17th International Mining Congress and Exhibition of Turkey- IMCET2001, ©2001, ISBN 975-395-417-4

Simulation of Optimal Control for Mining Drilling Robot

E.Pop, M.Pop & M.Leba
Department of Informatics and System Control, University of Petrosani, Romania

ABSTRACT: In this paper, a system consisting of a mining drilling robot, microcontroller drives, and load model are considered. The drilling machine was modified to a three grades of mobility robot with rotation joints. Each joint is driven by a DC motor with power electronics. Control software is implemented in two parts of the system, one at high level in PC memory and the other at intermediary level in the microcontroller ROM memory as local BIOS. The mathematical model is determined as twelve transcendental equations written by the Denavit-Hartenberg methods. Inverting the equations with the software, three components of the position vector are determined. Thus, for each given position, it is possible to find the coordinates of the robot's elements. The gripper of me robot represents the electromechanical drilling machine, driven by an AC motor. A package of programs written in assembly language is implemented in a PC computer at high level İn order to optimize the drilling process, and implemented in the microcontrollers at intermediary level. The entire system was real-time simulated using a dSPACE kit.

1 INTRODUCTION

In this paper, a three grades of mobility robot with rotation joints is considered. These three joints are used for positioning the drilling machine.

This robot has a drilling machine on the last element and can be used for excavation in mining construction for increased safety of personnel (Fig.l).

To control the robot's joints, we use a SIEMENS CI67 16-bit microcontroller as a dedicated PLC to generate the movement trajectories according to optimal control algorithms (Anon., 1997).

To work in real-time we propose a distributed structure: a PC connected to a 16-bit microcontroller and power electronics at local level.

The software consists of two parts linked together with bi-directional data flow.

The first part solves the mathematical model to determine the rotation angles and the cutting speed (<a) of the drilling machine.

The second part determines the new position of the joints and the rotation direction, converts the angles into pulses and sends diese pulses to power electronics.

The entire process was simulated using a cinematic model according to the Denavit-Hartenberg method.

A program was written in assembly language so as to allow real-time working.

The programs run in real time, and ensure a high level of flexibility in the robot.

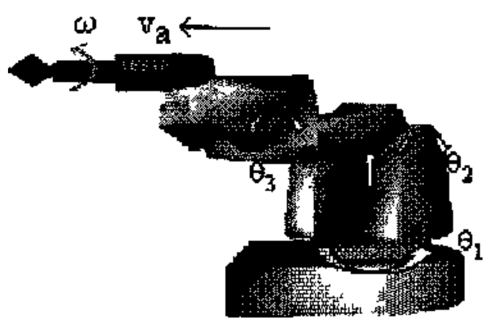

Figure 1. Drilling robot

In order to control the joints, the cinematic inverse problem was solved and then real-time simulated using dSPACE equipment with the expected results.

2 HARDWARE

The hardware consists of a PC, a SIEMENS CI67 16-bit microcontroller to control the joints, and a drilling machine operated by means of four power electronics modules (PE,) (Fig.2).

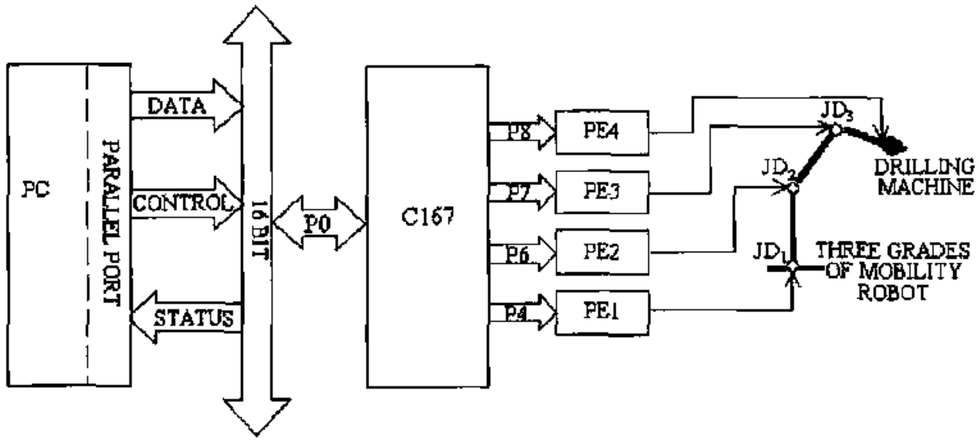

Figure 2. Hardware block diagram

The protocol data exchange is based on the handshaking principle. The process begins with the PC sending pooling data to identify each joint position. The parallel port of the PC, its 8-bit data section and two control bits and two status bits from the other two sections are used for this task.

These 12 bits represent the information for the 16-bit C167 microcontrolJer. The C167 decodes the information sent by the PC and determines the direction and angle value for each joint movement. This information is sent by means of four 8-bit output ports (P4, P6, P7, P8).

The signals generated by PE4 are to control the cutting speed of the drilling machine in order to optimize its efficiency.

This information is the input for the power electronics. The power electronics convert the received data into electrical pulses for the joint and drilling machine drives (JD,).

3 MATHEMATICAL MODEL

The mathematical model of the three grades of mobility robot is determined by the following conditions: the first group of three rotation joints is for the positioning mechanism, and to the last element we fixed a drilling machine.

We consider that the rotation angles are 9, (i= 1,3) and the dimensions of the positioning elements are d, (1=1,2,3). Then, the movement matrix $T_{0,3}$ is as follows:

$$T_{0,3} = \begin{bmatrix} n_{11} & o_{11} & a_{11} & d_{11} \\ n_{21} & o_{21} & a_{21} & d_{21} \\ n_{31} & o_{31} & a_{31} & d_{31} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Using the Denavit-Hartenberg method, the 12 movement equations are:

$$\cos\theta_1 \cdot \cos(\theta_2+\theta_3) = n_{11}$$

$$-\cos\theta_1 \cdot \sin(\theta_2+\theta_3) = o_{11}$$

$$-\sin\theta_1 = a_{11}$$

$$\cos\theta_1 \cdot \cos(\theta_2+\theta_3) \cdot d_3 + \cos\theta_1 \cdot \cos\theta_2 \cdot d_2 = d_{11}$$

$$\sin\theta_1 \cdot \cos(\theta_2+\theta_3) = n_{21}$$

$$-\sin\theta_1 \cdot \sin(\theta_2+\theta_3) = o_{21}$$

$$\cos\theta_1 = a_{21}$$

$$\sin\theta_1 \cdot \cos(\theta_2+\theta_3) \cdot d_3 + \sin\theta_1 \cdot \cos\theta_2 \cdot d_2 = d_{21}$$

$$-\sin(\theta_2+\theta_3) = n_{31}$$

$$-\cos(\theta_2+\theta_3) = o_{31}$$

$$0 = a_{31}$$

$$d_1 - \sin(\theta_2+\theta_3) \cdot d_3 - \sin\theta_2 \cdot d_2 = d_{31}$$

Inverting these transcendental equations, we obtain the angles 6,:

$$tg\theta_1 = \frac{d_{21}}{d_{11}}$$

$$\cos\varphi = \frac{d_{11}^2 + d_{21}^2 + (d_1 - d_{31})^2 + d_2^2 - d_3^2}{2 \cdot d_2 \cdot \sqrt{d_{11}^2 + d_{21}^2 + (d_1 - d_{31})^2}}$$

$$\cos(\theta_2 + \varphi) = \sqrt{\frac{d_{11}^2 + d_{21}^2}{d_{11}^2 + d_{21}^2 + (d_1 - d_{31})^2}}$$

$$\cos\theta_3 = \frac{d_{11}^2 + d_{21}^2 + (d_1 - d_{31})^2 - d_2^2 - d_3^2}{2 \cdot d_2 \cdot d_3}$$

To determine the optimum cutting speed, we use the equation:

$$\omega = k \cdot \frac{v_a}{s},$$

where k is constant, v_a is advance speed and s is the splinter width.

These equations represent the mathematical inverse model for determining the rotation angles and the drilling machine speed.

4 CONTROL ALGORITHM

For real-time simulation and digital control of the robot, the algorithm consists of several steps (Fig.3).

Given the input data, coordinates x, y, z and parameters of the drilling machine k, v_a s, we determine the movement matrix TQ,3, then calculate the angles 0, (i=l, ..., 3) and cutting speed co. After this, we calibrate the six movement equations and determine the movement angles for each joint. All these steps are executed by the PC, which displays the robot position on the screen.

The next steps will be executed by the CI67 microcontroller as follows: taking the data of specific angles from the PC, this will be saved in the corresponding microcontroller RAM memory. Then the rotation direction will be found, the timing pulses corresponding to the specific equation of angles, and the pulses will be sent to the power electronics. In addition, the microcontroller sends feedback to the PC related to the position and other parameters of movement.

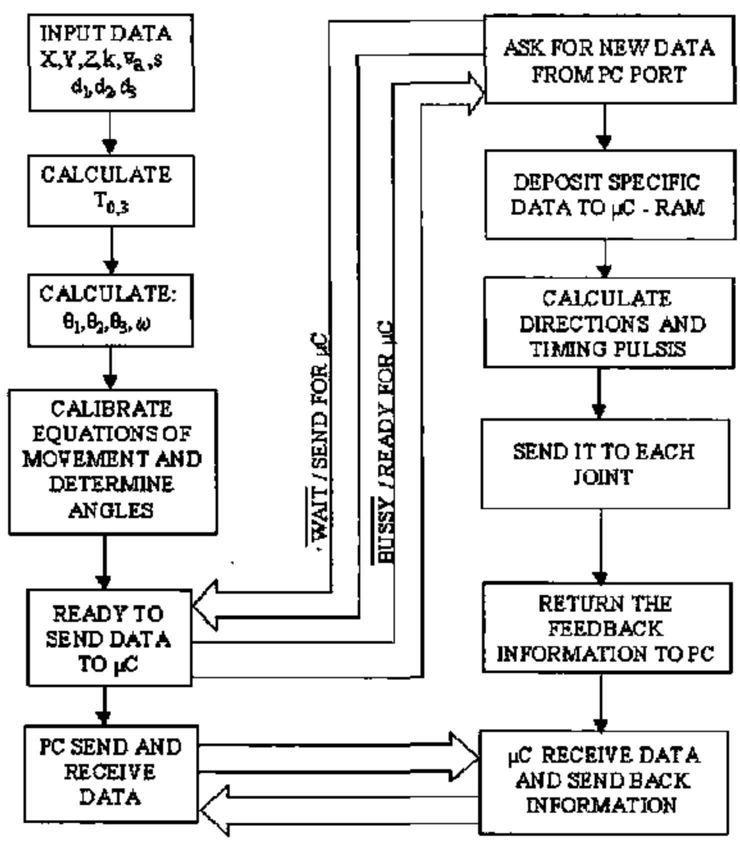

Figure 3. Control algorithm diagram

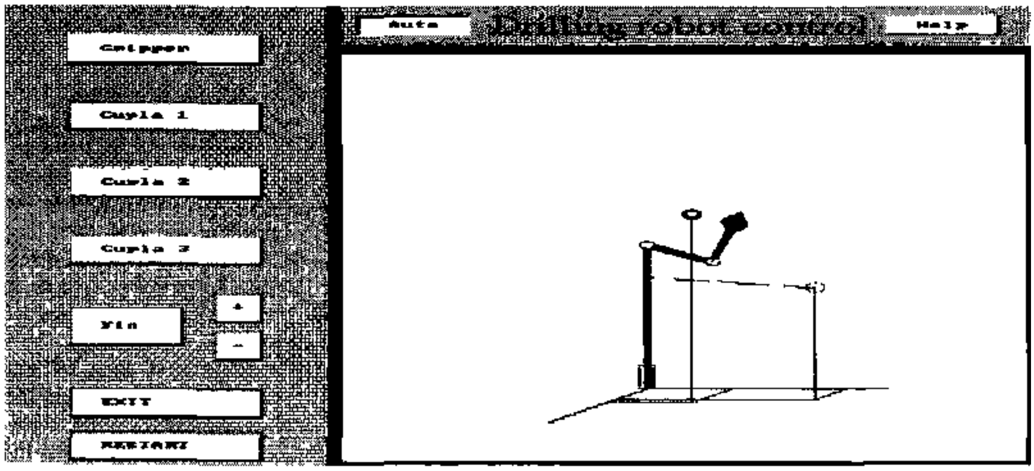

Figure 4 Assembly language program display.

5 SOFTWARE CONTROL

The software consists of two parts: at the PC level, there is a program written in assembly language for solving the inverse model and for graphic simulation. At the microcontroller level, there is a program written in C167 assembly language.

The PC software determines the movement angles, which are transmitted to the next level. It also displays a windows graphical simulation and control of the robot (Fig.4).

The microcontroller software generates the angle for each joint using its ports, compares it with the last position, and establishes the direction of movement according to the new position. Then, the microcontroller outputs the string of pulses to the power electronics.

The PC software written in 1-80X86 assembly language consists of over 2000 statements, of which the main program is;

```
start' call getcar
   mov ax,0a000h
   mov es,ax; initialize es=video address
   mov ax,@data
   mov ds,ax
   mov ah,0; puts VIDEO mode
   mov al,12h ; MODE=12h, 640x480
   int 10h
   call EnableMouse
redo: call HideMousePointer
   call desenlnit
   call ShowMousePointer
   callinit
   call InversSursa ; calculates the inverse matrix
   call scrieVal
   call InversDestinatia
   call desen
   call buton      ; polls the buttons
   call TestButtonPress
   cmp restart, 1
   je redo; reinitializes the robot
   mov ah,0; wait for a key to be struck
   int 16h
   mov al,3; clear display
   mov ah,0
   mov bh,0
   int 10h
   mov ax,4c00h   ; exit to O.S.
   int21h
end start
```

6 CONCLUSIONS

1. A robust three grades of mobility robot together with a drilling machine can become a drilling robot if it is software controlled.

2. In terms of hardware structure, the robot has a PC, a 16-bit microcontroller and corresponding power electronics for each joint.

3. The sophisticated software written in assembly language for the PC and microcontroller ensures a programmed drilling job, optimal control, high flexibility and user-friendly graphical support.

REFERENCES

Pop, E., Leba, M. & Egn, A. 2001 Robotic control systems *Ed Did & Pedagogica*

Pop, E. & Pop, M. 1999. Assembly language programming for [-80x86 microprocessors. *Ed.Did & Pedagogica*

Anon, 1997 CI 67 *SIEMENS Data Book*