**Experiment/ Task:** Stepper Motor Interfacing with 8086 Microprocessor.

1. Clockwise & Anticlockwise Rotation.
2. Speed Change.
3. Show Speed into 7-segment Display.

**Objective:**

* Interfacing a stepper motor with 8086 Microprocessor.
* Change motor Speed.
* Show speed into a 7-segment display.

**Software Use:**

* Proteus 8 Professional V8.13

**Proteus Component List**

* 74HC373 Letch
* 8086 Microprocessor
* 8255A PPI
* CELL -12V
* Logic-state
* ULN2003A Motor Driver
* Stepper Motor
* White-LED
* 7-Segment Display

**Block Diagram:**

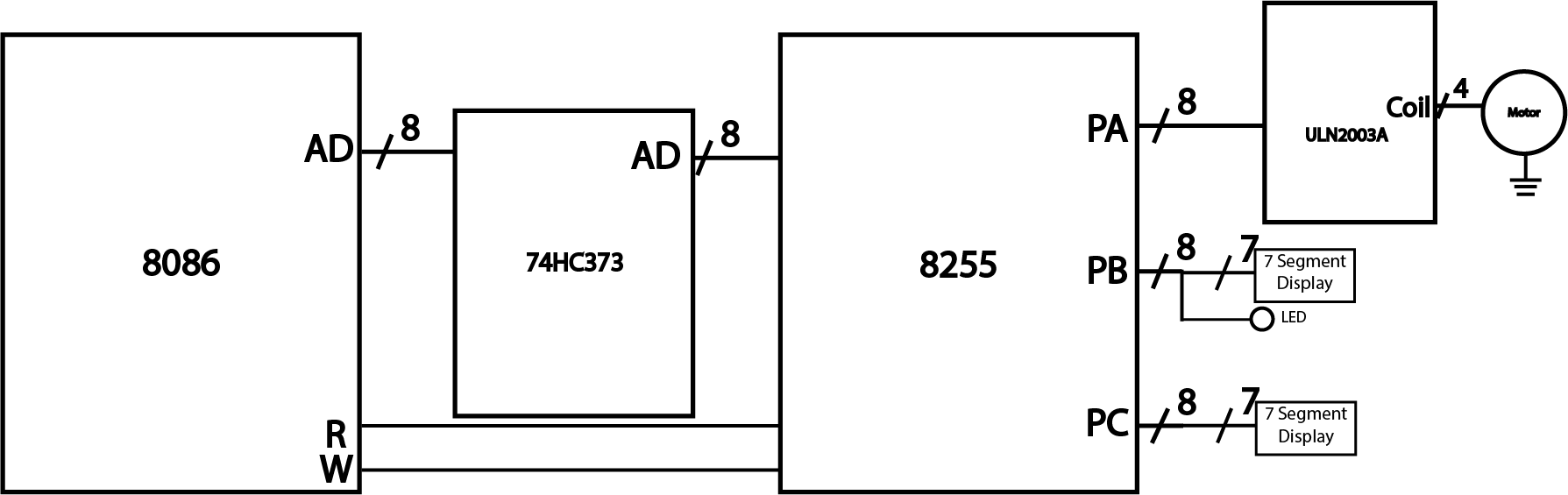
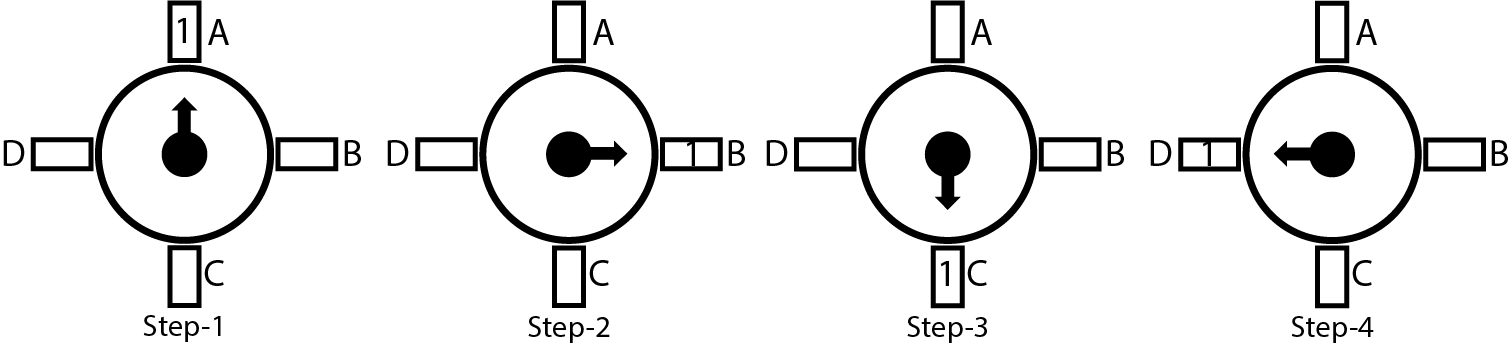


Fig: Block Diagram of stepper motor interfacing

**Coil Full Mode Sequence:**

****

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Steps** | **A** | **B** | **C** | **D** |  | |
| 1 | 1 | 0 | 0 | 0 | 08H | CLOCKWISE  (90o Rotation) |
| 2 | 0 | 1 | 0 | 0 | 04H |
| 3 | 0 | 0 | 1 | 0 | 02H |
| 4 | 0 | 0 | 0 | 1 | 00H |
| 5 | 1 | 0 | 0 | 0 | 08H |
| 6 | 0 | 0 | 0 | 1 | 00H | ANTICLOCKWISE  (90o Rotation) |
| 7 | 0 | 0 | 1 | 0 | 02H |
| 8 | 0 | 1 | 0 | 0 | 04H |
| 9 | 1 | 0 | 0 | 0 | 08H |
| 10 | 0 | 0 | 0 | 1 | 00H |

**Port Address:**

Port A: 80H

Port B: 82H

Port C: 84H

CW: 86H

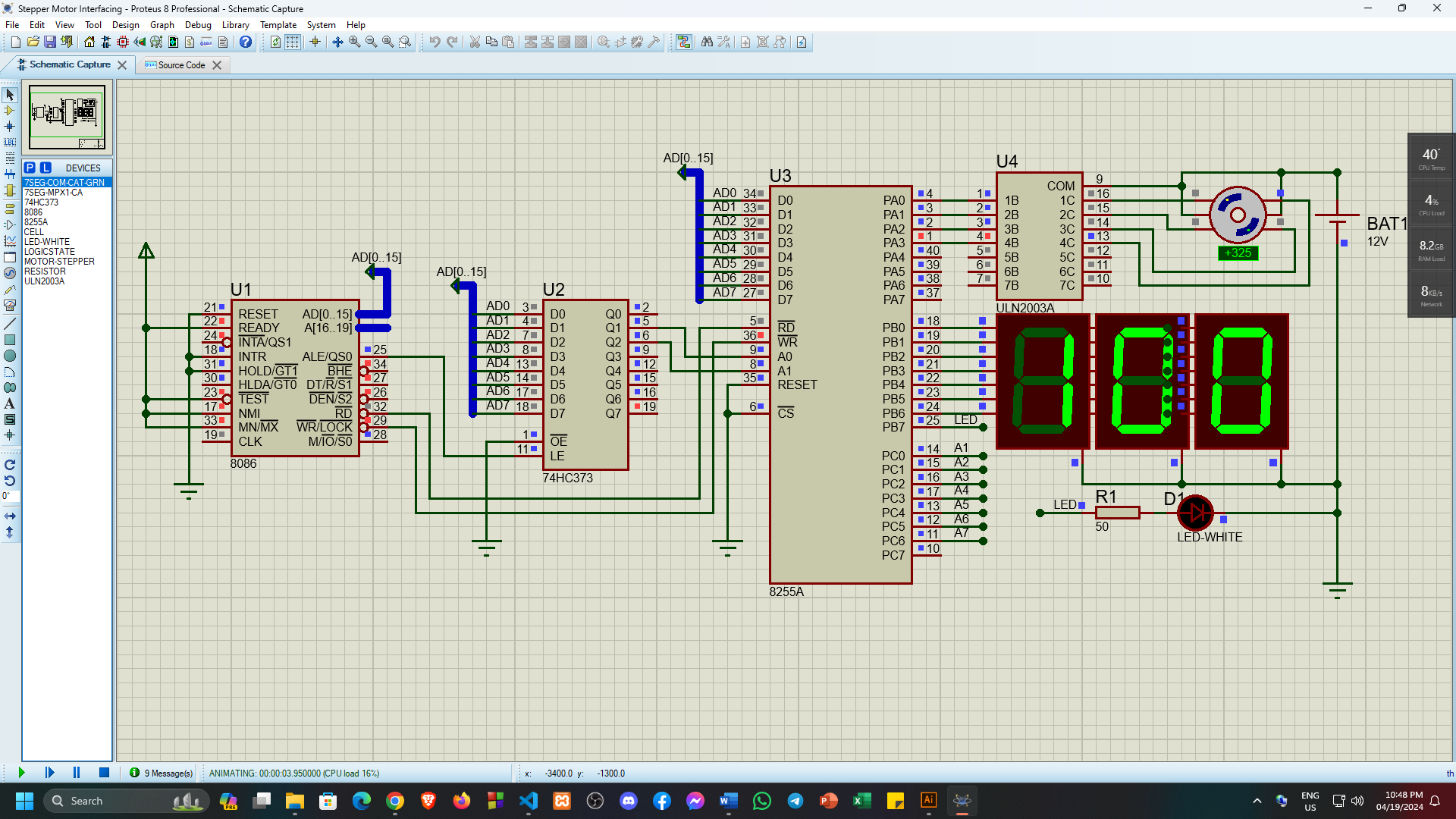
CWR: 10000000B

**Code:**

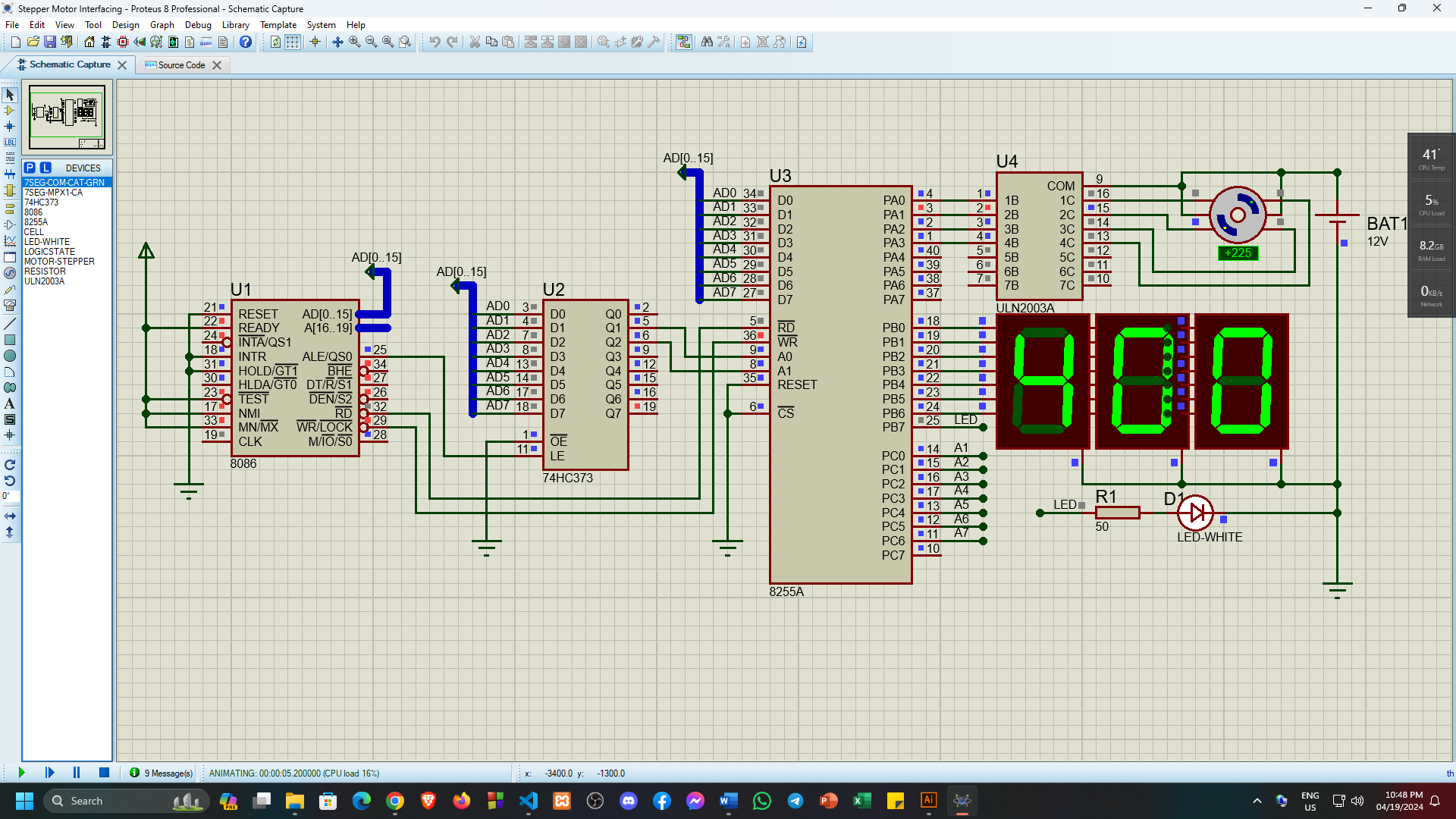
|  |  |  |
| --- | --- | --- |
| CODE SEGMENT PUBLIC 'CODE'  ASSUME CS:CODE  START:  ; port addresses  porta equ **80**h  portb equ **82**h  portc equ **84**h  config equ **86**h    mov al, **10000000**b ; i/o mode with output port  mov dx, config  out dx, al    mov dx, portc  mov al, **00111111**b ; digit 0  out dx, al    ENDLESS:  mov al, **00000110**b ; digit 1  mov dx, portb  out dx, al  mov dx, porta  mov al, **00001000**b ; clockwise  out dx, al  call delay100  mov al, **00000100**b ; clockwise  out dx, al  call delay100  mov al, **00000010**b ; clockwise  out dx, al  call delay100  mov al, **00000001**b ; clockwise  out dx, al  call delay100  mov al, **00001000**b ; clockwise  out dx, al  call delay100  mov al, **00000001**b ; anticlockwise  out dx, al  call delay100  mov al, **00000010**b ; anticlockwise  out dx, al  call delay100  mov al, **00000100**b ; anticlockwise  out dx, al  call delay100  mov al, **00001000**b ; anticlockwise  out dx, al  call delay100  mov al, **00000001**b ; anticlockwise  out dx, al  call delay100 | mov al, **11100110**b ; digit 4  mov dx, portb  out dx, al  mov dx, porta      mov al, **00001000**b ; clockwise  out dx, al  call delay400  mov al, **00000100**b ; clockwise  out dx, al  call delay400  mov al, **00000010**b ; clockwise  out dx, al  call delay400  mov al, **00000001**b ; clockwise  out dx, al  call delay400  mov al, **00001000**b ; clockwise  out dx, al  call delay400  mov al, **00000001**b ; anticlockwise  out dx, al  call delay400  mov al, **00000010**b ; anticlockwise  out dx, al  call delay400  mov al, **00000100**b ; anticlockwise  out dx, al  call delay400  mov al, **00001000**b ; anticlockwise  out dx, al  call delay400  mov al, **00000001**b ; anticlockwise  out dx, al  call delay400  JMP ENDLESS  delay100:  mov cx, **05**d01h ; 100 ms delay  l1: nop  loop l1  ret  delay400:  mov cx, **0**FED6h ; 400 ms delay  l2: nop  loop l2  ret    CODE ENDS  END START |  |

**Simulation Screenshots:**

* Rotate with 100 ms speed.



* Rotate with 400 ms speed.



**Discussion:**

In this assignment I’ve build a circuit using 8086 microprocessor, 8255 ppi, stepper motor & 7-segment display that will rotate the stepper motor and show the speed of the motor into the 7-segment display. By using 8086 I’ve rotated the stepper motor clockwise 90 degrees. For this I need to find the addresses of 8255 PPI, control word register address. For rotating the stepper motor clockwise and anticlockwise I’ve found the coils step sequences. Where the sequence steps are given in the table of stepper motor sequence. The port addresses are 80h, 82h, 84h, 86h and the control word register was 10000000B (80H). I’ve connected the stepper motor with Port A of the stepper motor and the 7-segment display with Port B & Port C. Here the 7-segment display will show the delay speed for the rotation. Where I’ve set two types of delay 100 ms delay and 400 ms delay. In the circuit when the delay is 100 ms the motor will rotate faster and when the delay is 400 ms the motor will rotate slower. And the 100 & 400 will show into the 7-segment display. Then after set all of those things I wrote the assembly language program for this circuit.