A
Project Report
on
" Gesture Controlled Wheelchair Using ATmega32"

Prepared by

Rahul Suhagiya (17EC091)

Under the guidance of

Dr. Sagar B. Patel

Prof. Dhara M. Patel

Submitted to

Charotar University of Science & Technology

for Partial Fulfillment of the Requirements for the

Degree of Bachelor of Technology

in Electronics & Communication

EC448 - Project

of 4th Semester of B.Tech

Submitted at

DEPARTMENT OF ELECTRONICS & COMMUNICATION

Faculty of Technology & Engineering, CHARUSAT

Chandubhai S. Patel Institute of Technology

At: Changa, Dist: Anand – 388421

April 2021

CERTIFICATE

This is to certify that the report entitled "Gesture Controlled Wheelchair Using ATmega32" is a bonafide work carried out by **Rahul Suhagiya(17EC091)** under the guidance and supervision of **Dr. Sagar B. Patel** & **Prof. Dhara M. Patel**for the subject **Project (EC448)** of 4th Semester of Bachelor of Technology in Electronics & Communication at Faculty of Technology & Engineering (C.S.P.I.T.) – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate himself, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Under the supervision of,

Dr. Sagar B. Patel
Associate Professor
Department of EC Engineering,
C.S.P.I.T., CHARUSAT-Changa.

Prof. Dhara M. Patel
Assistant Professor
Department of EC Engineering,
C.S.P.I.T., CHARUSAT-Changa.

Dr. Trushit K. Upadhyaya
Head of Department,
Department of Electronics & Communication
C.S.P.I.T., CHARUSAT- Changa, Gujarat.

Chandubhai S Patel Institute of Technology (C.S.P.I.T.)

Faculty of Technology & Engineering, CHARUSAT

At: Changa, Ta. Petlad, Dist. Anand, Gujarat - 388421

ABSTRACT:

The aim of this project is to prepare a Gesture Controlled Wheelchair for the physically disabled people who face difficulty in moving from one place to another in day today life. It is useful for physically disabled person with his hand movement with the help of hand gesture recognition. The system is classified into gesture unit and wheelchair unit. This wheelchair can be control by simple hand gestures. Depending on the direction of the Acceleration, AVR controller controls the wheelchair ways like LEFT, RIGHT, FORWARD, & BACKWARD.

Application:

- Amputees - missing legs and/or arms.
- People with weak or no upper body movement.
- Paralyzed children.

Advantages:

- Can be used as an assistance for physically challenges people.
- Useful for moving heavy loads from one place to another place.

Disadvantages:

- If power supply fails system won’t work.

ACKNOWLEDGEMENT

I take this opportunity to express my profound gratitude and deep regards to my guide **Dr. Sagar B. Patel** and **Prof. Dhara M. Patel** and coordinator of E&C department of CSPIT, **Dr. Trushit Upadhyaya**, for their exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by them time to time shall carry me a long way in the journey of life on which I am about to embark.

I also take this opportunity to express a deep sense of gratitude to Mentor **Prof. Jignesh J. Patoliya** for their cordial support, valuable information and guidance, which helped me in completing this task through various stages.

Rahul Suhagiya (17EC091)
CSPIT, CHARUSAT

TABLE OF CONTENT

List of Figures

Chapter1: Introduction of Project

- Wheelchairs are useful for people for whom walking is difficult or impossible due to some illness, injury or disability. There are different types of wheelchairs. Here we present the project to control a wheelchair through finger gestures. It uses flex sensors to record finger gestures. The gesture signals are fed to ATmega32 microcontroller, which processes the information to control four DC motors in order to move the wheelchair in any desired direction.

1.1: Improvements in available technology:

- Voice-controlled wheelchairs are the latest development. These can be driven just by giving voice commands. A more advanced and intelligent version of the wheelchair is controlled directly through human mind, such as the one used by renowned scientist Stephen Hawking. In case a person is unable to move the wheelchair even with joystick or voice command, an alternative is gesture-controlled wheelchair.

1.2: Project Overview:

- Motorized wheelchairs, are driven by joystick. ... The gesture signals are fed to ATmega32 microcontroller, which processes the information to control four DC motors in order to move the wheelchair in any desired direction. Flex sensors are widely used to convert finger gestures into equivalent electrical signals.

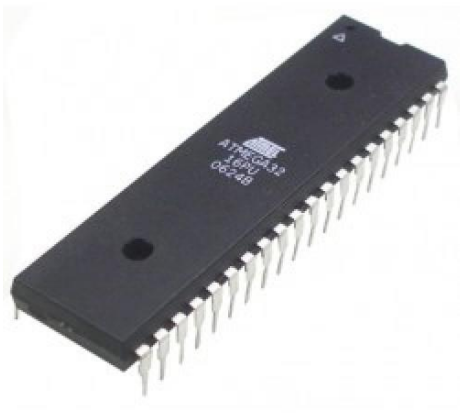

Figure 1: Atmega32A Microcontroller

Chapter 2: Project Description

This Project is using less hardware component but many software. All the function played by each hardware & software is described below:

2.1: Hardware Description:

ATmega32 microcontroller:

Takes flex sensor input from the internal analogue-to-digital controller (ADC) and checks whether it exceeds certain threshold limit when the sensor is bent. Rotates right and left motors in forward or reverse direction to move the wheelchair in the desired direction. Displays wheelchair movement on the LCD and also displays the direction of the wheelchair through LED indicators.

L293D driver:

It provides sufficient voltage and current for DC motors to drive the wheelchair.

2.2: Software Description:

Atmel Studio: Atmel Studio is the integrated development platform (IDP) for developing and debugging all AVR microcontroller applications. The Atmel Studio IDP gives you a seamless and easy-to-use environment to write, build and debug your applications written in C/C++ or assembly code.

2.3: Block Diagram:

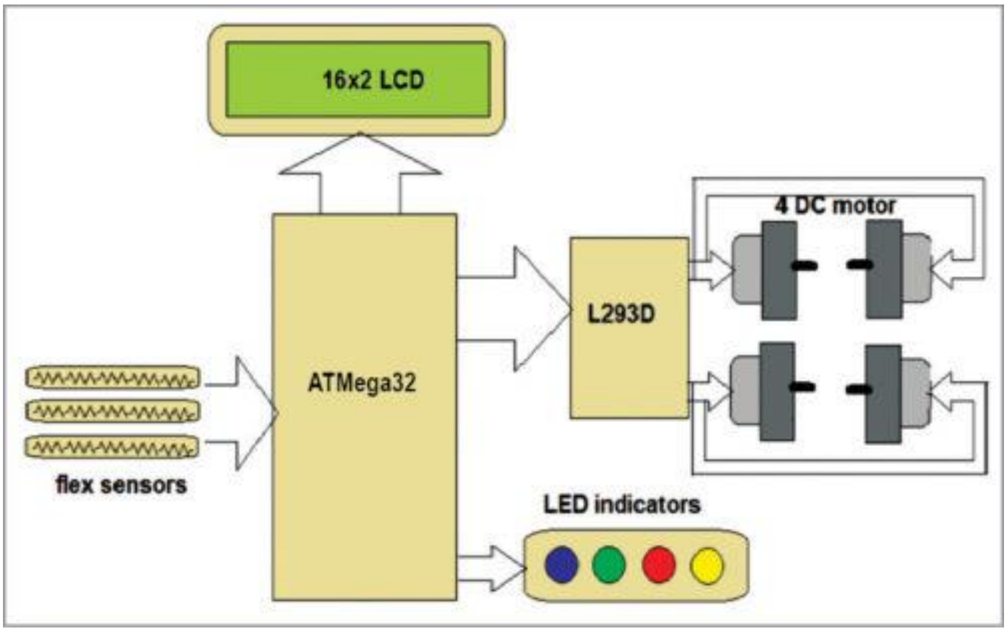

Figure 2: Block Diagram

2.4: Circuit Diagram:

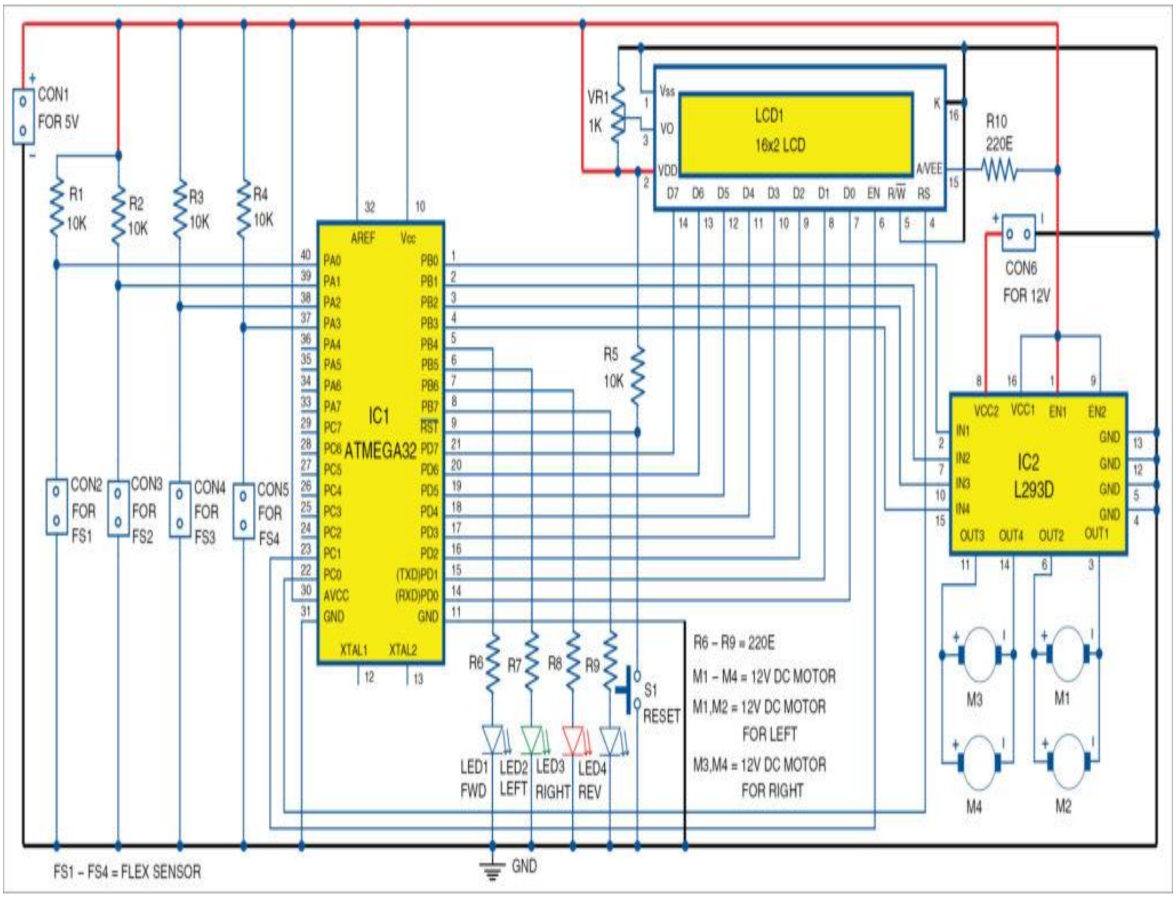

Figure 3: Circuit Diagram

Chapter 3: Component Details

Here list of all components with their specification is provided.
each & every component gives all the data of a component so please refer it.

3.1: Component List:

- Atmega32A Microcontroller
- L293D Driver Ic
- DC Motor
- Flex Sensor
- LCD

3.2: Component Technical Specifications:

Atmega32A Microcontroller Specifications:

- 32K bytes of in system self programmable flash program memory
- 8 Channel, 10 bit ADC
- 32 programmable I/O lines
- Programmable serial USART
- Master/slave SPI serial interface
- two 8-bit timers/counters with separate prescalers and compare modes
- 4 PWM channels
- 1024 bytes EEPROM

L293d Driver IC Specifications:

• Motor Driver:	L293D
• Maximum Operating Voltage:	36V
• Peak Output Current per Channel:	1A
• Minimum logic voltage:	4.5V
• Maximum logic voltage:	7V

Chapter 4: Implementation of Project

The complete hardware setup should be done as per instruction given below because all the C code has been written as per that configuration only. Copy the same code as provided because it will only work with the hardware setup we have done before. So, strictly follow the hardware & software setup instruction

4.1: Hardware Implementation:

- First, Load the program in Atmega32A via Loader.
- After load the load then all motor, Driver IC, And LCD connections With Atmega32A.
- Then, Connecting all the components with Atmega32A give the 12V in Driver IC And Connect All motor with IC.
- Connect the 5V power supply with Atmega32A microcontroller.

4.1: Software Implementation:

```
#include <avr/io.h>
#include <util/delay.h>
#include <string.h>
 void senddata(unsigned char data)
 {
        _delay_ms(2);
        PORTC |= 0x01;
        PORTD=data;
        PORTC |= 0x02;
        PORTC &= 0xFD;
 }
void sendcmd(unsigned char cmd)
 {
        _delay_ms(2);
        PORTC &= 0xFE;
        PORTD=cmd;
        PORTC |= 0x02;
```

```
        PORTC &= 0xFD;
 }
void printstr(char *s)
{
        uint8_t l,i;
        l = strlen(s);                  // get the length of string
        for(i=0;i<l;i++)
        {
                senddata(*s);                   // write every char one by one
                s++;
        }
}
void lcd_init()
 {
         DDRD = 0xFF;
         DDRC = 0x03;
         PORTC = 0x00;
         PORTD = 0x00;
         sendcmd(0x3E);
         sendcmd(0x0E);
         sendcmd(0x06);
         sendcmd(0x01);
         printstr("Gesture Control");
         sendcmd(0xC3);
         printstr("Wheel Chair");
         _delay_ms(2000);
 }
void adcinitliz()
 {
        ADMUX = (1<<REFS0);
        // select reference for ADC
        ADCSRA =(0<<ADPS2) | (1<<ADPS1) | (1<<ADPS0);  // select clock s

 }
unsigned int convert()
 {
        unsigned int tmp1,tmp2,tmp3,t;
        tmp1 = (ADCL & 0x0F);
        tmp2 = (ADCL >> 4) & 0x0F;
        tmp3 = (ADCH & 0x0F);
        tmp2 = tmp2*16;
        tmp3 = tmp3*256;
        t = tmp1+tmp2+tmp3;
        return t;
 }

void main()
 {
```

```
        unsigned int
adc_value,ch_count=0,fwd_flag=0,rev_flag=0,right_flag=0,left_flag=0;
        unsigned int
msg1_flag=0,msg1_count=0,msg2_flag=0,msg2_count=0,msg3_flag=0,msg3_c
ount=0,msg4_flag=0,msg4_count=0,msg5_count=0;
        DDRB = 0xFF;
        PORTB = 0x00;
//      TCCR0=0x6B;
   lcd_init();
        adcinitliz();
        while(1)
         {
                ch_count++;
                if(ch_count==1) ADMUX = 0x40 ;
                else if (ch_count==2) ADMUX = 0x41 ;
                else if (ch_count==3) ADMUX = 0x42;
                else if (ch_count==4) ADMUX = 0x43;
                ADCSRA = (1<<ADEN) | (1<<ADSC);
                while(!(ADCSRA & (1<<ADIF)));
                ADCSRA = (1<<ADIF);
                adc_value = convert();
                if(ch_count==1)
                 {
                        if(adc_value>820)
                         {
                                PORTB = 0x15;
                                fwd_flag=1;
                                msg1_flag = 1;
                         }
                        else
                                {fwd_flag=0;msg1_count=0;msg1_flag=0;}

                  }
                else if(ch_count==2)
                 {
                        if(adc_value>820)
                         {
                                PORTB = 0x21;
                                left_flag=1;
                                msg2_flag=1;
                         }
                        else {left_flag=0;msg2_count=0;msg2_flag=0;}
                 }
                else if(ch_count==3)
                 {
                        if(adc_value>820)
                         {
                                PORTB = 0x44;
```

```
                        right_flag=1;
                        msg3_flag=1;
                    }
                else {right_flag=0;msg3_count=0;msg3_flag=0;}
            }
            else if(ch_count==4)
            {
                if(adc_value>820)
                {
                        PORTB = 0x8A;
                        rev_flag=1;
                        msg4_flag=1;
                }
                else {rev_flag=0;msg4_count=0;msg4_flag=0;}
                ch_count=0;
            }
            if((fwd_flag==0) && (rev_flag==0) && (left_flag==0) &&
(right_flag==0))
            {
                PORTB = 0x00;
                msg1_flag=0;
            }
            if(msg1_flag==1)
            {
             if(msg1_count==0)
              {
                {
                        sendcmd(0x01);
             printstr("Wheel Chair is");
             sendcmd(0xC0);
             printstr("moving forward");
                        //msg1_flag=0;
                        msg1_count++;
                        msg5_count=0;
                }
            }
    }
            else if(msg2_flag==1)
            {
             if(msg2_count==0)
              {
                {
                        sendcmd(0x01);
             printstr("Wheel Chair is");
             sendcmd(0xC0);
             printstr("turning left");
                        //msg2_flag=0;
                        msg2_count++;
```

```
                    msg5_count=0;
                }
            }
        }
        else if(msg3_flag==1)
            {
             if(msg3_count==0)
              {
                {
                    sendcmd(0x01);
             printstr("Wheel Chair is");
             sendcmd(0xC0);
             printstr("turning right");
                    msg3_count++;
                    msg5_count=0;
                }
            }
        }
      else if(msg4_flag==1)
            {
             if(msg4_count==0)
              {
                {
                    sendcmd(0x01);
             printstr("Wheel Chair is");
             sendcmd(0xC0);
             printstr("moving reverse");
                    //msg4_flag=0;
                    msg4_count++;
                    msg5_count=0;
                }
            }
        }else
            {
                if(msg5_count==0)
                {
                    sendcmd(0x01);
            printstr("Wheel Chair");
            sendcmd(0xC0);
            printstr("is stop");
                    msg5_count++;
                }
            }
        }
    }
```

Chapter 5: Application of Project

- Automated Wheelchair can be operated by a wireless remote which can the reduce wiring arrangements.
- We can use voice command IC’S to interface our voice signal with microcontroller.
- This system can be extended by GSM which sends an SMS during emergency.
- Amputees - missing legs and/or arms.
- People with weak or no upper body movement.

Conclusion

- While working on this project, we learnt to interface Atmega32A with the motor and Driver IC. So, it makes easier to learn interfacing of other components with Atmega32A microcontroller when needed in future. We also learnt to troubleshoot many issues related to hardware & software. Many facts came to be known related to all this component while working on this project.

- While working on this project We have gained a lot of hands-on experience with this microcontroller. We think that experience of working on Atmega32A programming will help us in future to operate on another microcontroller programming. We also learned to use ADC and LCD programming in Atmega32A which is use in many applications.

References

Special Thanks To:

https://electronicsforu.com/electronics-projects/gesture-controlled-wheelchair-atmega32/2
https://www.researchgate.net/publication/316551968_Hand_Gesture_Controlled_Wheelchair

Data Sheets:

https://www.alldatasheet.com/view.jsp?Searchword=Atmega32a%20datasheet&gclid=EAIaIQobChMIprja1uaj5QIVwTUrCh0CQQweEAAYASAAEgJBr_D_BwE

https://binaryupdates.com/learn-basics-avr-microcontroller/