PWM Based Speed Control of DC Geared Motor Using 802.15.4 LAN Standard Protocol

[1]Jeetender Singh Chauhan, [2]Dr. Rajesh Singh, [3]Sunil Semwal, [4]Shashank Mishra
[1]Dept. of Instrumentation & Control Engg., Graphic Era University, Dehradun, Uttarakhand, India
[2]Institute of Robotics Tech., Univ. of Petroleum and Energy Studies, Dehradun, Uttarakhand, India
[3]Dept. of Electrical and Electronics Engg., Graphic Era University, Dehradun, Uttarakhand, India
[4]Dept. of EE & I, University of Petroleum and Energy Studies, Dehradun, Uttarakhand, India

Abstract
The D.C Motor is widely used in industrial applications in various configurations i.e. D.C Shunt Motor, D.C Compound Motor, D.C Series Motor. This paper gives a brief insight about wireless speed Control of a D.C Motor so that it can be used more efficiently in industrial applications. The setup consists of a transmitter which generates control signals which are wirelessly transmitted to the receiver. The receiver according to the control signals controls the speed of the D.C motor provided by Pulse Width Modulation (PWM).The control signals are transmitted using RF wireless module Zigbee. Hence forth the speed of the D.C Motor can be controlled wirelessly through a control room which makes the system sustainable.

Keywords
D.C Motor, Wireless Speed Control, Pulse Width Modulation, Zigbee, Sustainable

I. Introduction

The applications of D.C motor range from household products i.e. vacuum cleaners, hair driers to Industrial applications i.e. reciprocating machines, presses shears. As the applications of D.C Motor are vivid there is a need of a sustainable system which can be able to utilize the same D.C motor for various applications. This paper provides a system that can utilized to use D.C motor for various applications. We can utilize the D.C Motor for various applications by controlling the speed and orientation according to the field of interest.The system consists of a Transmitter module which generates control signals which contains information about the speed and orientation of motor depending on the application area. These signals are transmitted to the Receiver wirelessly through wireless RF module Zigbee. The receiver according to the control signals controls the speed of the motor by Pulse Width Modulation (PWM).

Pulse Width Modulation (PWM) is the technique of utilizing switching devices to produce the effect of a continuously varying analog signal. This PWM conversion generally has very high electrical efficiency and can be used in controlling either a three-phase synchronous motor or a three-phase induction motor .It is desirable to create three perfectly sinusoidal current waveforms in the motor windings, with relative phase displacements of 120°. The production of sine wave power using a linear amplifier system would have low efficiency, maximum of 64%. Efficiency can be increase up to 95% if instead of the linear circuitry, fast electronic switching devices are used, depending on the properties of the semiconductor power switch.The result is a load current waveform that depends mainly on the modulation of the duty ratio.

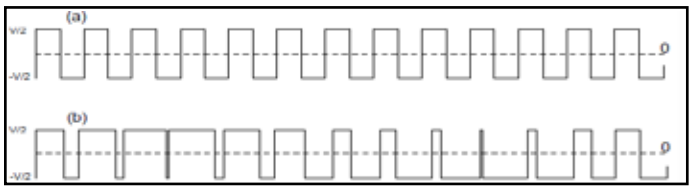


Fig 1: Wave Form at Different Duty Cycles

II. Hardware Development

Hardware of this wireless 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.This consists of a transformer which is used to step down the AC voltage, IN4007 diodes used to form a bridge rectifier to convert AC to DC, capacitor 1000uF which used as a filter circuit, 7805 regulator to obtain a 5V at the output of the regulator, 330 ohm resistance, LED as indicator.

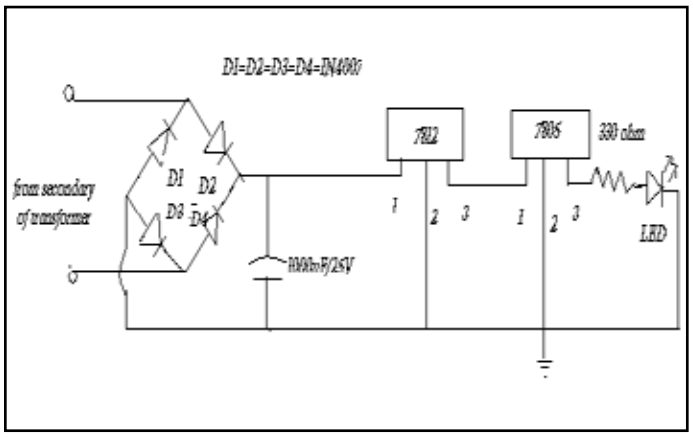


Fig. 2: Diagram of Supply Section

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 per MHz, allowing the system designed to optimize power consumption versus processing speed. Further it also minimizes the cost of this personal area network.

3. Zigbee Module (Transmitting Module (RF Modem, 9600 bps, TTL Logic)

It is a low power and low cost 2.4 GHz transceiver designed for wireless applications. 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. 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.We are 16*2 LCD 's which have 16 columns and 2 rows with 16 hardware pins connected as pin 1,3and 16 are connected to ground , pin 2 and 15 are connected to +5v pin 3 ,4,5 are RS ,RW and enable respectively enable pin is always low . data pins of LCD are 11,12,13,14 which are used for 4 bit parallel communication.

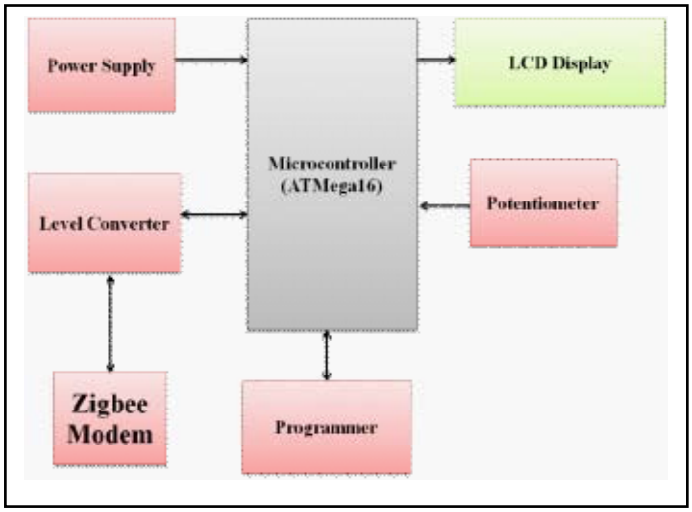


Fig. 3: Block Diagram of Transmitter Section

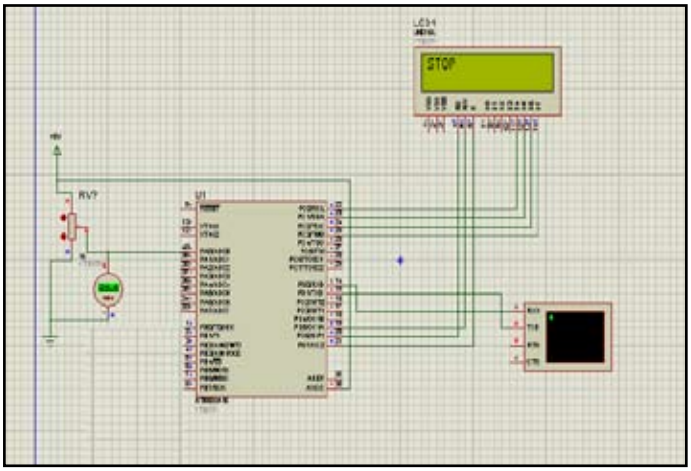


Fig. 4: Simulation Diagram of Transmitter

B. Slave Section

1. Power Supply Unit

It is same as describe described above.

2. Zigbee Module (Receiving module (RF Modem, 9600 bps , TTL logic)

It is a low power and low cost 2.4 GHz transceiver designed for wireless applications. 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. 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.

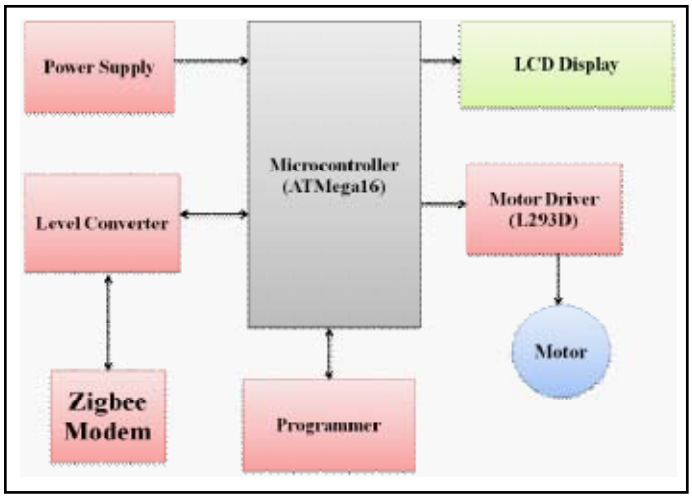


Fig. 5: Block Diagram of Receiving Section

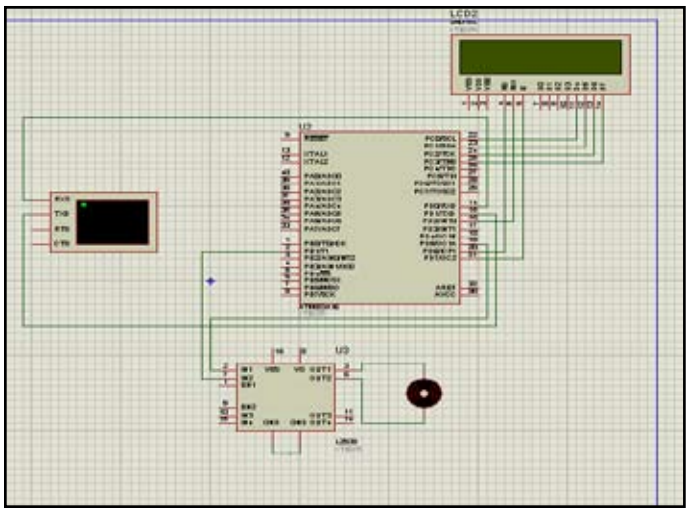


Fig. 6: Simulation Diagram of Receiving Section

3. MAX232 (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.

4. 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 over temprature protection . It consistes of a half H bridge to provide high current

in order to drive motors.

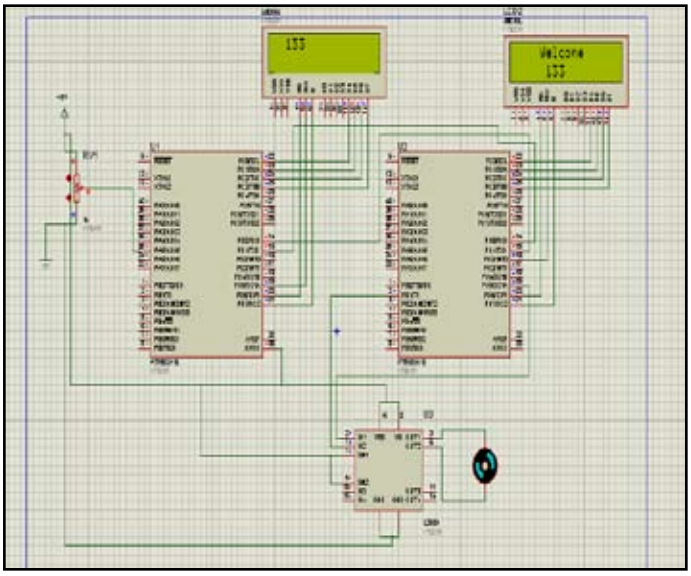


Fig. 7: Complete Simulation of Model

III. Software Development

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

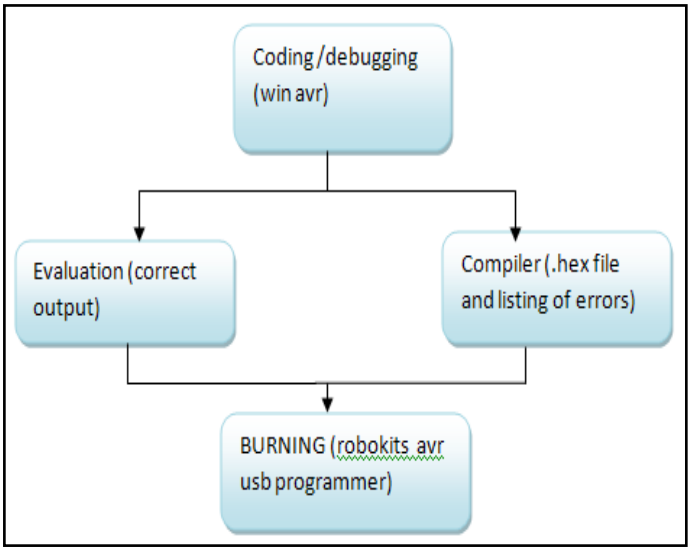


Fig. 8: Steps for Software Development

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 on the 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 the EEPROM, 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 strings 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.

```
void USARTInit(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 host and 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. Adc.c

This file contains the code for controlling the ADC of ATMEGA'S. This is contain two major functions initializeADC( ), int ReadADC(uint8_t ch). This helps us to read various sensors .

(i)-Initialization of ADC:

```
Initialize ADC()
  {
   ADMUX=(1<<REFS0);// For Aref=AVcc;
   ADCSRA=(1<<ADEN)|(7<<ADPS0);
  }
```

(ii) Read data from ADC:

```
Int ReadADC(uint8_t ch)
  {
//Select ADC Channel ch must be 0-7
//Start Single conversion
 //Wait for conversion to complete
//Clear ADIF by writing one to it
return(ADC);
}
```

5. 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 int i=0;i<value;i++)
      {
       _delay_ms(1);
    }
}
```

(ii) controlling of the motor

```
Void motor(char data)
{
Switch(data)
{
Case 'a': motor at 100%;
          Break;

Case 'b': motor at 75%;
          Break;

Case 'c': motor at 50%;
          Break;

Case 'd': motor at 25 %;
          Break;

Default: motor at 0%;
          Break;
 }
}
```

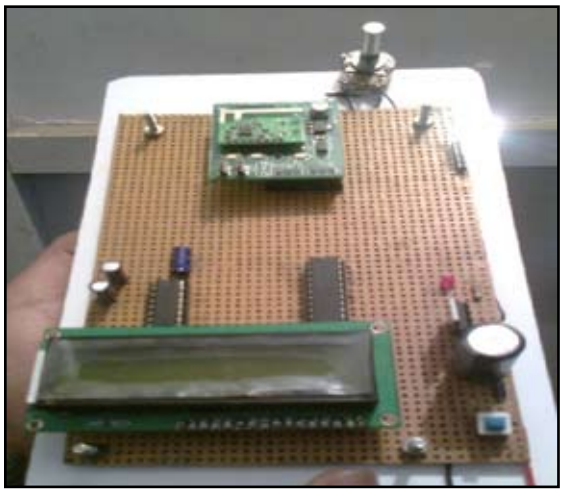


Fig. 9: View of Transmitting Section



Fig. 10: View of Receiving Section



Fig. 11: View of Complete System

IV. Conclusion

The designing of a sustainable system to control the speed and orientation of a geared D.C Motor was successfully implemented in this paper. The system is capable of controlling the speed of a remotely located D.C motor from a control room wirelessly, consequently can be realized for a variety of application. The paper provides a platform for further advancement in the field of industrial use of D.C motors.

References

[1] Sanjay Singh Rajesh Singh, Su kumar Ray Chaudhri "Design of Nodes in Wireless Sensor Network Using RF Module ", In international journal Association of advancement in combinatorial Sciences.

[2] Rajesh Singh Vivek Kaundal, Sanjay Singh, "Automatic Meter Reading in WSN Using 2.4ghz Tranceiver With Matlab Data Logger", in International Transactions in Mathematical Sciences and Computer (ITMSC)".

[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) [Online] Available: http://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, pp. 225-229

[5] Rajesh Singh, Vivek Kaundal, Arpit Jain,"Wireless personal area network design and simulation to locate accidental information using 2.4 ghz transceiver module", international journal of engineering science & advanced technology, Vol. -2, Issue 2, pp. 297 – 300

[6] Rajesh Singh, Vinod Kumar, Pankaj Joshi,"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,



Dr.Rajesh Singh received his B.E degree in Electronics & Communication from Dr. BR Ambedkar University Agra, India. M.Tech (Gold Medalist) in Digital Communication from Rajeev Gandhi Technical University Bhopal, India & also completed Ph.D in wireless personal area network design and simulation using zigbee transceiver module. He is currently attached with University of Petroleum in Energy Studies Dehradun, India in the department of Electrical , Electronics & Instrumentation department and Head Institute of Robotics technology.Dr. Rajesh Singh Has published 29 papers in national/ international conferences/ journals. He is also the reviewer of ICEOE conference.



Jeetender Singh chauhan received his B.Tech degree in Electronic & Communication from Sagar Institute Of Technology and Management Barabanki U.P., India and pursuing M.Tech. in Instrumentation and Control Engineering from Graphic Era University Dehradun, Uttrakhand, India.



Sunil Semwal received his B.Tech degree in Instrumentation Engineering from University Science and Instrumentation Centre Srinagar Garhwal, Uttrakhand, India and M.E. degree in Electronics from Punjab Engineering College, Chandigarh, Punjab, India. Currently he is working as Assistant Professor in Electrical and Electronics Engineering department at Graphic Era University Dehradun, Uttrakhand, India. He is presented 11 papers in national/ internatioal conferences/journals.



Shashank Mishra Pursuing his B.Tech in Electronics Engineering from University of Petroleum and Energy Studies Dehradun, india. He has presented 3 papers in national/international conferences/journal.