```
.MODEL SMALL          // model for execution
.STACK 100H           // declaring stack
.DATA

   CR EQU 0C263H      // address of control register
   PA EQU 0C260H      // address of port A
   PB EQU 0C261H      // address of port B
   PC EQU 0C262H      // address of port C

   Message1 DB 'DEMONSTRATION PROGRAM FOR
              STEPPER MOTOR', 13, 10, '$'
   Message2 DB 13,10, 'The program is running...',
              13, 10, '$'
      // declaring two messages

.CODE
   start:
   MOV AX, @DATA          // moving data from DATA
   MOV DS, AX                to DS
   MOV AH, 9H             // display Message 1
   MOV DX, OFFSET Message1
    INT 21H
                          // display Message 2
   MOV AH, 9H
   MOV DX, OFFSET Message2
    INT 21H
    MOV DX, CR
    MOV AL, 80H           // initialize 8255
    OUT DX, AL            // motor in I/O mode of operation,
                             all ports act as output
    MOV BL, 50            // counter
```

```
BEGIN: MOV AL, 11H            // excite A winding
 CALL OUT_A
 CALL DELAY
 MOV AL, 22H                  // excite B winding
 CALL OUT_A
 CALL DELAY
 MOV AL, 44H                  // excite C winding
 CALL OUT_A
 CALL DELAY
 MOV AL, 88H                  // excite D winding
 CALL OUT_A
 CALL DELAY
 DEC BL
 JNZ BEGIN                    // loop for 50 times
 MOV AH, 4CH                  // terminate
 INT 21H
 OUT_A: MOV DX, PA            // copy address of port A
  OUT DX, AL                  // write data to port A
  RET
 DELAY: MOV CX, 0FFFH               // add delay
 D2: MOV AX, 05FFH
 D1: DEC AX
  JNZ D1
  DEC CX
  JNZ D2
  RET
 END START
```