H. Eren and Y. Nakazato, “Position sensing of mobile robots for team operations,” IMT94, pp. 185-8, 1994.

[5] M. Rude, T. Rupp, K. Matsumoto, S. Sutedjo and S. Yuta, “IRoN: an inter robot network and three examples on multiple mobile robots’ motion coordination,”IEE97, pp. 1437-44, 1997.

[6] R. Singh, “A Distributed PIC Microcontroller Architecture for AGV Application,” MSc thesis, University of the South Pacific, Suva, Fiji, 2006.

[7] G. C. Onwubolu, I. Jannif, M. Tazil, and A. Singh, “Mobile robotic platform controller with an effective data communication protocol,” Journal of Engineering Manufacturer-Proceedings Part B, Vol 220, pp. 1175-1188, 2006.

Bibliography

Rajeev K Piyare received his Bachelor’s Degree in Electrical & Electronics Engineering from University of The South Pacific, Fiji, 2008. Currently he is with Department of Electrical & Electronics Engineering, Fiji National University and pursuing his Postgraduate Diploma in Electronics Engineering. His area of interests includes automation, robotics and image processing.

Ravinesh Singh is with the School of Engineering, University of The South Pacific, Suva, Fiji. His area of search and teaching interest includes microprocessors and robotics.

ISBN: 978-988-19251-2-1
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)