→ Interfacing of keyboard with 8086 up

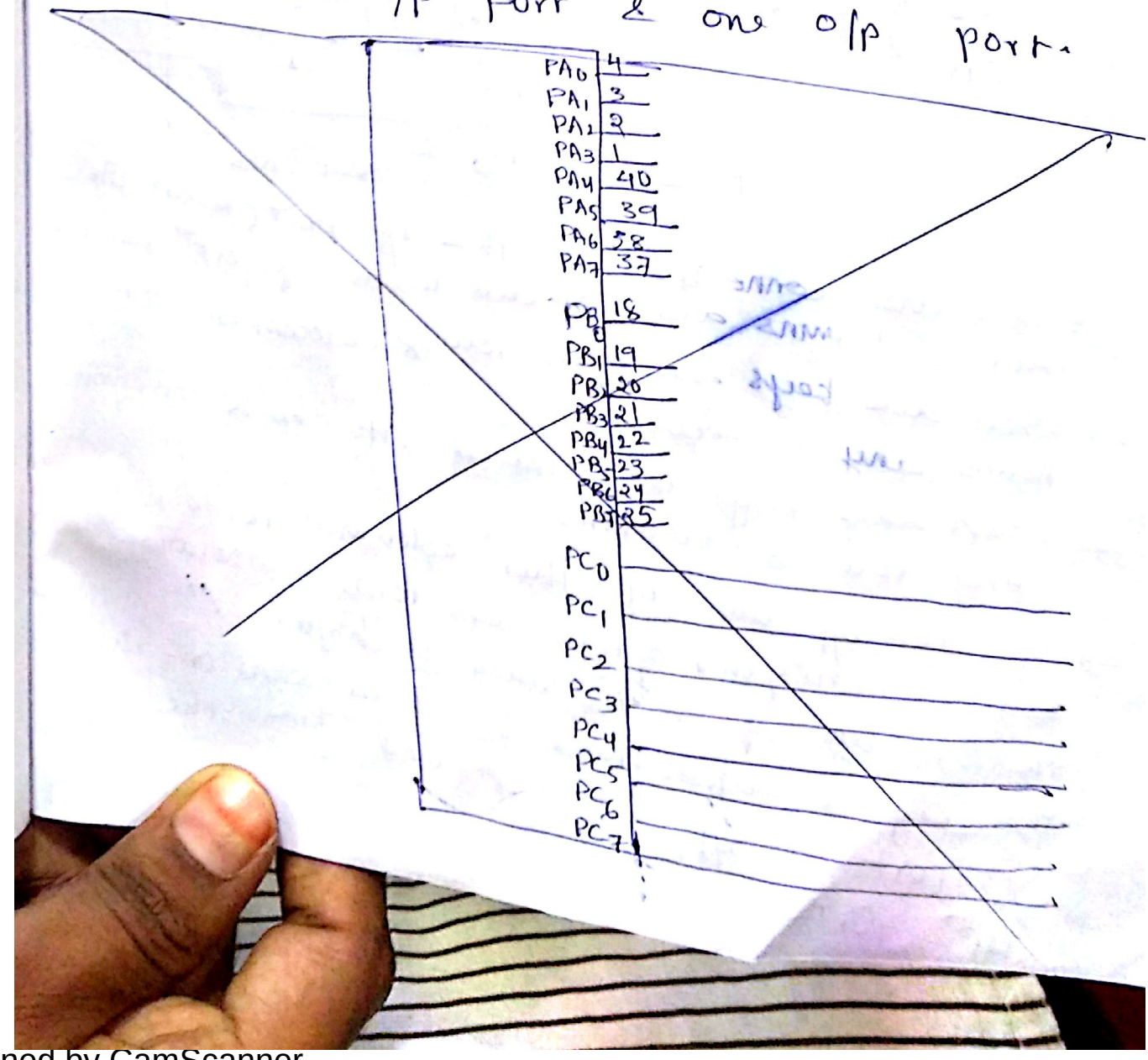
4x4 keyboard interfacing.

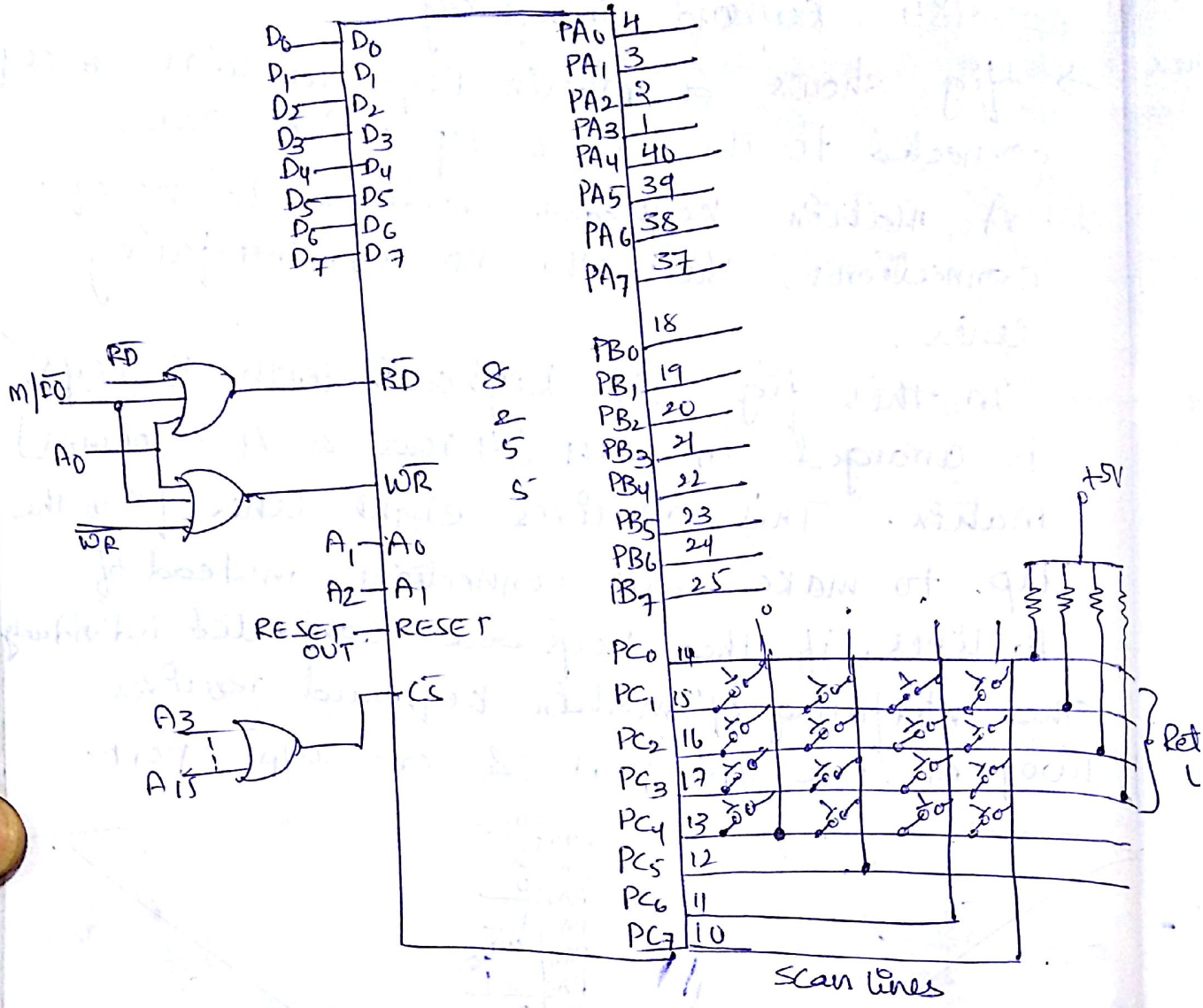
→ fig shows a matrix keyboard with 16 keys connected to the 8086 up using 8255.

→ A matrix keyboard reduces the no. of connections, thus the no. of interfacing lines.

→ In this fig the keyboard with 16 keys, is arranged in 4x4 (4 rows & 4 columns) matrix. This requires eight lines from the up to make all connections instead of 16 lines. if the keys are connected individual

The interfacing of matrix keyboard requires two ports: one i/p port & one o/p port.





→ Rows are connected to the i/p port (return lines) and columns are connected to the o/p port (scan lin

→ when all keys are open row & colum do not have any connection.

→ when any key is pressed, it shorts corresponding row & column.

→ If the o/p line of this column is low it makes corresponding row line low, otherwise the status of row lines is high.

→ The key is identified by data sent on the o/p port & i/p code received from the i/p port.

When any key is pressed or not.

→ Make all colour lines zero by sending low on all o/p lines. This activate all keys in the keyboard matrix.
(i.e: When scan lines are logic high, the status on the return lines do not change, it will remain logic high.)

→ Read the status of return lines. If the status of all lines is logic high, key is not pressed; otherwise key is pressed.

→ Activate keys from any one column by making any one column line zero.

→ Read the status of return lines. The zero on any return line indicates key is pressed from the corresponding row & selected column. If the status of all lines is logic high, key is not pressed from that column.

→ Activate the keys from the next column & repeat 2 & 3 for all columns.

→ scan lines are connected to the port C_L of 8255 & return lines are connected to the port C_U of 8255.

Return lines

Display Interfacing :-

→ Most of the µp & µc instruments & machines need to display letters of the alphabet & numbers to give directions or data values to the users.

→ This information can be displayed using CRT, LED or LCD displays.

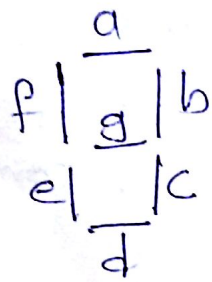
→ CRT displays are used when a large amount of data is to be displayed.

→ The system where a small amount of data is to be displayed ; a simple LED & LCD displays are used.

LED displays

7 segment display is a LED display.

→ 7 segment displays are generally used as numerical indicators & consists of a no. of LED's arranged in seven segments



→ Any number b/n 0 to 9 can be indicated by lighting the appropriate segments.

→ The 7 segments are labeled a to g & dot is labled as h.

→ by forward biasing different LED segments we can display the digits 0 to 9.

→ For '0' → we need to light up a, b, c, d, e & f

" 5 → a, f, g, c & d.

→ 7 segment displays are of two types

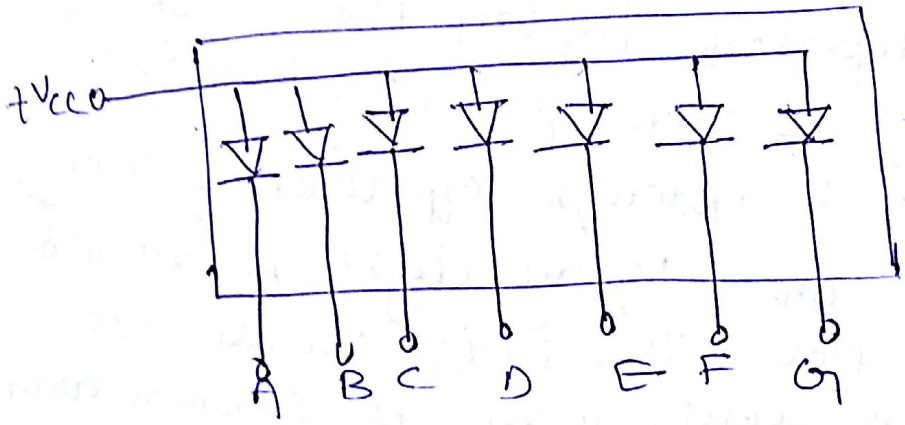
(1) → common anode type

(2) → " Cathode "

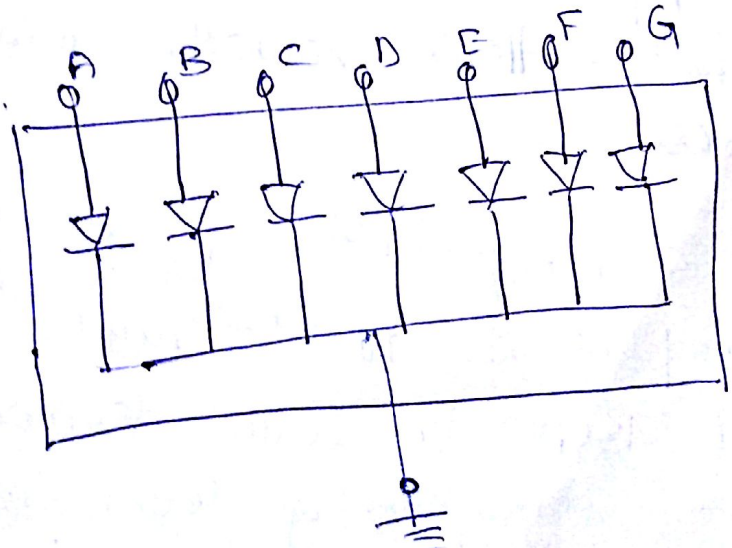
→ In common anode all anodes of LEDs are connected together &

→ In common cathod, all cathodes are connected together.

(1) common anode type



(2) common cathod type



Interfacing LED Displays

→ Multiplexed display method is used

→ fig shows the 4 seven segment displays connected using multiplexed method. Here, common anode seven segment LED's are used.

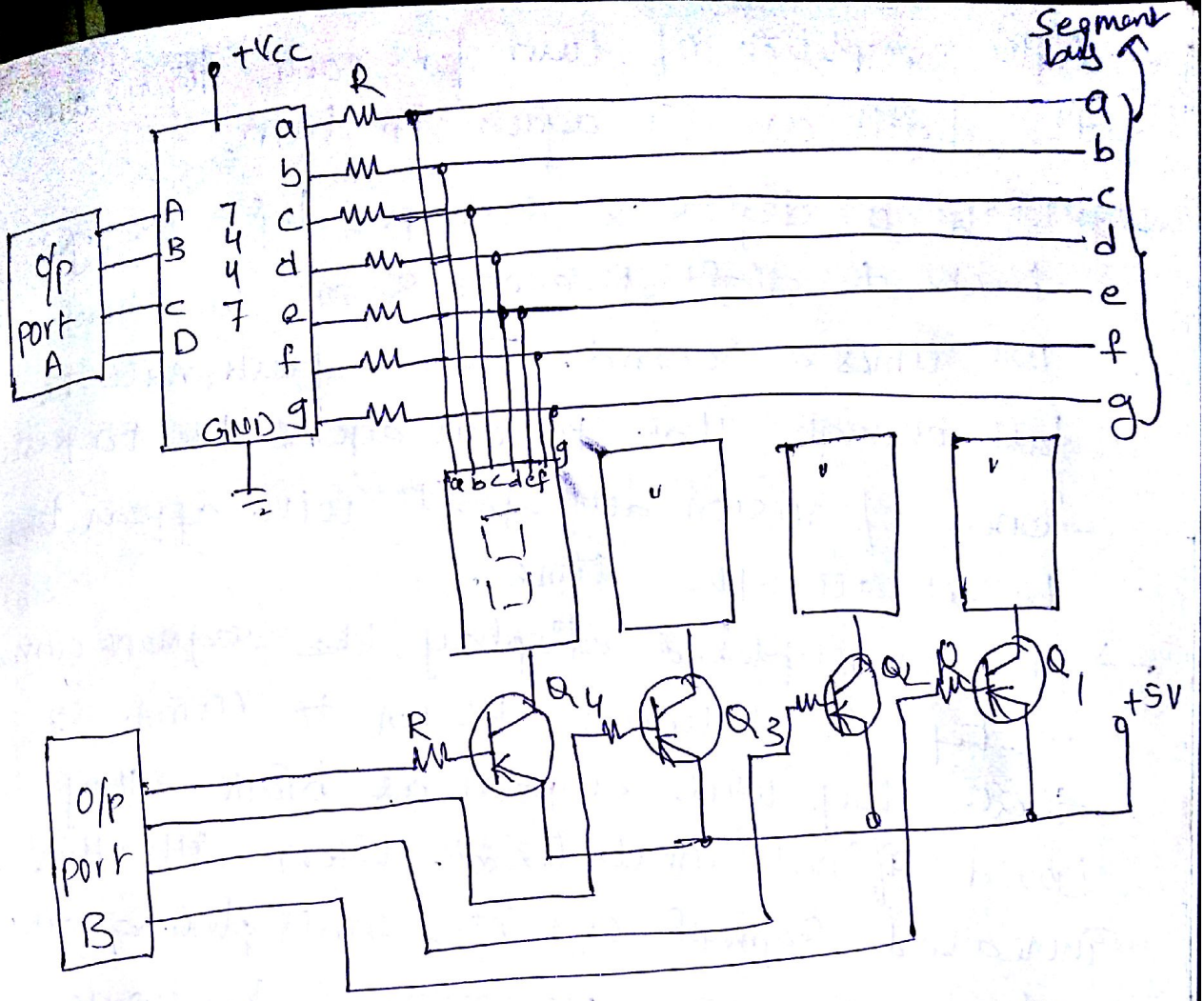
→ Anodes are connected to +5V through transistors. Cathods of all seven segments are connected in parallel & then to the o/p of 7447 IC (BCD to seven segment decoder) through resistors.

→ They would show the same number only if all the digits are turned on at the same time.

→ In Multiplexed display the segment information is send for all digit on the common lines (through o/p lines of 7447) but only one display digit is turned on at a time. The PNP transistors connected in series with the common anode of each digit act as an ON & OFF switch for that digit. Here's how the multiplexing process works.

→ The BCD code for digit 1 is first o/p from port A, to the 7447.

→ The 7447, BCD to seven segment decoder o/p's the corresponding seven segment code on the segment LED lines



→ The transistor Q1 connected to digit 1 is then turned on by outputting a low to that bit of port B. all of the rest of the bits of port B are made high to ensure no other digits are turned on.

→ After 2 ms, digit 1 is turned OFF outputting all highs to port B.

→ The BCD code for digit 2 is then o/p to the port A & bit pattern to turn on digit 2 is o/p on port B.

→ After 2 ms, digit 2 is turned off & the process is repeated for digit 3. & digit 4.

→ After completion of turn for each digit, all the digits are lit again in turn.

→ With 4 digits & 2 ms per digit we get back to digit 1 every 8 ms or about 125 times a second. This refresh rate is fast enough that, to our eye & due to persistence of vision all digits will appear to be lit all the time.

→ In Multiplexed display, the segment current is kept in between 40 mA to 60 mA so that they will appear as bright they would if not multiplexed. Even with this increased segment current, multiplexing gives a large saving in power & hardware components.

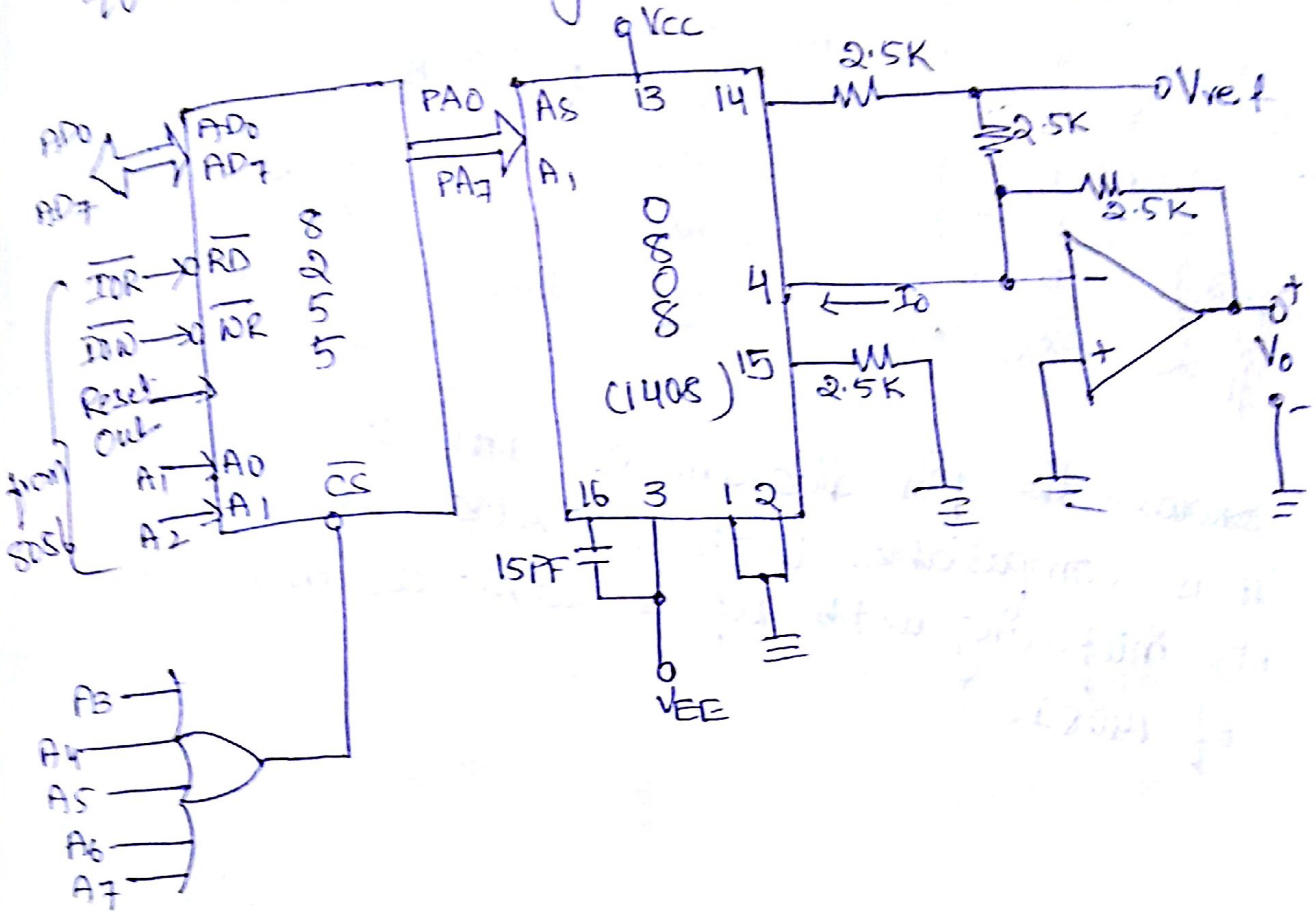
→ In multiplexed 7-segment display connected in 8086 with 8255 port A & port B are used as simple latched o/p ports. port A provides the segment data i/p's to the display & port B provides a means of selecting a display position at a time for multiplexing the displays. 8255 is addressed using direct addressing mode so only A0-A7 lines are used to decode the address for 8255

PA = 00 H PC = 04 H

PB = 02 H CR = 06 H.

DAC Interface :-

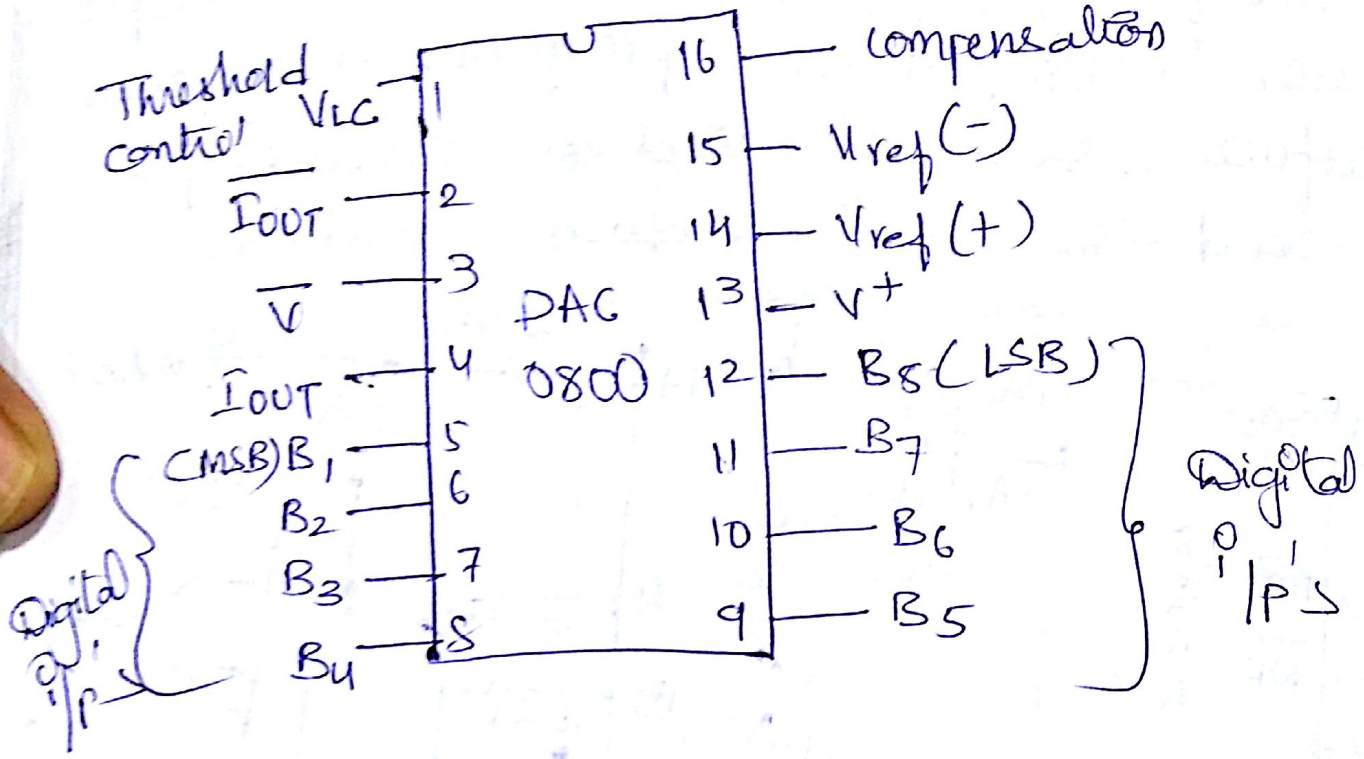
→ Interfacing of DAC 0808 with microprocessor 8086. Here programmable peripheral interface 8255 is used as parallel port to send the digital data to DAC



I/O map for 8255

port / Register	Address
Port A	00
Port B	02
Port C	04
control Register	06

DAC 0800 IC



shows the pin diagram of DAC 0800. It is compatible with IC 1408 - thus its interfacing with µp is same as that of 1408 IC.

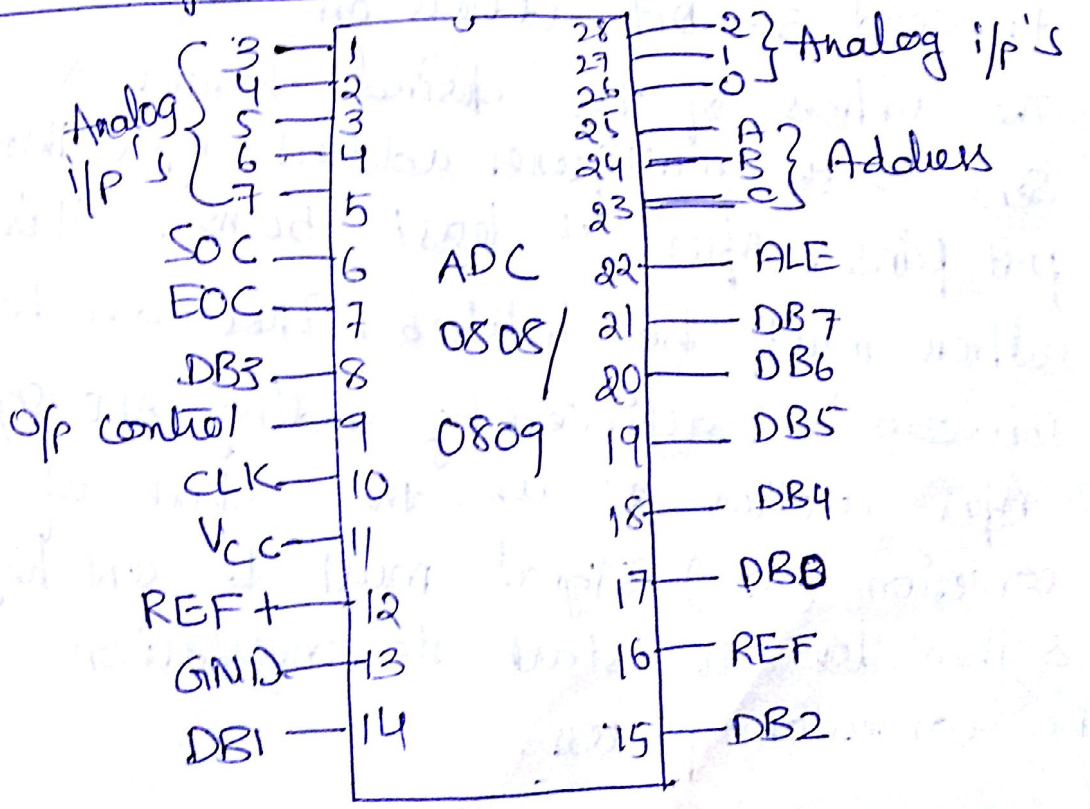
ADC Interfacing

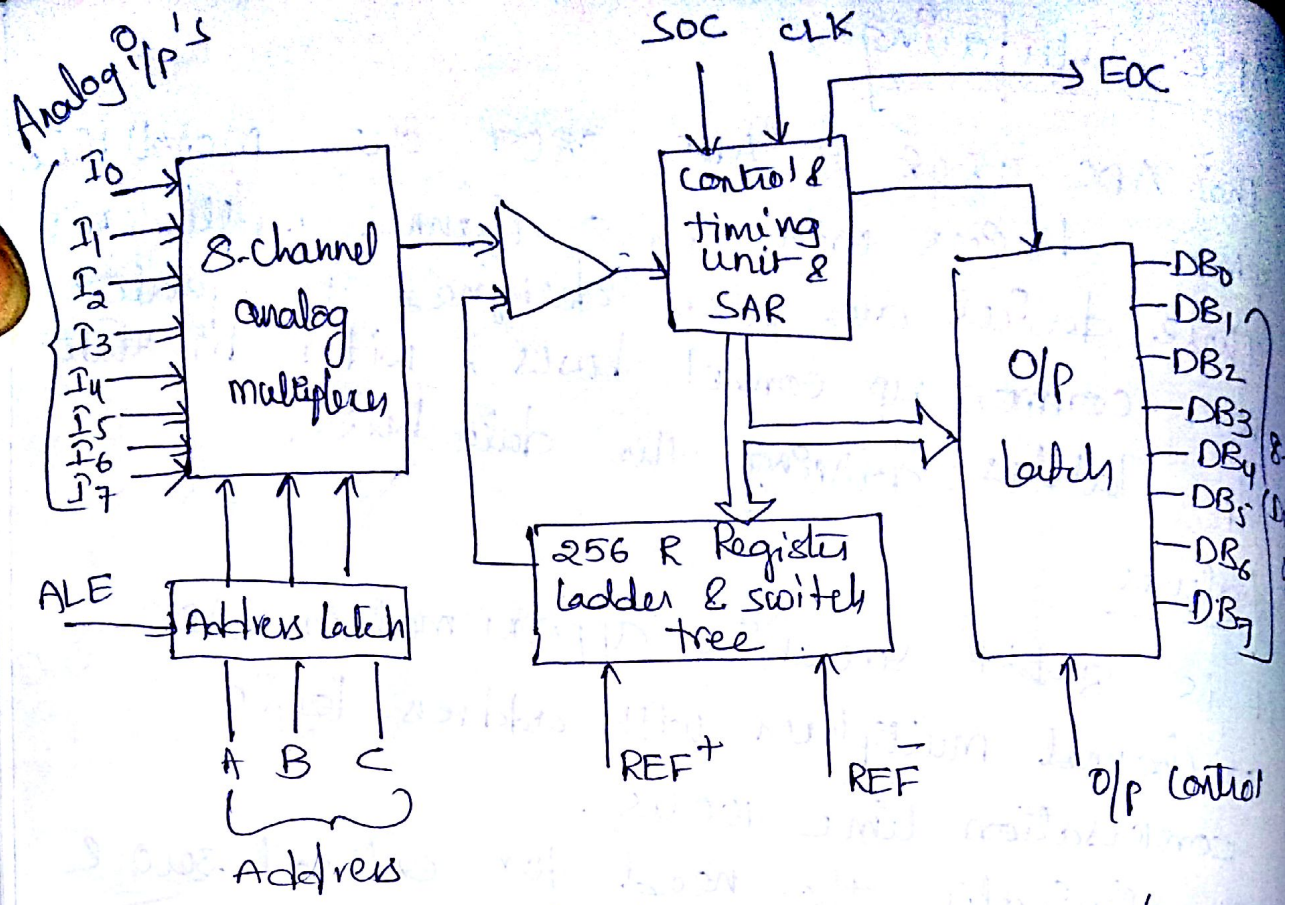
The ADC 0808 & ADC 0809 are monolithic CMOS devices with an 8-channel multiplexer. These devices are also designed to operate from common µp control buses, with tri-state o/p latches driving the data bus.

features

→ It is 8-bit successive approximation ADC.
→ 8-channel multiplexer with address logic.
→ conversation time 100 µs.
→ It eliminates the need for external zero & full-scale adjustments.
→ Easy to interface to all µps.
→ It operates on single 5V power supply.
→ o/p meet TTL v/g level specifications.

pin-diagram





Block diag of ADC 0808/0809

Operation

→ ADC 0808/0809 has 8 i/p channels, so to select desired i/p channel, it is necessary to send 3-bit address on A, B, & C i/p

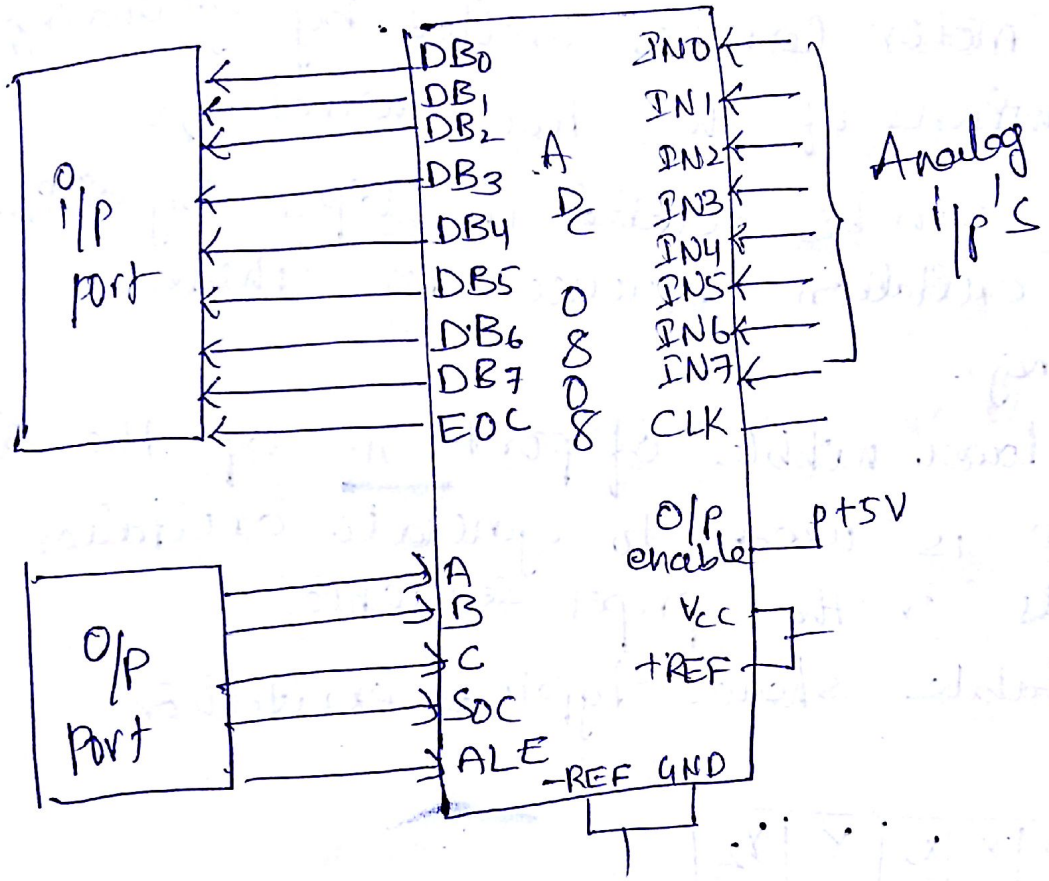
→ The address of the desired channel is sent to the multiplexer address i/p's through port pins. After at least 50 ns, this address must be latched. This can be This can be achieved by sending ALE signal

→ After another 2.5 us, the start of conversion (SOC) signal must be sent high & then low to start the conversation i.e conversion process.

→ To indicate end of conversion ADC activates EOC sig.

→ The up sys can read converted digital word through data bus by by enabling the o/p enable sig after EOC is activated.

Interfacing



Stepper motor Interfacing

→ a stepper motor is a digital motor. It can be driven by digital sig.

→ fig shows typical 2-phase motor interfacing using 8255. It has centre tap winding.

→ The centre taps of these windings are connected to the 12V supply. due to this, motor can be excited by grounding four terminals of the two windings.

→ motor can be rotated in steps by giving proper excitation sequence to these windings.

→ The lower nibble of port A of the 8255 is used to generate excitation signals in the proper sequence

→ The table shows typical excitation ser

step	X_1	X_2	Y_1	Y_2
1	0	1	0	1
2	1	0	0	1
3	1	0	1	0
4	0	1	1	0
1	0	1	0	1

→ The given excitation sequence rotates the motor in clockwise direction.

→ To rotate motor in anticlockwise direction we have to excite motor in a reverse sequence.

The excitation seq is called full step sequence in which excitation ends of the phase are changed in one step.

→ Interfacing -

→ It has two phases, with centre tap winding. the centre taps of these winding are connected to 12V supply.

→ Due to this, motor can be excited by grounding four terminals of the two windings

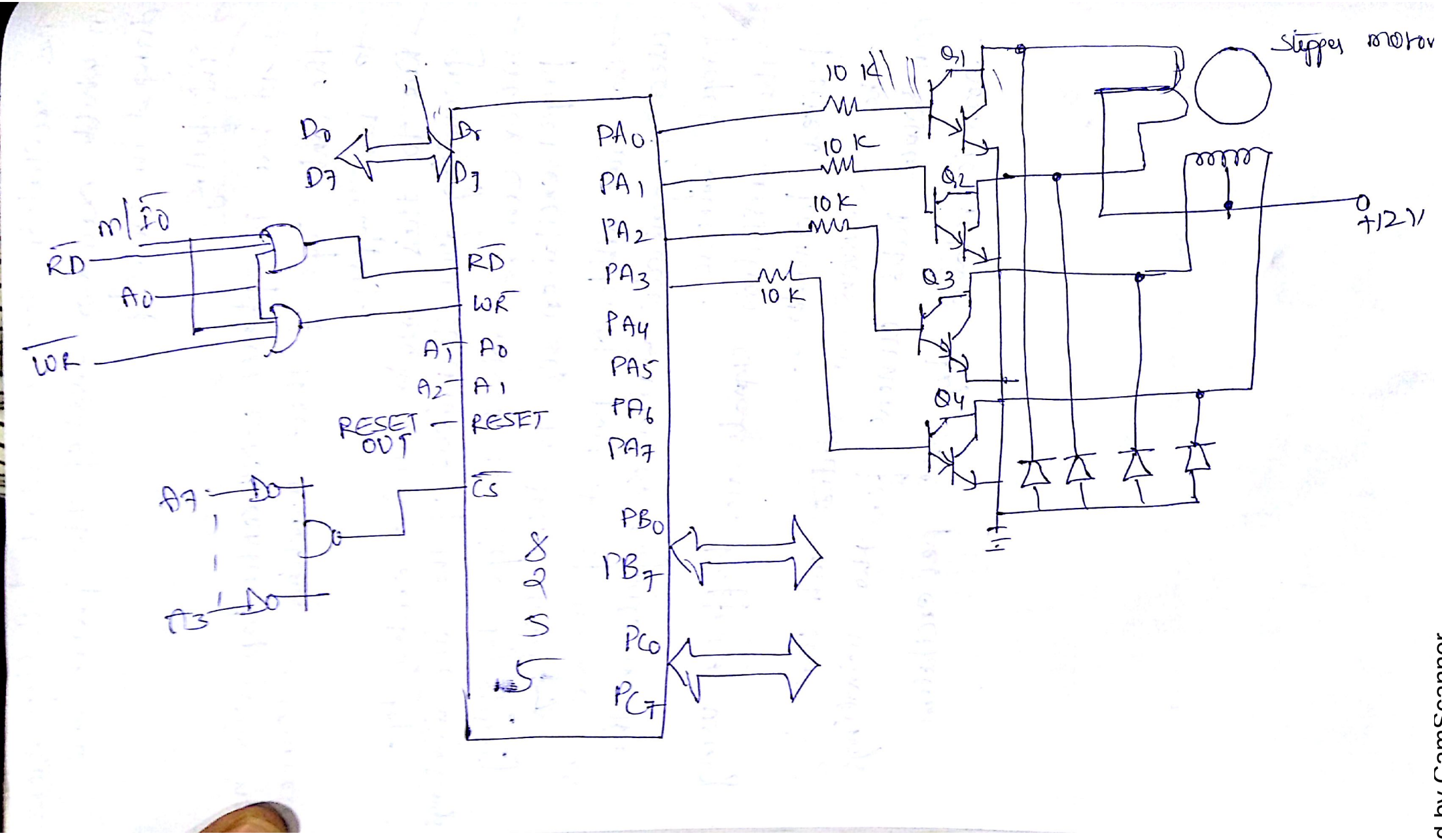
→ motor can be rotated in steps by giving proper excitation seq to these windings.

→ The lower nibble of port A of the 8255 is used to generate excitation signals in the proper sequence.

→ these excitation signals are buffered using driver transistors.

→ The transistors are selected such that they can source rated current for the windings

→ motor is rotated by 1.8° per excitation.



1) Addition & Subtraction of two 16-bit registers

2) Multiplication of Signed 16-bit numbers
Division " " " "

3) ASCII Adjust after addition.

4) The sum of n natural numbers.

5) Sum of cubes of natural numbers.

6) Find the maximum number in given string.

7) " " minimum " " "
" .

8) write a program (ALP) to finding Ascending order

9) " " " " " "
descending order.

10) DAA (Decimal adjust after addition)
DAS " " " "