LCD Interfacing:

LCD is finding widespread use replacing LEDs.

* The declining prices of LCD.
* The ability to display numbers, characters + graphics.
* Incorporation of refreshing controller into LCD.
* Ease of programming for character + graphics.

Pin description + operation:

RS: Register Select

There are 2 registers inside LCD.

If RS=0 the register is selected, allowing user to send command such as clear display, cursor etc..

If RS=1 the register is selected, allowing user to send data to be displayed on LCD.

R/w: Read / write

R/w allows user to write information to LCD or read information from it.

R/W=1 when reading. R/W=0 when writing.

E: Enable.

The enable pin used by LCD to latch information presented to its data pins.

D0-D7: Data lines.

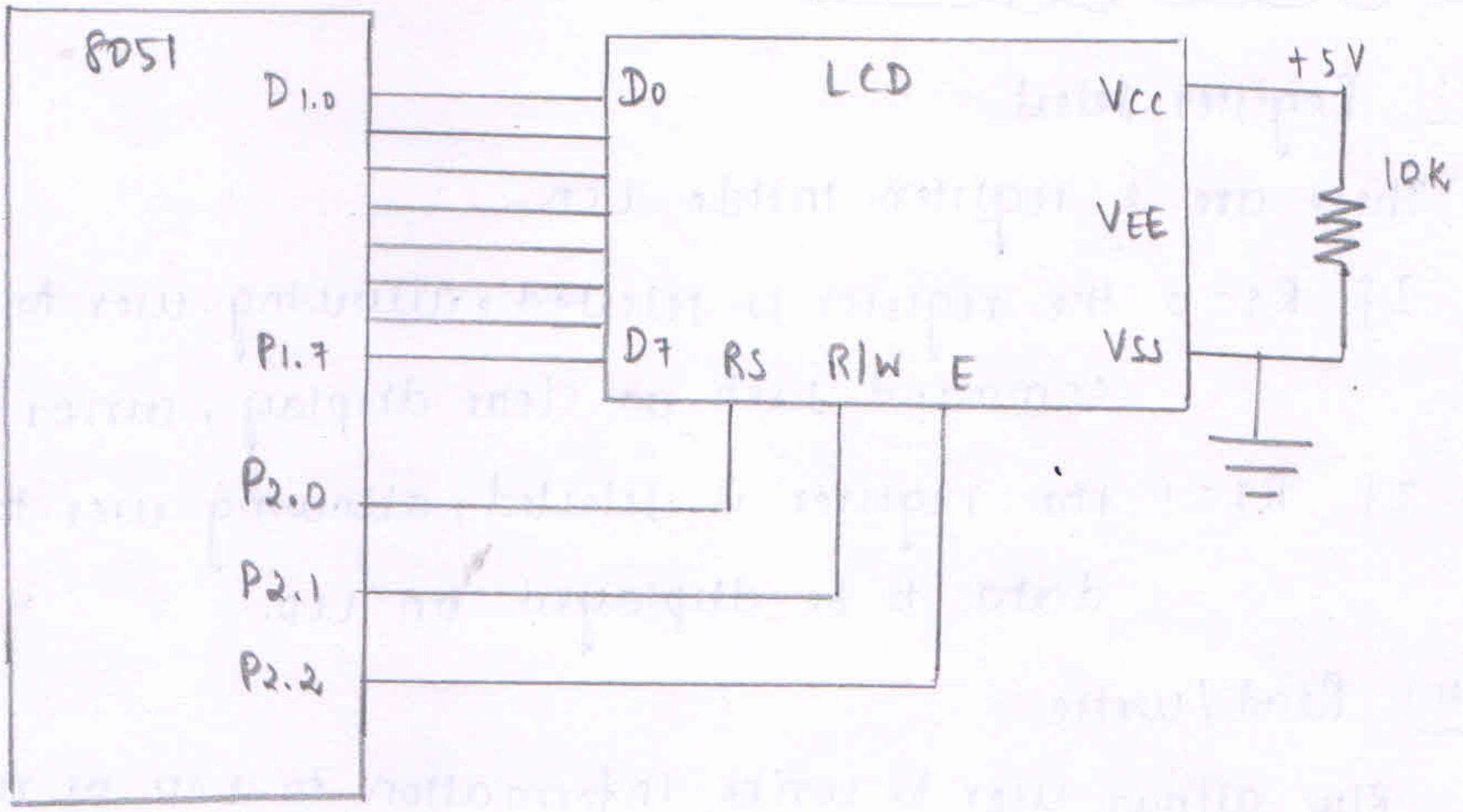
The 8 bit data pins used to send information to LCD or read the contents of LCD's registers. While RS=1, to display letters & numbers send ASCII codes for letters A-Z, a-z and

While RS=0 used to check busy flag bit (D7)

If RS=0, R/W=1 D7=1 busy flag is set & LCD is busy taking care of operations.

If RS=0, R/W=1 D7=0 busy flag is reset & LCD is ready to receive information.

Fig: LCD interfacing with 8051



LCD program:

Program to send commands & data to LCDs with timedelay

Calls a time delay before sending next data / command.

P1.0 - P1.7 are connected to LCD data pins D0-D7

P2.0 is connected to RS pin of LCD.

P2.1 is connected to R/W pin of LCD.

P2.2 is connected to E pin of LCD.

```
        ORG     0H
        MOV     A,#38H
        ACALL   COMNWRT
        ACALL   DELAY
        MOV     A,#0EH
        ACALL   COMNWRT
        ACALL   DELAY
        MOV     A,#01
        ACALL   COMNWRT
        ACALL   DELAY
        MOV     A,#06H
        ACALL   COMNWRT
        ACALL   DELAY
        MOV     A,#84H
        ACALL   COMNWRT
        ACALL   DELAY
        MOV     A,#'N'
        ACALL   DATAWRT
        ACALL   DELAY
        MOV     A,#'O'
        ACALL   DATAWRT
AGAIN:  SJMP    AGAIN
COMNWRT:
        MOV     P1,A
        CLR     P2.0
        CLR     P2.1
```

```
        CLR   P2.2
        RET
DATA WRT:
        MOV   P1,A
        SETB  P2.0
        CLR   P2.1
        SETB  P2.2
        ACALL DELAY
        CLR   P2.2
        RET.
DELAY:  MOV   R3, #50
HERE2:  MOV   R4, #255
HERE:   DJNZ  R4, HERE
        DJNZ  R3, HERE2
        RET
        END.
```

keyboard interfacing:

Keyboards are organized in a matrix of rows & columns. The CPU accesses both rows & columns through ports. 8x8 matrix keys connected. When key is pressed, a row & column make a contact, otherwise no connection between rows & columns.

Scanning & identifying keys:

1) 4x4 matrix connected to two ports.
2) The rows connected to an output port
3) The columns connected to an input port.

4) If no key has been pressed, reading the input port will make all columns as 1 since it is connected to Vcc.

5) If key is pressed, one of the column will have 0 since it is provide path to ground.

It is the function of microcontroller to scan keyboard continuously to detect & identify the key pressed.

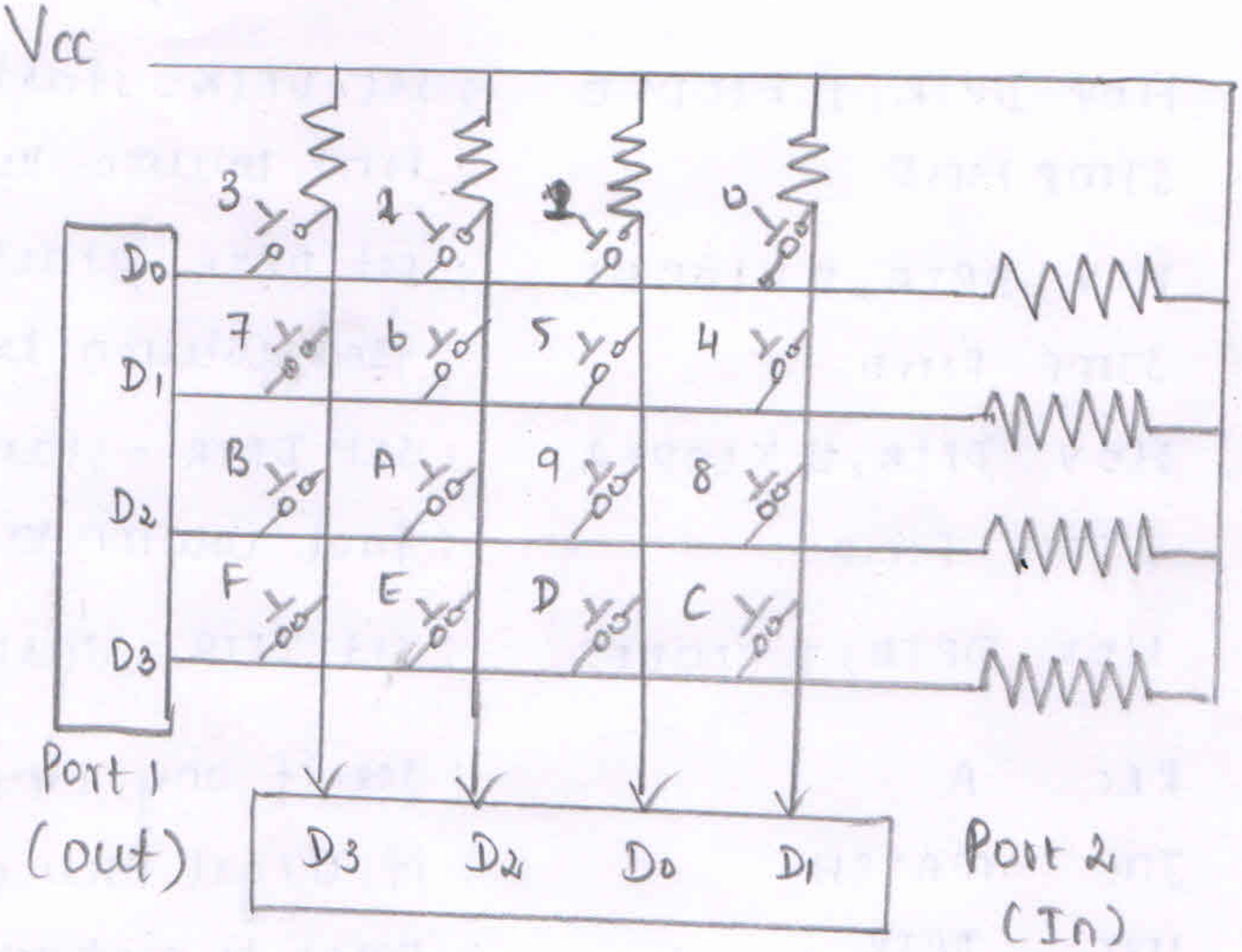


Fig: Matrix keyboard connection to ports

Algorithm to detect & identify key:

1) To make sure that preceding key has been released, 0 as output to rows 1 as output to columns.

2) If key pressed, the columns are scanned over until it has 0 on it.

3) To detect which row key pres, it grounds one row reading the column each time.