**DEPT. OF ELECTRICAL & ELECTRONICS ENGINEERING**

**SRM UNIVERSITY, Kattankulathur – 603203.**

|  |
| --- |
| Title of Experiment :11. DC Motor Speed measurement and Control |
| Name of the candidate : GAUTAM NAG  Register Number :RA1811005010278  Date of Experiment : 21/04/2021  Date of submission **:21/04/2021** |

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO:** | **MARKS SPLIT UP** | **MAXIMUM MARKS (50)** | **MARKS OBTAINED** |
| 1 | PRE LAB | 5 |  |
| 2 | PROGRAM | 25 |  |
| 3 | EXECUTION | 15 |  |
| 4 | POST LAB | 5 |  |
| TOTAL | | 50 |  |

**Staff Signature**

**11.DC Motor Speed Measurement and Control**

**PRE-LAB**

1. **What are the modes used in keyboard modes?**

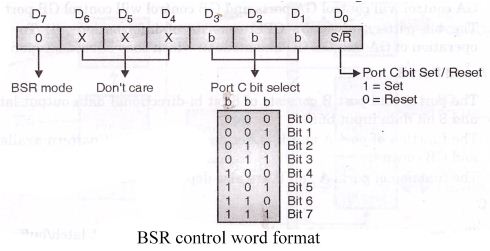
* Scanned Keyboard mode with 2 Key Lockout.
* Scanned Keyboard with N-key Rollover.
* Scanned Keyboard special Error Mode.
* Sensor Matrix Mode.

**2.What are the methods of speed control in DC Motor?**

* Resistance variation in the armature circuit: This method is called armature resistance control or Rheostat control.
* Variation of field flux Ф This method is called field flux control.
* Variation of the applied voltage.

**3.Draw the control word format BSR Mode?**

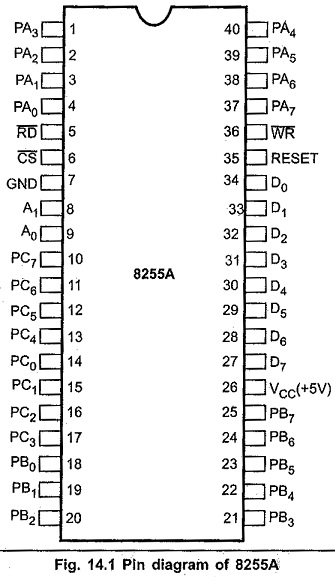
Control Word format in input/output mode Control Word and BSR Mode Format Page 2 The figure shows the control word format in the input/output mode. This mode is selected by making D7 = '1' . D0 or D1 or D3 or D4 are "SET", the corresponding ports act as input ports



**4. What are the features of 8279 Keyboard controller?**

* Simultaneous keyboard and display operations.
* Scanned keyboard mode.
* Scanned sensor mode.
* 8-character keyboard FIFO.
* Right or left entry 16-byte display RAM.
* Programmable scan timing.
* Used for Interaction between keyboard and different microprocessor.

5.Draw the pin diagram of 8255 PPI



**11. DC Motor Speed Measurement and Control**

**Aim:**

To write an assembly language program to control the speed of DC Motor using

EdSim 51.

**Apparatus required:**

**Hardware Requirement :**

8051 Microcontroller kit, Power supply

**Software Requirement :**

8051 EdSim

**EDSIM51 PROGRAM-**

MOV TMOD,#50H ; PUT TIMER 1 IN EVENT COUNTING MODE

SETB TR1 ; START TIMER

MOV DPL, #LOW (LEDcodes); I PUT THE LOW BYTE OF THE START ADDRESS OF THE 7 SEGMENT

; I CODE TABLE INTO DFL

MOV DPL, #HIGH (LEDcodes); PUT THE HIGH BYTE INTO DPH

CLR P3.4 ; I

CLR P3.3 ;I ENABLE DISPLAY 0

**AGAIN:**

CALL SETDIRECTION ;SET THE MOTOR’S DIRECTION

MOV A,TL1 ; MOVE THE TIMER 1 LOW BYTE TO A

CJNE A,#10,SKIP ;IF THE NIBBLE OF REVOLUTION IS NOT 10 SKIP NEXT INSTRUCTIONS

CALL CLEARTIMER; IF THE NIBBLE OF REVOLUTION IS 10,RESET TIMER 1

**SKIP:**

MOVC [A,@A+DPTR](mailto:A.@A+DPTR); I GET 7 SEGMENT CODE FROM CODE TABLE-THE INDEX INTO THE TABLE IS

; I DECIDED BY THE VALUE IN A

; I (EXAMPLE: THE DATA POINTER POINTS TO THE START OF THE

; I TABLE-IF THERE ARE TWO REVOLUTIONS,THEN A WILL CONTAIN TWO,

; I THEREFORE THE SECOND CODE IN THE TABLE WILL BE COPIED TO A

MOV C,F0; MOVE MOTOR DIRECTION VALUE THE CARRY

MOV ACC.7,C; AND FROM THERE TO ACC.7( THIS WILL ENSURE DISPLAY0’S DECIMAL POINT

;WILL INDICATE THE MOTOR DIRECTIONS

MOV P1,A ;I MOVE ( 7 -SEG CODE (OR) NUMBER OF REVOLUTIONS AND MOTOR DIRECTION

;I INDICATOR TO DISPLAY 0

JMP AGAIN ; DO IT ALL AGAIN

**SETDIRECTION:**

PUSH ACC ; SAVE VALUE OF A ON STACK

PUSH 20H; SAVE VALUE OF LOCATION(FIRST BIT -ADDRESSABLE IN RAM) ON STACK

CLR A;CLEAR A

MOV 20H,#0 ;CLEAR LOCATION 20H

MOV C,P2.0; PUT SW0 VALUE IN CARRY

MOV ACC.0, C; THEN MOVE TO ACC.0

MOV C,F0 ;MOVE CURRENT MOTOR DIRECTION IN CARRY

MOV 0, C ;MOV TO LSB OF LOCATION 20H

CJNE A,20,CHANGE DIR ;I COMPARE SW0(LSB OF A) WITH F0(LSB OF 20H)

;I -IF THERE ARE NOT SAME,THE MOTOR’S DIRECTION NEEDS REVERSED

JMP FINISH ; IF THEY ARE THE SAME,MOTOR DIRECTION DOES NOT NEED TO BE CHANGE

CHANGEDIR:

CLR P3.0 ;I

CLR P3.1; I STOP MOTOR

CALL CLEARTIMER; RESET TIMER I

MOV C, P2.0; MOVE SW0 VALUE TO CARRY

MOV F0, C; AND THEN TO F0-THIS IS THE MOTOR DIRECTION

MOV P3.0,C; MOVE WS0 VALUE (IN CARRY TO MOTOR CONTROL BIT 1

CPL C; INVERT THE CARRY

MOV P3.1,C ; I AND MOVE IT TO MOTOR CONTROL BIT 0

;I VALUE TO CONTROL BIT 1 AND MOTOR WILL START

; I AGAIN IN THE NEW DIRECTION

**FINISH:**

POP 20H ; GET ORIGINAL VALUE FOR LOCATION 20H FROM STACK

POP ACC ;GET ORIGINAL VALUE FOR A FROM THE STACK

RET ;RETURN FROM SUBROUTINE

CLEARTIMER:

CLR A ;RESET REVOLUTION COUNT IN A TO ZERO

MOV TR1 ;STOP TIMER 1

MOV TL1, #0 ;RESET TIMER 1 LOW BYTE TO ZERO

SETB TR1 ;START TIMER 1

RET; RETURN FRO SUBROUTINE

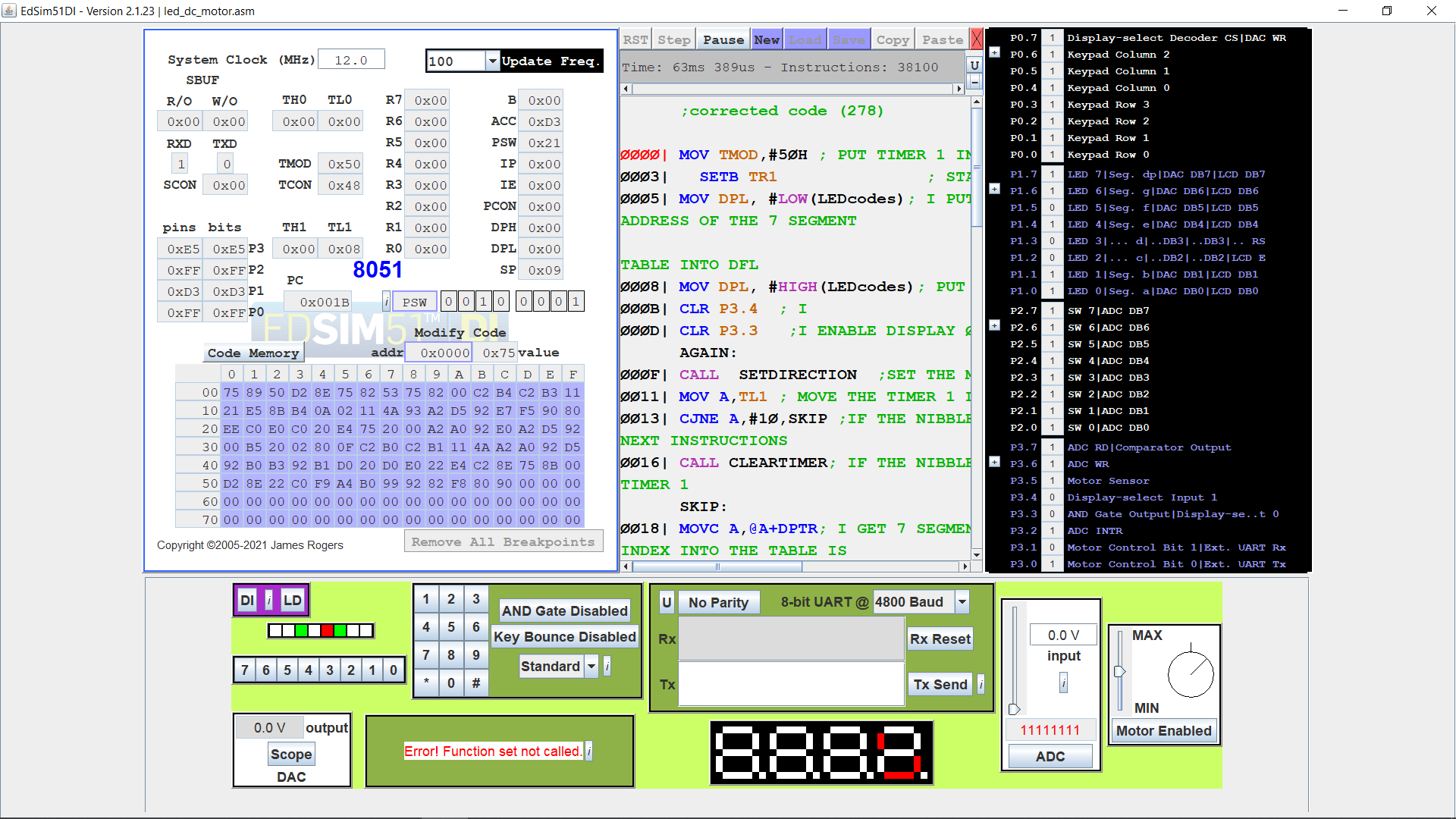
LEDCODES : I THIS LABEL POINTS TO THE START ADDRESS OF THE 7-SEGMENT CODE TABLE

; I WHICH IS STORED IN PROGRAM MEMORY USING THIS COMMAND BELOW

DB

11000000B,11111001B,10100100B,10110000B,10011001B,10010010B,10000010B,11111000B,10000000B,10010000B

**SIMULATION :**



**Result:**

Thus an assembly language program of speed control of DC Motor program was executed.

**POST-LAB**

1. **What is a programmable peripheral device?**

PPI 8255 is a general purpose programmable I/O device designed to interface the CPU with its outside world such as ADC, DAC, keyboard etc. ... It consists of three 8-bit bidirectional I/O ports i.e. PORT A, PORT B and PORT C. We can assign different ports as input or output functions

1. **What is handshake port?**

Handshaking is a technique of communication between two entities. However, within TCP/IP RFCs, the term "handshake" is most commonly used to reference the TCP three-way handshake. For example, the term "handshake" is not present in RFCs covering FTP or SMTP.

1. **What is resolution in DAC?**

The resolution of a DAC is given by the number of bits, N. The resolution is the smallest increment of output that the DAC can produce. An 8-bit, DAC has a resolution of 8 bits, or one part in 28

1. **What are the different types of ADC?**

* Successive Approximation (SAR) ADC.
* Delta-sigma (ΔΣ) ADC.
* Dual Slope ADC.
* Pipelined ADC.
* Flash ADC.

**5.List the function performed by 8279.**

* Simultaneous keyboard and display operations.
* Scanned keyboard mode.
* Scanned sensor mode.
* 8-character keyboard FIFO.
* Right or left entry 16-byte display RAM.
* Programmable scan timing.
* Used for Interaction between keyboard and different microprocessor.