Wireless Personal Area Network based Semiautonomous robot using 802.15.4 b LAN standard protocol (ZIGBEE) with MATLAB GUI for coal mine Uses

Rajesh Singh[1] Madhu Sharma[2]
[1]HEAD, Robotics Institute R&D, Assistant Professor,
University of petroleum and Energy studies,
Dehradun,INDIA
[2]College of Engineering Studies, Assistant Professor,
University of petroleum and Energy studies,
Dehradun,INDIA
rsingh@ddn.upes.ac.in[1]
madhusharma@ddn.upes.ac.in[2]

Vivekkaundal[3]PurnenduShekharPandey[4]
[3]College of Engineering Studies, Assistant Professor,
University of petroleum and Energy studies,
Dehradun,INDIA
[4]Electronics and Communication Deptt., Lecturer
THDC IHET, College,Tehri,Uttrakhand, INDIA
vkaundal@ddn.upes.ac.in[3]
purnendu12345@gmail.com[4]

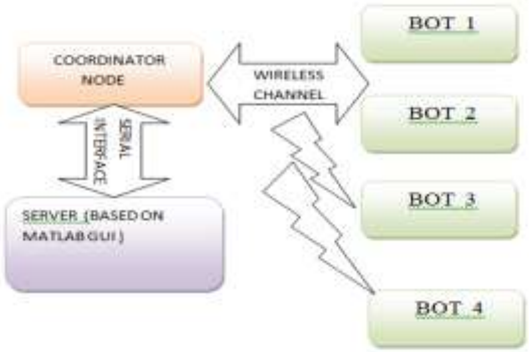
Abstract--**Being a nonrenewable and very efficient source of energy coal has been widely used to produce electricity; in many industries we use coal as a fuel also. Coal mining has been a very dangerous activity, underground mining hazards include gas explosions and suffocation which leads to death of coal miners every year and figures are very alarming. It may cause chronic lung disease and other respiratory disorders and shorten the life expectancy of miners. To overcome this problem we can introduce wirelessly controlled robots with the help of Graphical User Interface (GUI) of MATLAB, which are capable of doing mining and detecting the gas contents and temperature and send it back to the control room. Now the data which is now available in the receiving end determines the further action of robots. The whole process is cost effective and economically sound. Further these wireless robots have broad applications like in wireless home security applications, spy and war robots as they can make through in enemy areas just to track their activities. Other applications like in Nuclear Power Plant we can send them in Radioactive area to analyze things which is normally not possible for humans.**

Keywords**—WPAN,ZIGBBE, MATLAB GUI**

I. *INTRODUCTION*

The wireless network comprises of autonomous devices using sensors to effectively monitor physical or environmental parameters which play an important role in different industrial locations including industrial process monitoring, control and automation, environment and habitat monitoring etc. These may include various parameters such as temperature. Sound, vibration, pressure, motion or pollutants. Here we are more concern about application of such wireless robots using GUI with the help of MATLAB.We prefer GUI over other terminals (like Hyper Terminal, V1.9B Terminal and AVR Terminal etc.) because:

- GUI is user friendly interface as it allows user to interact with electronic devices with the help of images rather than text commands.
- As we are using it with MATLAB, we can plot the performance plot of (say temperature, concentration of several of gases etc.



Fig(1.0): Block diagram of wireless robot controlled through MATLAB using GUI

The above fig (1.0) represents the block diagram of wireless robot controlled through MATLAB using GUI. Here coordinator node has to receive the data which is transmitted by the different BOT's .then data is transferred to the PC using serial interface to the GUI . If the readings are above a certain value then the BOT's are manage to perform certain operation.

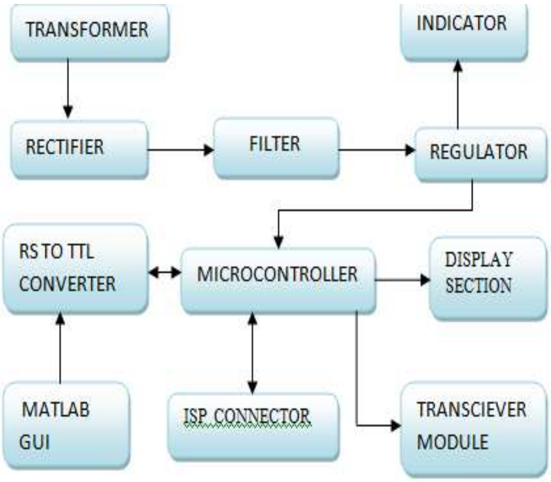
II. Hardware Development

Hardware of this wireless controlled bot through matlabgui system basically constitute of two parts: Transmitting or Slave End node design and receiving or coordinator node design.

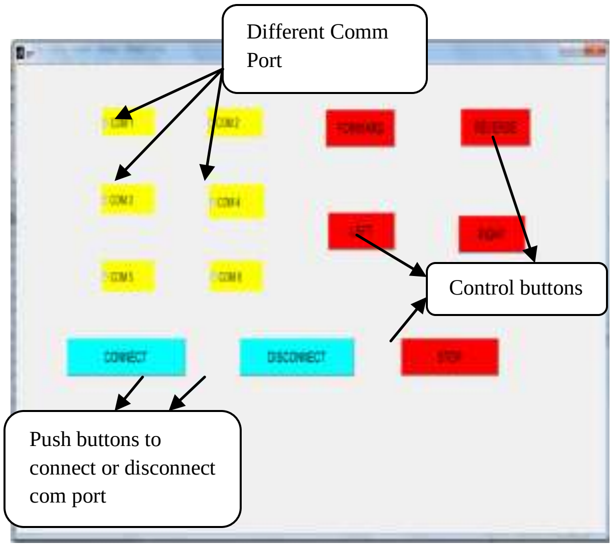
***(a)** Slave end node design*-Components of the slave system is given below

(1)Power supply modules-This module is basically designed to achieved 5V,500mA.Thisconsists of a transformer,IN4007 diodes, capacitor 1000uF,7805 regulator,330 ohm resistance, LED as indicator.

***(2)** Embedded microcontroller*-There is a whole wide range of microcontroller available in the market. But this particular project is developed using AVR series of microcontroller (ATMEGA16) because of its inbuilt ADC port and its variable frequency. ATmega16 is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega16 achieves throughputs approaching 1 MIPS perMHz, allowing the system designed to optimize powerconsumption versus processing speed. Further it also minimizes the cost of this personal area network.



Fig(2.0):Block diagram of Data transmitting section transmitting Section

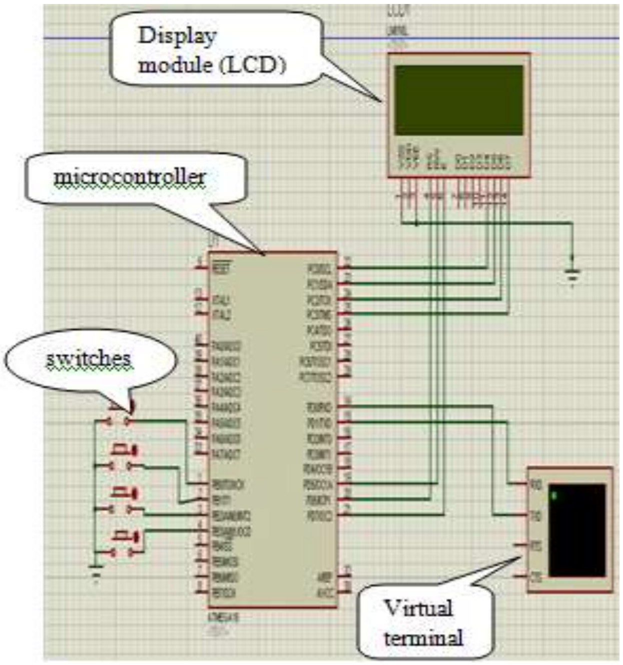


Fig(2.1):MATLAB GUI for controlling the system connected at transmitting end

***(3)** Zigbee module*-This is a single chip low power radio frequency transceiver module, which can facilitate the OEM designers to design their remote control applications in remote control in the quickest way. These modules are based on IC CC2500 by Texas Instrument. It works at frequency of 2.4GHz. The main operating parameters and the 64-byte transmit/receive FIFOs of CC2500 can be controlled with the help of an SPI interface. In a typical system, the CC2500 can be used together with a microcontroller and some passive components.

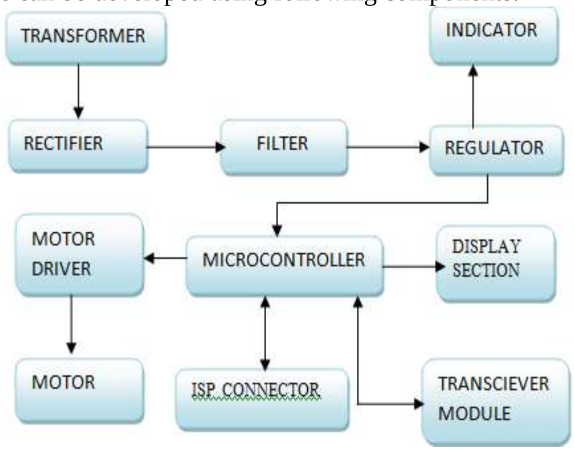
***(5)** Display module*-The LCD(liquid crystal display) unit receives character codes (8 bits per character) from a microprocessor or microcomputer, latches the codes to its display data RAM (80-byte) DD RAM for storing 80 characters, transforms each character code into a 5 ´ 7 dot-matrix character pattern, and displays the characters on its LCD screen.

Finally, the hardware model of transmitting end is simulated using Proteus software (trial). Data which is sensed by the alcohol sensor will be processed by the controller and displayed through LCD and corresponding data also send using TX (15) pin of microcontroller. The running data on TX line is displayed using Virtual terminal in Proteus software. Various indicators like Multimeters, LCD data Display, variable data meter and virtual data terminal are shown in figure (2.2).



(2.2) Simulation model of transmitting using Proteus trial software

(b) Coordinator node design -The hardware of this node can be developed using following components:

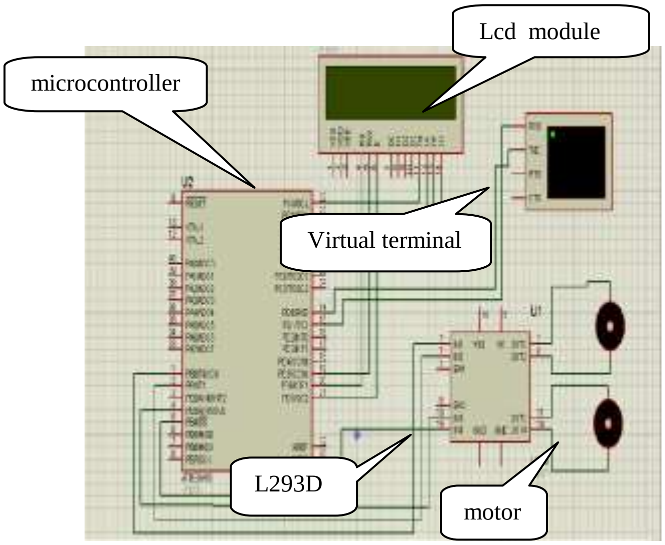


(2.3) :Block diagram of Data collecting section (Receiving Section)

(1) *Power supply unit*-It is same as describe described above.

(2) *Zigbee module*-Same as receiving module used in Node1 to Node 5 design.

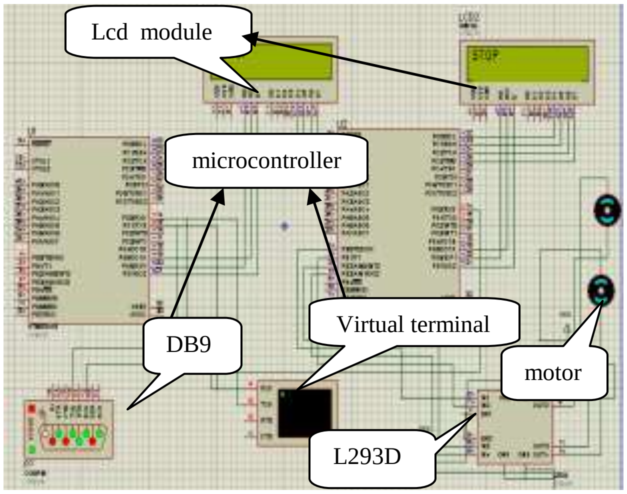
(3) MAX *232(level converter)*-MAX232 is a dual driver/receiver IC that includes a capacitive voltage generator to supply EIA-232 voltage levels from a single 5-V supply [2]. Each receiver converts EIA-232 inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V and a typical hysteresis of 0.5 V, and can accept ±30-V inputs. Each driver converts TTL/CMOS input levels into EIA-232 levels. This can be made to work with the help of a few capacitors attached to it.



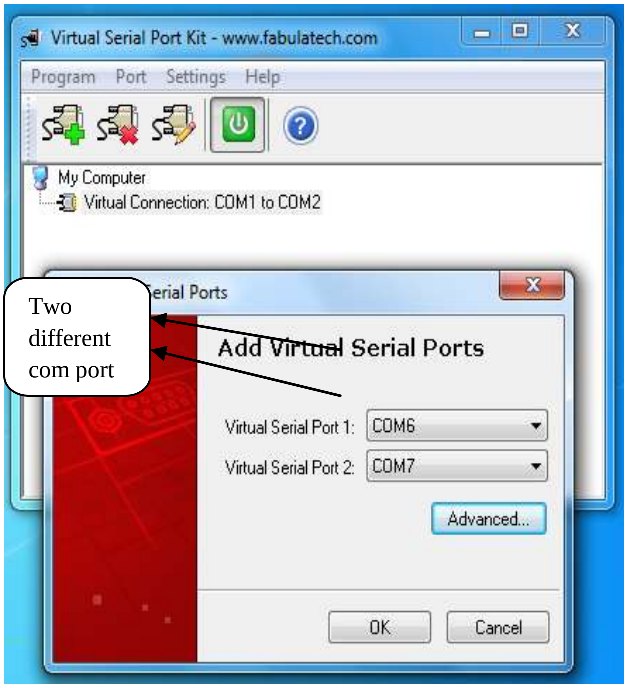
Fig(2.4): Simulation model of receiving end using Proteus trial software

(4) *USB to serial cable*-To interfaces the coordinator node with the other nodes.

(5)L293D(*motor driver IC*)-This IC is high voltage ,high current four channel driver designed to accept DTL or TTL logic.This can provide 600mA output current capability per channel and providing 1.2 peak output current(non repetitive) per channel and also have internal overtemprature protection . It consistes of a half H bridge to provide high current in order to drive motors.



Fig(2.5): Simulation model complete system using Proteus trial software


Virtual Serial Port Kit - www.fabulatech.com
Program Port Settings Help
My Computer
Virtual Connection: COM1 to COM2
Serial Ports
Add Virtual Serial Ports
Virtual Serial Port 1: COM6
Virtual Serial Port 2: COM7
Advanced...
OK
Cancel
Two different com port

Fig(2.6):Snapshot of virtual serial port used for connecting Proteus com port with MATLAB GUI

III. *SOFTWARE DEVELOPMENT*

Microcontroller, when it is used to operate as a wireless network involves following steps:

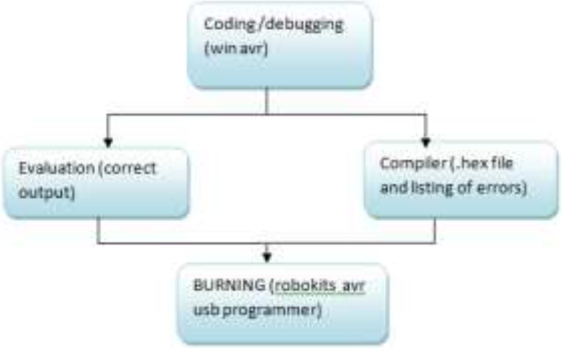

Coding /debugging (win avr)
Evaluation (correct output)
Compiler (.hex file and listing of errors)
BURNING (robokits avr usb programmer)

Fig (3.0): Steps for software development

(a) *Coding / Debugging*- Coding or debugging is one in a high-level language (such as c or java). Compiler for a high level language helps to reduce production time. To program the microcontrollers WinAVR [2] was used using C language. The source code has been commented to facilitate any occasional future improvement and maintenance. WinAVR is a suite of executable, open source software development tools for the Atmel AVR series of RISC microprocessors hosted on the Windows platform. It includes the GNU GCC compiler for C and C++. WinAVR contains all the tools for developing onthe AVR. This includes AVR-gcc (compiler), AVR-gdb (debugger) etc.

(b) Compiling-After compiling the program, it is converted to machine level language in the form of o's ans1's.This file is called as the Hex file and is saved with the extension (.Hex). The compiler also generates errors in the program which should be removed for proper execution of the program.

(c) Burning-Burning the machine language (hex) file into the microcontroller's program memory is achieved with a dedicated programmer, which attaches to a PC's peripheral. PC's serial port has been used for the purpose. for this purpose Ponyprog programmer was used to burn the machine language file into the microcontroller's program memory. Ponyprog is serial device programmer software with a user-friendly GUI framework available for Windows95/98/ME/NT/2000/XP and Intel Linux. Its purpose is reading and writing every serial device. It supports I²C Bus, Micro wire, SPI EEPROM, and the Atmel AVR and Microchip PIC microcontroller. The microcontrollers were programmed in approximately two seconds with a high speed-programming mode. The program memory, which is of Flash type, has, just like theEEPROM, a limited lifespan. On AVR microcontroller family it may be reprogrammed up to a thousand times without any risk of data corruption Atmega16 Programmer (ISP) which is used to burn the program into AVR microcontrollers.

(d) *Evaluation*-If the system performs as desired by the user and performs all the tasks efficiently and effectively the software development phase is over and the project is ready to be installed in any of the industrial sites as a personal area network. If not, the entire process is repeated again to rectify the errors. One of the difficulties of programming microcontrollers is the limited amount of resources the programmer has to deal with. In PCs resources such as RAM and processing speed are basically limitless when compared to microcontrollers. In contrast to a PC, the code on microcontrollers should be as low on resources as possible, but being cost effective and power efficient makes it a better option. In the programming of the proposed system is used the following .c and .h file

(1)lcd.c -This c file contains the code for control of functionality of the attached LCD module. The code controls the initialization of the LCD, data writing on the LCD, and also the movement, characteristics and location of the cursor. It offers the facility to write data on the LCD character-by-character or string-wise. The command set used in the software is based on the command set used in the LCD based on

Hitachi HD44780 ICs. This file contain INitlcd( ), remove ( ), display ( ) and displayint ( ).

(i) *to initialize the LCD:*

```
Void INitlcd( )
{
This function Initializes the lcd module
    must be called before calling lcd related functions

    Arguments:
    style = LS_BLINK,LS_ULINE(can be "OR"ed for combination)
    LS_BLINK :The cursor is blinking type
    LS_ULINE :Cursor is "underline" type else "block" type
}
```

(ii) to display to LCD :

```
Void display( const char  *data)
{
This function Writes a given string to lcd at the current cursor
    location.
Arguments:
    msg: a null terminated string to print}
```

(2) lcd.h- This header file contains all the constant variable values and names of the subroutines used by various files used in the software. It clearly indicates which variable can be used as a global variable and which of the subroutines can be used across the software files.

(3) Usart_lib.c- This file contains the code for controlling the USART of ATMEGA'S. This is contain three major functions USARTInit ( ), USARTReadChar ( ) and USARTWriteChar ( ).

Initialization of USART:

This function will initialize the USART.

```
voidUSARTInit(uint16_t ubrr_value)
{
UBRR= ubrr_value;  //Set Baud rate
UCSRC=(1<<URSEL)|(3<<UCSZ0);// Set Frame Format
UCSRB=(1<<RXEN)|(1<<TXEN);// //Enable The receiver and transmitter
}
```

Reading From The USART :

This function will read data from the USART.

```
char USARTReadChar()
{
   while(!(UCSRA & (1<<RXC))) //Wait until a data is available
   {
   //Do nothing
   }
return UDR;   //Now USART has got data from hostand is available is buffer
}
```

Writing to USART:

```
void USARTWriteChar(char data)
{
while(!(UCSRA & (1<<UDRE)))   //Wait until the transmitter is ready
 {
   //Do nothing
  }
UDR=data;   //Now write the data to USART buffer
}
```

(4)** functions used in program **: The code which is used to program the controller include some functions as :

***(i)** to provide delay in the program*

```
Void delay (unsigned char value)
{
For(unsigned inti=0;i<value;i++)
     {
      _delay_ms(1);
     }
}
```

(ii) *controlling of the motor*

```
Void motor(char data)
{
Switch(data)
{
Case 'a': motor forward;
            Break;
Case 'b': motor reverse;
            Break;
Case 'c': motor left;
           Break;
Case 'd': motor right;
           Break;
Default: motor stop;
           Break;
 }
}
```

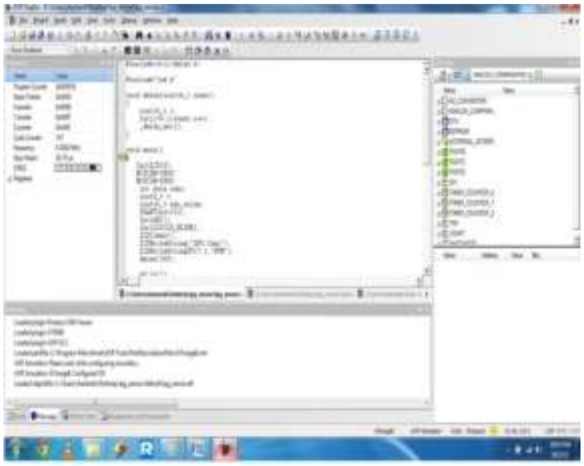
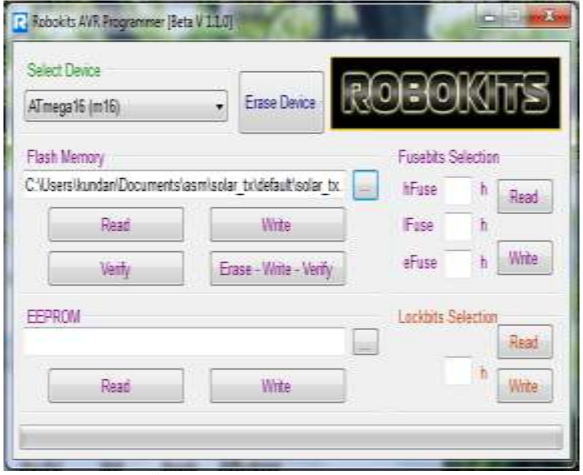


Fig (3.1): snapshot C coding for proposed system using of AVRstudio4



Robokits AVR Programmer [Beta V 1.1.0]

ATmega16 (m16)

C:\Users\kundan\Documents\asm\solar_tx\default\solar_tx

Fig (3.2): view of hardware AVR programmer (ISP) by Robokits

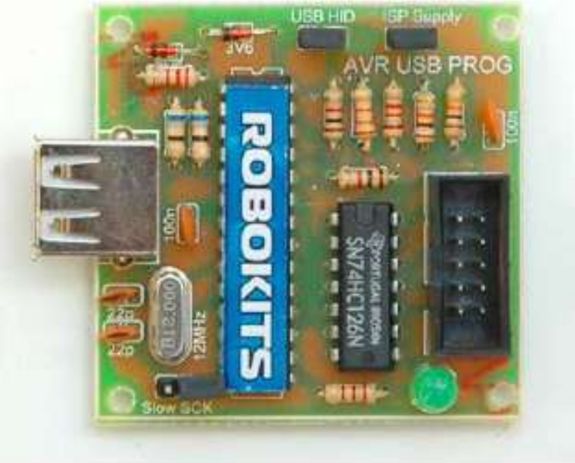


Fig (3.3): snapshot of robokits AVR programmer window to erase, write and verify .h file generated by AVRstudio4 to target

(e) *Development of GUI using MATLAB :*

To develop a GUI in MATLAB open the matlab and open new GUI file(.fig) and select the buttons and place them in the .fig file after running this window a window will open in which code for the working of GUI is written . After writing code for every buttons that you were placed in .fig file run that window after this a new window will open which contains the GUI through which we can use our hardware and control it .

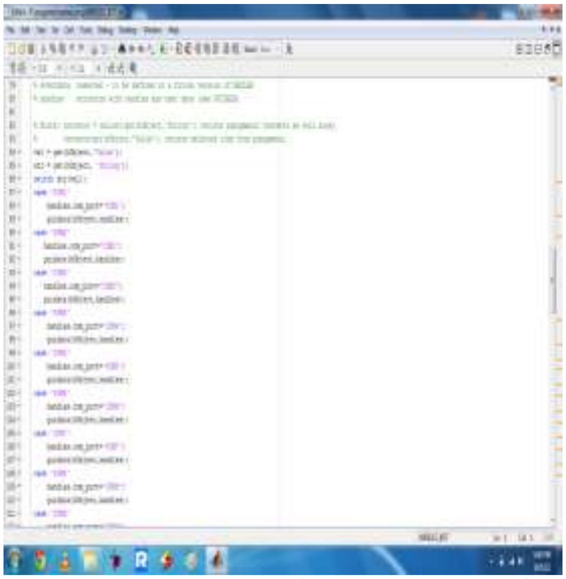


Fig (3.4): snapshot MATLAB coding for proposed GUI using of MATLAB 7.11.0.584 (trial version)

View of system

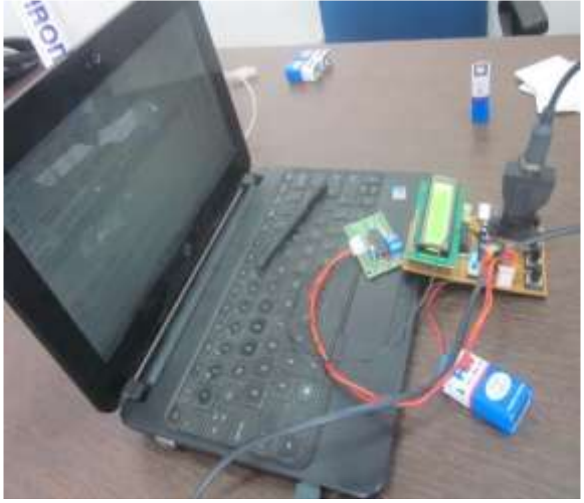


Fig (3.5): View of transmitting side of the proposed system.

Fig (3.6): View of receiving side of the proposed system

IV. CONCLUSION

In this paper, a wireless area network is designed using various hardware and software (GUI of MATLAB and Embedded C) techniques which are user friendly and economically sound. This project has wide range of applications in coal mines, nuclear power plants, spy and war robots etc. As the interface is Graphic based due to which it is very easy to use and easily accessible also. Now further advancement in proposed system can improve the working efficiency of robots by rectifying their intelligence level.

REFERENCES

[1] Sanjay Singh Rajesh Singh and Su kumar Ray chaudhri"DESIGN OF NODES IN WIRELESS SENSOR NETWORK USING RF MODULE " in international journal Association of advancement in combinatorial Sciences Ref.: ITAS-2-4/10 Date: 15-12-10 .

[2] Rajesh Singh VivekKaundal and Sanjay Singh, "AUTOMATIC METER READING IN WSN USING 2.4ghz TRANCEIVER WITH MATLAB DATA LOGGER" in ""International Transactions in Mathematical Sciences and Computer(ITMSC)" (ISSN-0974-7273) .

[3] Rajesh Singh, Akanksha, Shashank Mishra, Ankit Joshi " WIRELESS PERSONAL AREA NETWORK NODE DESIGN AND SIMULATON OF ALCOHOL SENSOR USING ZIGBEE TRANSCEIVER MODULE" International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 3, May-Jun 2012, pp.133-138

[4] SINGH R, MISHRA S, JOSHI P, "Wireless pressure monitoring in wireless sensor network using 2.4 GHz transceiver module" " ICCCT 2011, IEEE conference on computer and communication technology " at MNNIT Allahabad ISBN: 978-1-4577-1385-9, Page no 225-229

[5] Rajesh Singh, VivekKaundal, Arpit Jain "WIRELESS PERSONAL AREA NETWORK DESIGN AND SIMULATION TO LOCATE ACCIDENTAL INFORMATION USING 2.4 GHZ TRANSCEIVER MODULE" [IJESAT] INTERNATIONAL JOURNAL OF ENGINEERING SCIENCE & ADVANCED TECHNOLOGY ISSN: 2250–3676, Volume-2, Issue-2, 297 – 300

[6] Rajesh Singh, Vinod Kumar, PankajJoshi"WPAN node design for Gathering Real Time Data through V1.9b terminal using 2.4 GHz Transceiver Module" "International Journal of Electronics and Computer Science Engineering ", ISSN- 2277-1956