2D-Linear Motion Controller for Step Motors Based on a 8-bit microcontroler

Dejan Mitic[1], Dragan Antic[2], Zoran Jovanovic[3]

***Abstract* – In this paper a solution of motion controller for turning machine with stepper motors is shown. Motion controller is based on 8-bit microcontroller that receives commands from the supervisor computer dedicated to perform CNC program and communicate with operator. Special set of commands has been developed to functionally cover all specific needs for turning machine motions.**

***Keywords* – motion controller, CNC unit, step motor, turning machines**

I. INTRODUCTION

One of the most important machine industry improvements, almost comparable with new technology revolution, is certainly automation of production machines by supplementing them with the computer numerical control unit. CNC machines become, de facto, standard equipment of modern production lines. They are programmed by specially designed and standardized control language that has many 'dialects' depend on type of machine. Interface between computer and machine play a vital role in a control loop. Sensors, like position decoders, indexer, boundary switches, etc. send information about position, temperature and other statuses to the control unit. Accordingly to the control program, control unit acts via a range of actuators as motors, valves, pumps, heaters etc. The most important activity of any machines is motion. Depend on type of motor and motion complexity many type of motion controller has been developed. In this paper we shall present a solution for small and precise turning machine equipped with stepper motors and indexer.

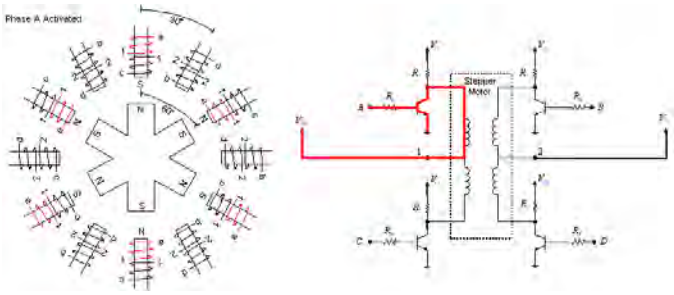

Fig. 1. Stepper motor phases activation

[1]Dejan Mitic is with Tehnopetrol, Sretena Mladenovića 32/10, 18000 Niš, Serbia and Montenegro, E-mail: dmitic@elfak.ni.ac.yu
[2]Prof. dr Drangan Antic is with the Faculty of Electronic Engineering, Beogradska 14, 18000 Nis, Serbia and Montenegro, E-mail: dantic@elfak.ni.ac.yu
[3]Mr Zoran Jovanovic is with the Faculty of Electronic Engineering, Beogradska 14, 18000 Nis, Serbia and Montenegro, E-mail: zoki@elfak.ni.ac.yu

II. DESCRIPTION OF MOTION CONTROLLER HARDWARE

Motion controller is applied for bipolar stepper motor (Fig. 1) control without closed loop.

Solution is based on microcontroller that is networked with other control modules of CNC unit. For that purpose we have chosen 8-bit AT89C2051 microcontroller and dedicate its ports to I/O signals as it is shown on Table I.

TABLE I
I/O SIGNALS ON AT89C2051 MICROCONTROLLER

N.P	Pin	Port	Sig.	N.P	Pin	Port	Sig.
1	RST		SRD	20	VCC		
2	RxD	P3.0	SWR	19		P1.7	DIRX
3	TxD	P3.1		18		P1.6	CLKX
4	XTAL2			17		P1.5	DIRZ
5	XTAL1			16		P1.4	CLKZ
6	!INT0	P3.2	BSY	15		P1.3	NLSX
7	!INT1	P3.3	SIND	14		P1.2	PLSX
8	T0	P3.4	ICLK	13	AIN1	P1.1	PLSZ
9	T1	P3.5		12	AIN0	P1.0	PPSX
10	GND			11		P3.7	PPSZ

Interface signals (Table II) provide connection of motion controller with relevant sensors and actuators.

TABLE II
DESCRIPTION OF THE INTERFACE SIGNALS

Sig.	Description
SRD	Serial read
SWR	Serial write
BSY	Busy
SIND	Indexer start
ICLK	Indexer clock
DIRX	X-axis direction
CLKX	X-axis clock
DIRZ	Z-axis direction
CLKZ	Z-axis clock
NLSX	X-axis negative limit switch
PLSX	X-axis positive limit switch
PLSZ	Z-axis positive limit switch
PPSX	X-axis precise position switch
PPSZ	Z-axis precise position switch

III. Description of Motion Controller Software

We developed software solution that completely meets all specific requirements that motion control for turning machines has.

CONTROL COMMANDS

A set of commands (Table III) has been developed to control the motion controller:

III Table
Control comands

Mnemonic	Description
INIT	Initialization
FMOV	Fast federate move
ZMOV	Move to machine zero
PMOV	Programmed feedrate move
STOP	Stop motion
FFDR	Set fast motion federate
MFDR	Set machining federate
PFDR	Set program federate
DELY	Delay
CSTP	Clock based motion step
ISTP	Indexer based motion step
PSTN	Set position
RDPS	Read position
RDST	Read status

INIT

Initialize the motion controller with default parameters for position and feedrates, clear all queues and put the motion controller in ready state.

FMOV

Fast move to specified position. This command is issued for fast positioning without machining. Usually, it is direct translation of G00 or some complex motion CNC commands.

ZMOV

Fast move to machine zero position defined with limits switches. Usually, command is issued for positioning machine for tool change or final machining position setup.

PMOV

Programmed move to specified position. Command is issued for motion during machining. Usually, it is direct translation of G01 or any other machining motion.

STOP

Abrupt stop movement. Command is issued for abrupt stop commanded by operator.

FFDR

Set fast feedrate used during FMOV and ZMOV command. Command is issued for immediate altering fast feedrate value. It allows to operator continual changing of maximal traverse speed during fast movement.

MFDR

Machining feedrate used during PMOV Command is issued for immediate altering normal feedrate value. It allows to operator continual changing machining speed.

PFDR

Machining feedrate used during PMOV. Command is issued for changing feedrate value upon fetching next motion command.

DELY

Delay machining for specified amount of time. Command is issued usually due to G04 command or during complex movement when there is need for short pause between two consequent movements.

CSTP

Set clock timer as source for stepping. Dedicated to simple motion control. Feedrate depends only on the timer clock. Speed in CNC program is expressed in mm/min.

ISTP

Set indexer timer as source for stepping. Dedicated to thread machining motion control. Feederate depends on speed of the main spindle. It allows special machine motion typical for turning machines, such as motion for producing threads.

PSTN

Set actual position. Command is issued to setup new coordinate. The actual position takes provided arguments values.

RDPS

Read actual position. Command is a request for actual position. Motion controller returns value of the actual position to computer. Usually, such information is presented to operator via appropriate user interface.

RDST

Read motion control status flags. Command is a request for status flags of motion controller. According to these flags computer as well as operator is informed about motion status, potential reaching limit positions or unpredictable situations.

SOFTWARE MODULES

THE PATH PLANNER MODULE covers all motion that turning machine should perform. It is linear motion planner for 24-bit discrete 2D motion space with two timing sources. The clock timing step source allows simple machining motion. The indexer timing step source is using for thread machining. The Bresenham's line algorithm has been used for discretization of motion space. The same algorithm has been used for discretization of the synchronized spindle-tool motion space in case of the indexer-controlled motion. Algorithm allows minimal inaccuracy on motion path caused by discretization. For reason of processing speed algorithm

has been split on two phases: motion preparing and motion executing. First phase is executed immediately upon interpreting command as a motion command. On the end of first phase all calculated data for phase two are saved as parameters of the new pending movement. Phase two is executed in every step cycle while the actual position does not reach final position.

THE PROCESSING MODULE is virtual processor, which fetches calculated data from the first phase of Bresenham's line algorithm and executes motion by following the second phase of the mentioned algorithm. Second phase is executed in determined moments according to motion feedrate and type of timer source (clock or indexer).

```
processing:
if (prosessor.isready()) {
      cmd = fetch();
      start(cmd);
}
if                (!processor.isstopped())
processor.run();
goto processing;
```

THE MODULE FOR ASYNCHRONOUS SERIAL COMMUNICATION runs input and output communication queues with error-free communication protocol. This protocol provide link between computer and motion controller as well as other microcontroller sustems dedicated to other tasks of control.

```
async_comm:
if (mssg_received()) {
      if (validate()) {
            cmd = extract_cmd();
            cmdq.put(cmd);
      } else {
            send(NAK);
      }
}
if (mssg_ready() == true) {
      send(mssg);
      while (ackn() == NAK) resend(msg);
}
goto async_comm;
```

THE INTERPRETER MODULE is supplied from pipelined command queue, which could accept three commands in advance, feeding interpreter continuously. It allows smooth motion changes to succeeding path. Also interpreter is able to find in advance commands that have to be executed immediately upon receiving (STOP, RDPS, RSST, etc).

```
interpret:
if (cmdq.isempty()) goto interpret;
while (cmdq.isimmcmd()) {
      icmd = cmdq.getimmcmd();
      processor.execute(icmd);
}
if (processor.ispend()) goto interpret;
processor.put(cmdq.getcmd());
goto interpret;
```

IV Conclusion

In this paper we shown practical realization of 2D-linear motion controller for stepper motor based on 8-bit microcontroller AT89C2051 and supplemented with unique software solution based on set of commands that cover all specific requirements of turning machine motion. Achieved results met all expectations. Implementation of preprocessing and preparing motion command in advance gave smooth transition between adjacent trajectories and resulting in high quality machining.

References

[1] Douglas W.J., *Control of Stepping Motors*, 5.2.10, 10.8, 10.9 and 10.10 of the Handbook of Small Electric Motors edited by W. H. Yeadon and A. W. Yeadon, McGraw-Hill, 2001.

[2] Koren, Y., *Control of Machine Tools and Robots*, Applied Mechanics Reviews, Vol. 39, No. 9,Sept. 1986, pp. 1331-1338.

[3] Kramer R.T., Proctor M.F., Messina E., *The NIST RS274NGC Interpreter*, Intelligent Systems Division National Institute of Standards and Technology, Technology Administration U.S. Department of Commerce, Gaithersburg, Maryland 20899, NISTIR 6556, August 17, 2000