Journal of Advanced Research in Power Electronics and Power Systems
Volume 10, Print Issue 1 - 2023, Pg. No. 1-4
Peer Reviewed Journal

Review Article

Advancing High-Speed Stepper Motor Control: A PC-Based Approach with Wired Communication

Arun Prakash

EC Department, Assam Science and Technology University.

I N F O

E-mail Id:
ap0005784@gmail.com
Orcid Id:
https://orcid.org/0009-0006-0828-9816

How to cite this article:
Prakash A. Advancing High-Speed Stepper Motor Control: A PC-Based Approach with Wired Communication. *J Adv Res Power Electro Power Sys* 2023; 10(1): 1-4.

Date of Submission: 2023-05-10
Date of Acceptance: 2023-06-12

A B S T R A C T

Modern society's ubiquitous household goods and devices use stepper motors in a wide range of applications. In numerous industrial driving applications, the problem of sensing variables for control, such as position, velocity, or current, commonly arises. Given the high cost or environmental restrictions, it may be challenging to detect signals that precisely represent system variables, such as the absolute shaft position. In cases like this, we are compelled to estimate all or a portion of the missing variables using a sparse sample of data that could contain noise. The purpose of this project is to plan and construct a circuit that employs a microcontroller to fulfil the task of controlling the stepper motor through a keypad. As a result, the microcontroller's circuit can regulate both the step angle and the speed of the stepper motor. whenever the user types one of the keypad's many commands. This project describes in full how to link an extremely sensitive personal computer system with a high-voltage electrical device or a direct current/alternating current motor. We will be able to create the graphical user interface that will monitor and control the stepper motor's speed using the technology that has already been built. The hardware and the software, the project's two component parts, can be separated into their own parts. The suggested system's hardware and its connection to a computer through an RS232 serial port are both described. To establish communication, we are setting up an RS-232 connection between the PC and the controller.

Keywords: High Stepper Motor, R$232, PC, MAX 232

Introduction

The stepper motor was created to address a particular gap in the motor control literature. The most common uses for these motors are in measuring and control applications. Some examples of potential applications include volumetric pumps, CNC machines, inkjet printers. In general, stepper motors share a number of traits that make them especially well-suited for the tasks listed below. Stepper motors employ brushless motors. Traditional motor parts with the highest failure rates are the commutator and brushes. They also produce electrical arcs, which in some circumstances can be undesirable or even dangerous. The stepper motor won't revolve at a speed no matter how much resistance it is receiving from the load as long as the load does not exceed the torque rating of the motor. When stepper motors are in operation, their open-loop positions move in predetermined steps. The shaft can be kept in its current position thanks to the holding torque property. From the motors used in floppy discs to those used in large machinery, stepper motors are available in a wide range of sizes and powers. Utilising the microcontroller's Pulse-Width-Modulation (PWM) feature, it is possible to regulate the duty cycle of the stepper motor driving. PWM is a completely different approach than what is often used to control the speed of a stepper motor. The electrical power supplied to the motor is a square wave with changing pulse width or duty cycle but constant voltage. The percentage of a single cycle that happens within the duty cycle of a constantly occurring train of pulses is referred to as the "duty cycle" in this context. Since the power is off for the most of the time when the duty cycle is low, a low duty cycle translates to a low amount of electricity. The term "duty cycle" describes the ratio of "on" time to the typical interval or "period of time." The duty cycle is expressed as a percentage, with 100 percent denoting full operation of the instrument.[1,2]

Hardware Development

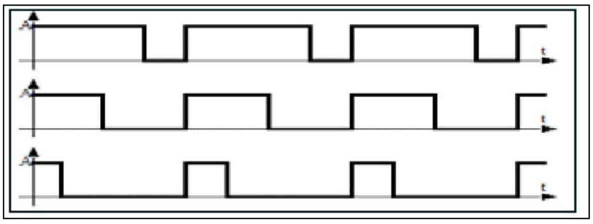

Figure 1.Wave form at Different Duty Cycles

The following illustrates, in block diagram form, the proposed system

The Power Supply Modules

The main function of this module is to supply 5 volts at a current of 500 milliamps. This is composed of a transformer to step down the AC voltage, IN4007 diodes to form a bridge rectifier to convert AC to DC, a 1000uF capacitor to serve as a filter circuit, 7805 regulators to produce 5V at the regulator's output, 330-ohm resistance, an LED to serve as an indicator.

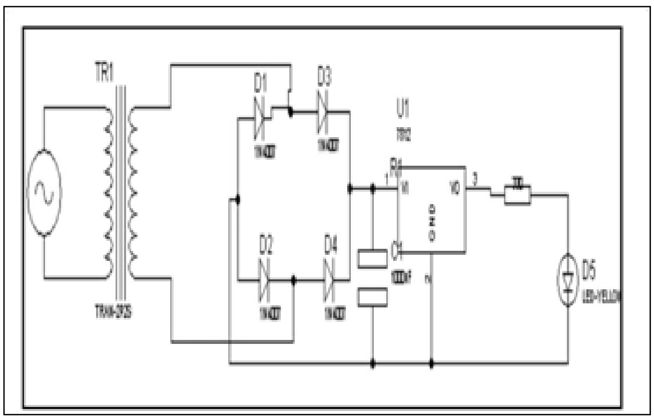

Figure 2.Diagram of Supply Section

Microcontroller (ATmega16)

Microcontrollers can be found almost anywhere. The ATMEGA16 from the AVR series was selected for this project due to its built-in ADC interface and configurable frequency. AVR RISC architecture is used by the low-power CMOS 8-bit microcontroller AT megal6. To achieve high throughput, the ATmegal6 executes potent instructions in a single clock cycle. This enables the system to maximise both processing speed and power consumption. It also lowers LAN expenses.[3]

Model C USB-to-serial cable

This node interfaces with the coordination node and other nodes.

(level converter IC)

The MAX232 dual driver/receiver IC produces EIA-232 voltage levels from a single source via a capacitive voltage generator.

5 V. Each receiver converts EIA-232 to 5 V TTL or CMOS. Receivers have a 1.3 V threshold. 0.5 V hysteresis, 30 V inputs. Each driver converts TTL/CMOS to EIA-232. We should be able to get this functioning with a few capacitors.

Liquid Crystal Display

The LCD (liquid crystal display) device receives 8-bit character codes from a microprocessor or microcomputer and latches them to its display data RAM (80-byte) DD RAM for storing 80 characters. The LCD (liquid crystal display) device then displays the characters on its LCD screen. LCD (liquid crystal display) We have 16 hardware pins and a 16x2 LCD. Pins 1, 3, 16 are grounded, whereas pins 2 and 15 are powered by 5 volts. Pins 3, 4, enable are labelled as RS, RW, enable, respectively. Display transmits four bits in parallel using the LCD data ports 11, 12, 13, 14.[14]

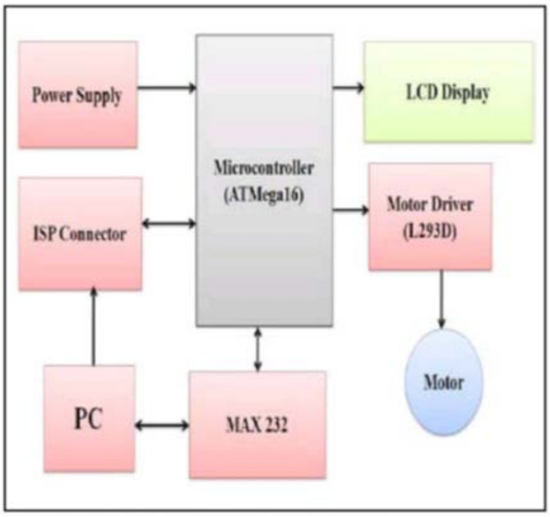

Figure 3.Block Diagram of System

F. DB9

The connector has 9 male and 9 female pins. In DB9, the number 9 refers to the total number of pins, the letter D refers to the two rows of pins that are parallel to one another and are shaped like the alphabet.

G. L293D (Motor Driver IC)

This integrated circuit is a high-voltage, high-current, four-channel driver that is designed to take DTL or TTL logic. This has the capability of delivering an output current of 600 mA per channel, as well as providing 1.2 peak output current (non repetitively) per channel, it also has built-in safety against internal overheating. It is made up of a Half H Bridge, which allows for a large current to be supplied in order to drive motors.[5]

H.RS-232

An RS-232 port was originally a common component of a personal computer, it served as a connection point for various peripheral devices such as modems, printers, mouse, data storage devices. The PC and the controller will communicate with one another using RS-232 cables.[6] In a high-level language (like C or java), programming and debugging are effectively the same thing. A high-level language's compiler can help reduce the duration of the production process. When writing programmes for the microcontrollers using WinAVR,[2] C was the preferred programming language. To make future maintenance and improvements more straightforward, the source code has been annotated. The Windows operating system is the host for the executable open source software development tools known as WinAVR. These tools are made to work with the RISC microprocessors in the Atmel AVR series. The GNU GCC compiler, which works with both C and C++, is included. WinAVR includes each and every tool required for AVR programming. This includes, among other things, the AVR-gce compiler and the AVR-gdb debugger.

Compiling

The code is then converted into a machine-level language composed of 0s and 1s when the programme has been compiled. The Hex file (Hex) is the name given to this file, which is saved with the Hex extension. Additionally, the compiler will result in errors in the software that must be corrected in order for it to execute successfully.[6]

Burning

It is possible to directly write the machine language (hex) file into the microcontroller's programme memory using a particular programmer that links to a PC's peripheral. This has been accomplished by using the computer's serial port. The machine language file was written into the programme memory of the microcontroller using the Ponyprog programmer in order to achieve this purpose. Pony Prog is a serial device programming tool that works with Intel Linux and Windows 95, 98, ME, NT, 2000, XP. It has a simple to use visual user interface. Data from and to all serial devices can be read and written using this device. Along with the I'C Bus and Micro wire communication protocols, it is compatible with the Atmel AVR and Microchip PIC microcontrollers. The microcontrollers were each programmed in less than two seconds using a high speed programming option. The programme memory is a Flash kind, it has a limited amount of time for use, just like an EEPROM. The programming can be reprogrammed an infinite number of times on the AVR microcontroller family without the risk of causing data loss using the Atmega16 Programmer (ISP), which is used to burn the programme into AVR microcontrollers. You can repeat this as many times as you like.

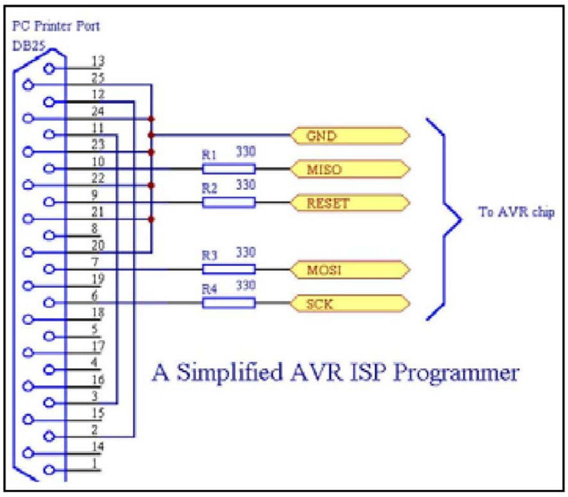

Fig. 5: Snapshot of Robokits AVR Programmer Window to Erase, Write and Verify .h File Generated by AVRstudio4 to Target

Figure 4

Assessment

The software development process is complete and the project is ready to be installed as a LAN if the system performs as required by the user and all tasks efficiently and effectively. The procedure is repeated if mistakes are made. Resources on microcontrollers are limited, which makes programming them challenging. PCs have an unlimited amount of RAM and computational power, unlike microcontrollers. Microcontroller code should be as cost- and power-efficient as is practical while still being resource-efficient. The software of the system. uses the.h and.c files.

ICD.C

The LCD module is managed by this c code. The LCD's initialization, data writing, pointer movement, characteristics, location are all controlled by code. It supports LCD data entering done character-by-character or string-by-string. The programme is built around the Hitachi HD44780 IC command set found in the LCD. It contains the functions initlcd(), delete(), show(), displayint.

Programmable operations

The controller's Source Code Contains

I programme delay

Void delay (unsigned char value) _delay ms(1);

Motor control

Motor dead (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;

Break; motor at 0%

}

Conclusion

Stepper motors are utilised in industrial, office, medical, automotive applications. This study successfully designed a system to control the speed of a Stepper Motor through a PC due to its simplicity, low cost, high dependability, ability to be controlled in an open-loop system. The study advances industrial stepper motor utilisation.

References

1. Alae AR, Negm MM, Kassas M. A PLC Based Power Factor Controller for a 3 Phase Induction Motor", IEEE Transactions on Energy Conversion, USA, 1065-1071,2000.
2. loannidis MG, Design and Implementation of PLC-Based Monitoring Control System for Induction Motor, IEEE Transactions on Energy C onversion, 19, No. 3, USA,2004
3. M.G. Loannidis, "Design and Implementation of PLC-Based Monitoring Control System for Induction Motor", IEEE
4. Transactions on Energy Conversion, 19, No. 3, USA, 2004
5. S. M. Bashi, I. Aris, S.H. Hamad Development of Single Phase Induction Motor Adjustable Speed Control Using M68HC1IE-9 Microcontroller", Journal of Applied Sciences S(2), pp. 249-252
6. Kumara MKSC, Dayananda PRD, Gunatillaka MDPR, Jayawickrama SS,"PC based speed controlling of a dc motor", A final year report University of Moratuwa llinious USA, 2001102
7. J.Chiasson,"Nonlinear Differential Geometric Techniques for Control of a Series DC Motor", IEEE Transactionson
8. Control Systems Technology, Vol. 2. pp. 35-42, 1994
9. Yodyium Tipsuwan, Mo-Yuen Chow, "FuzzylLogie microcontroller implementation for DC motor speed control", IEEE Transactions on Power Electronics 1999; 11(3): 1271-1276.