ET 472 – Integrated Control Systems

Lab 3
PLC Control of a Stepper Motor

Due Dates: The results of this lab will be combined with those of Lab 4 (upcoming) into a single lab report. You will build upon the control program created for this lab in Lab 4. The lab report will be due one week after Lab 4 has been performed.

Purpose: The purpose of this laboratory exercise is to control the direction, speed and number of rotations of a stepper motor using a PLC. Data bytes will be used to define the sequence of control outputs necessary to drive the motor through each step of its rotation. The speed of the motor will be controlled using a timer for pacing the output of each byte pattern. Counters will be used to keep track of the number of steps taken as the motor moves in both the forward and reverse directions.

Objectives: After completing this laboratory exercise, the student should be able to write a PLC program which controls a stepper motor to execute rotations of specific speed, direction and quantity.

Key Terms:
Stepper Motor
Stepper Motor Resolution
Move Instruction
QB0
Data Block
Compare Instruction

System Requirements:

The control program should drive a stepper motor which turns a lead screw. As the screw is rotated by the motor, a pointer on the screw travels a linear distance along a ruler. The ruler is scaled to show the distance that the pointer has traveled in millimeters.

The lead screw has a pitch of 20 threads per inch. The motor should rotate the lead screw so that the pointer moves precisely 20 millimeters along the ruler away from its home position and then reverses directions to return back to its home position. The entire procedure should be designed to take approximately 1 minute, 30 seconds.

During the return motion of the point, a strobe light should be activated.

Procedure:

1. Define four data bytes in variable (V) memory using the Data Block. These correspond to the required four step sequence bit patterns (see the Supplemental Information later in this handout)

2. Use the byte 0 of output memory (QB0) as the output to control the motor.

3. Program an on-delay timer (TON) to create a one shot "clock pulse" with a time interval of 20 ms.

4. Use a counter to count through the different output control byte patterns.

5. Use a second counter to count total number of steps taken by the motor.

6. Use the Move Byte (MOV_B) instruction to move the required data byte from variable memory to QB0.

7. Devise a control program which generates a sequence of output patterns to drive the motor so that the pointer moves 20 mm to the right of its home position before reversing and returning the same distance back to stop at the home position. The entire procedure should take approximately 1 minute, 30 seconds.

8. During the return phase of motion, a strobe light should be activated. Use Q0.4 as the output to enable the strobe light.

9. Compare the finish point to the original starting point on the motor shaft and determine if any error has occurred (e.g. too few or to many steps have been generated).

10. Modify your program to compensate for any error (if required) and be prepared to explain why any modification was necessary.

11. Submit a written laboratory report following the specified format.

SUPPLEMENTAL INFORMATION:

Table 1 below shows the order of the four control patterns which cause the motor to rotation in a clockwise direction as seen from the shaft side of the motor:

Step 1 → Step 2 → Step 3 → Step 4

To reverse the direction of rotation, the step sequence would be:

Step 4 → Step 3 → Step 2 → Step 1

Note that coils opposite one another in the motor are never ON simultaneously. Coil 1 is opposite Coil 2 and Coil 3 is opposite Coil 4. Hence, adjacent switches are actuated, two at a time, in a rotating pattern.

Table 1: Stepper Motor Control Input Sequence

Step	**Coil 1**	**Coil 2**	**Coil 3**	**Coil 4**
1	ON	OFF	ON	OFF
2	ON	OFF	OFF	ON
3	OFF	ON	OFF	ON
4	OFF	ON	ON	OFF

Move Function:

The figure below shows an example of using a compare instruction to move data from a byte in variable memory to a byte in output memory. In network 2, the contents of VB0 will be moved to output byte QB0 when counter C0 is equal to 3. In the subsequent network, the contents of VB1 will be moved to QB0 when C0 is equal to 2.

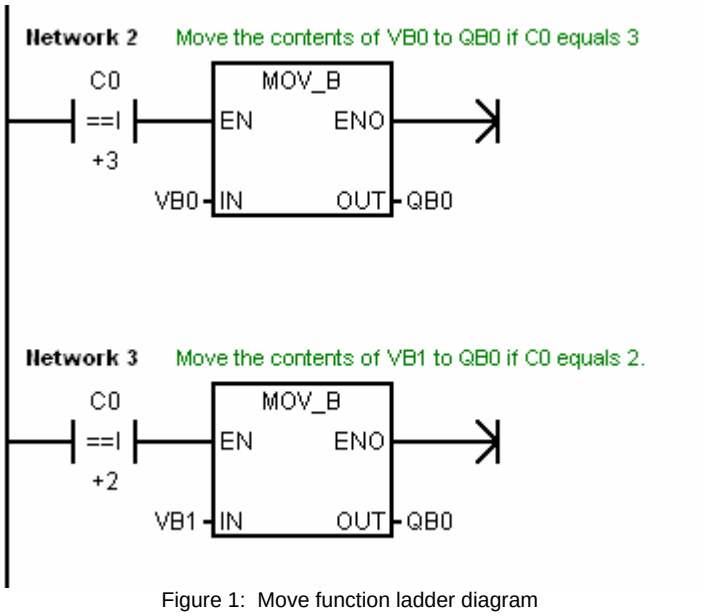

Figure 1: Move function ladder diagram

Data Block:

A data block allows you to store values in variable memory. In this lab, a data block will be used to store the contents of bit patterns required to control the stepper motor. You will need an entry in the data block to store each bit pattern in the stepper motor's control sequence. These bit patterns are shown below.

```
//
// DATA BLOCK TITLE COMMENTS
//
// Press F1 for help and example data block
//
VB10 10   // DATA FOR 1ST STEP SEQUENCE (BINARY 1010)
VB20 9    // DATA FOR 2ND STEP SEQUENCE (BINARY 1001)
VB30 5    // DATA FOR 3RD STEP SEQUENCE (BINARY 0101)
VB40 6    // DATA FOR 4TH STEP SEQUENCE (BINARY 0110)
```