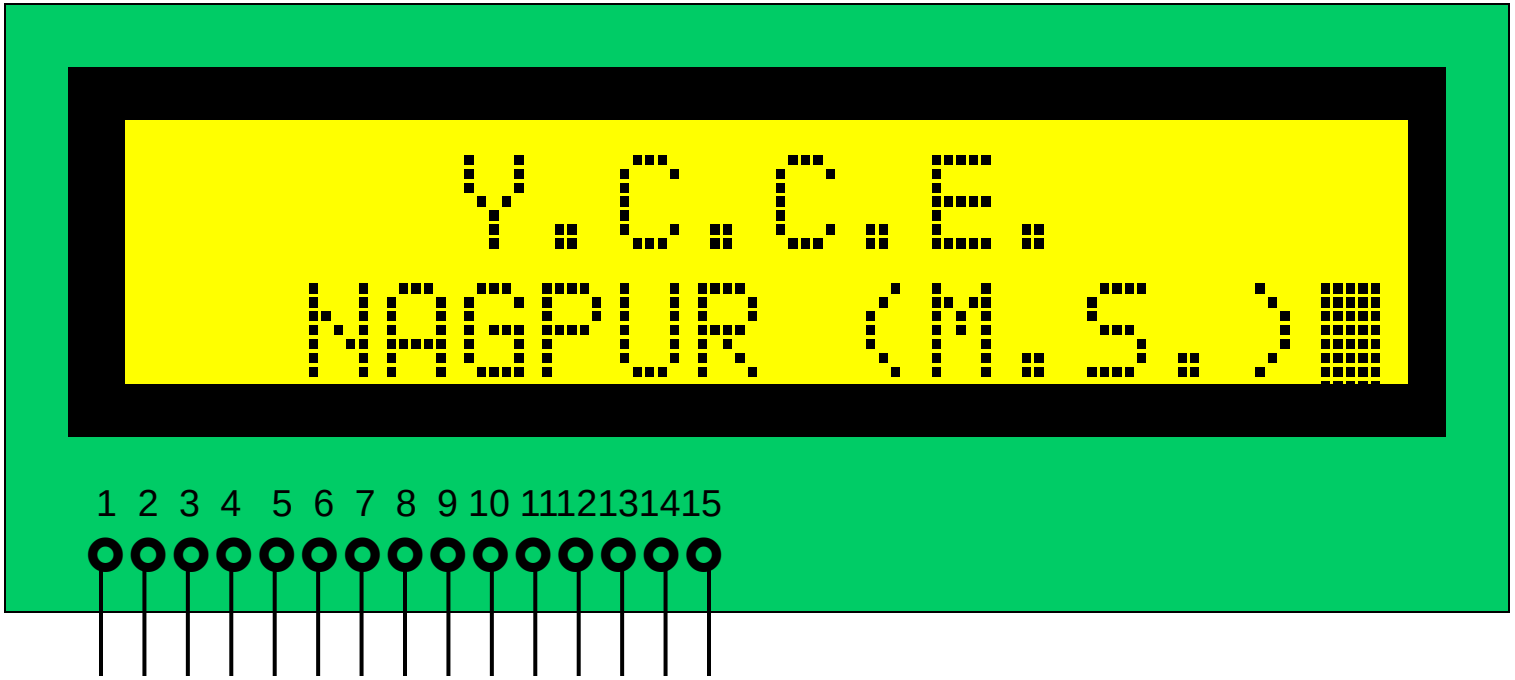
YCCE, Nagpur

MGI

LCD interfacing with 8051

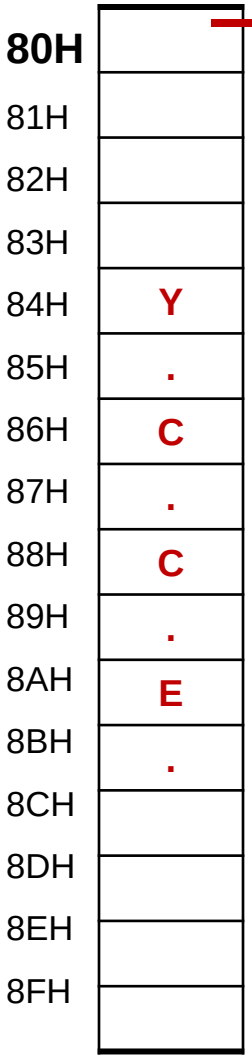
Dr. P. T. Karule
Professor
Department of Electronics Engineering
Yeshwantrao Chavan College of Engineering, Nagpur (M. S.)
Email: ptkarule@ycce.edu
Website: www.ycce.edu

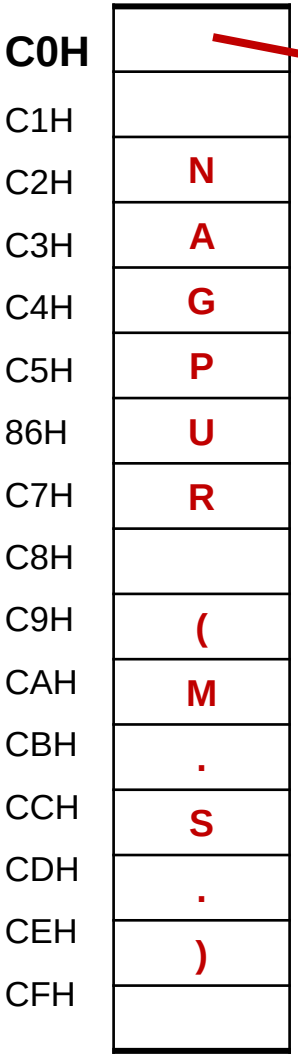
LCD 2 X16

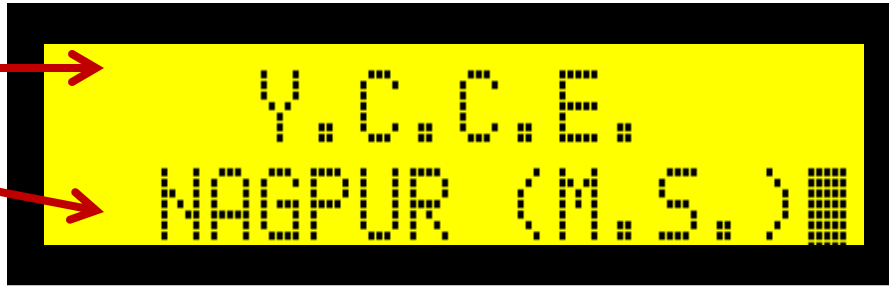


Pin No.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Symbol	GND	VCC	CONT	RS	R/W	EN	D0	D1	D2	D3	D4	D5	D6	D7	BL

LCD 2 X16 (Features)

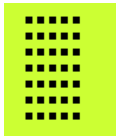






LCD Display RAM With address

LCD 2 X16 (Features)



- 5 x 7 dots plus cursor, 5 x 10 dots per character
- 4 or 8 bit interface with MPU is possible
- Display Data RAM (DR) : 80 x 8 bit, Max 80 characters
- Character Generator (CG) ROM : 160, 5 x 7 Character fonts
  32, 5 x 10 Character fonts
  Custom ROM codes available
- Character Generator RAM : Program write (64 x 8 bits)
  8, 5 x 7 Character fonts
  4, 5 x 10 Character fonts
- Both Display Data RAM & CG RAM can be read by MPU
- **Wide variety of operating instructions :**
  Display Clear, Cursor Home, Display ON / OFF,
  Display Cursor Blink, Cursor shift, display shift etc.

LCD 2 X16 (Pin Description)

Signal	In/Out	Function
D7 – D0	I / O	8 bit data bus used for bi-directional data transfer between µC & LCD module
D7 – D4	I / O	These lines are used as data bus in 4 bit operation.
D7	O	Can be used as Busy Flag
RS	I	Register Select 0 = Instruction Register (IR) 1 = Data Register (DR)
R / W	I	Signal to select Read or Write 0 = Write, 1 = Read
EN	I	Enable signal to start read / write operation

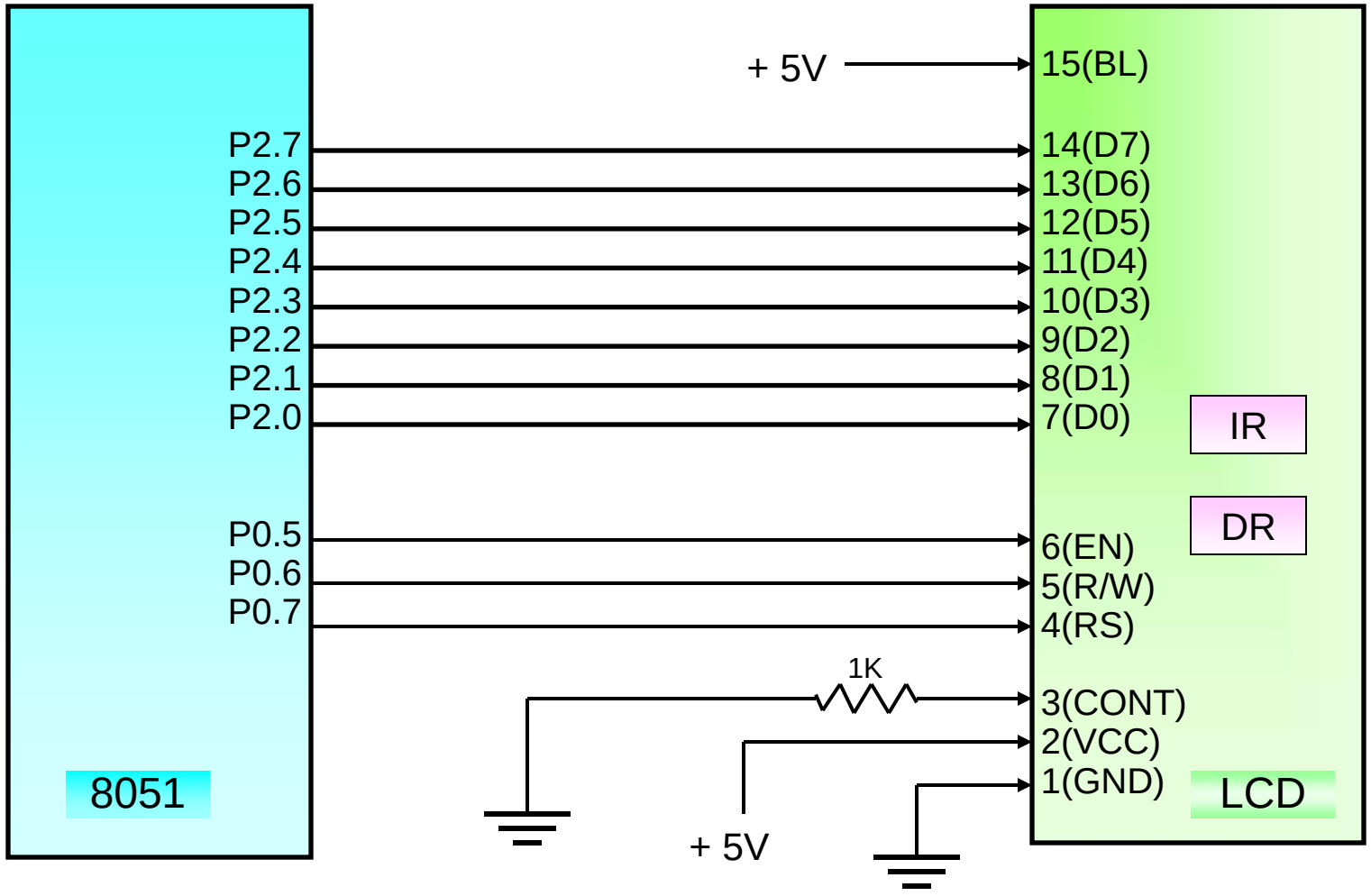
LCD 2 X16 (Registers)

- Controller for LCD has two 8 bit registers, the instruction register (IR) and data register (DR)
- IR is write only register used to store only instruction codes like display Clear, cursor shift or address of DD RAM etc.
- DR is read / write register used to store temporary data into DD RAM or CG RAM by an internal operation of the display controller.

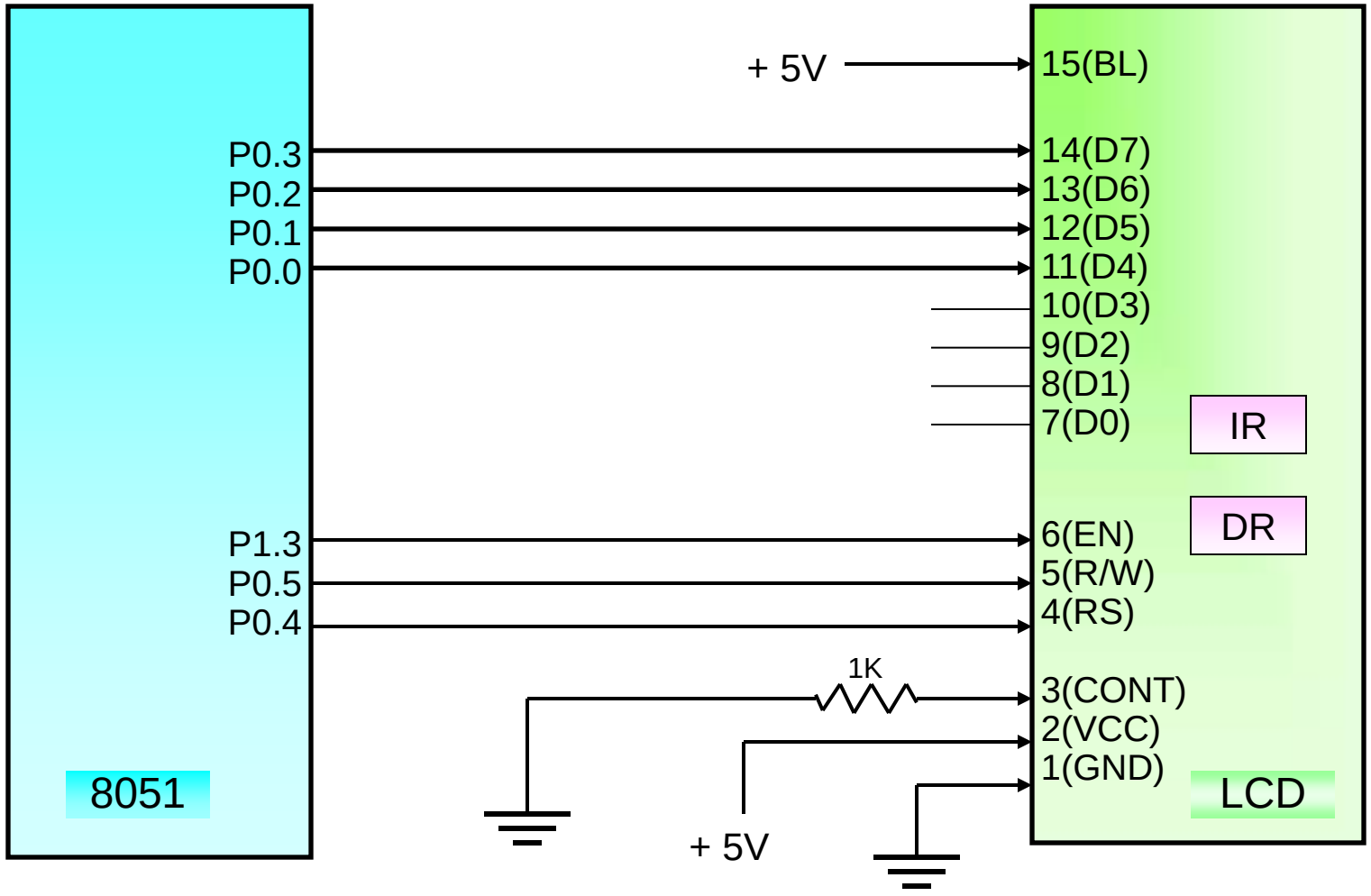
RS	R / W	Operation
0	0	IR write, internal operation (Display clear, cursor position, function set etc.)
0	1	Busy Flag (D7) & address counter (D6-D0) read
1	0	DR write, internal operation (DD RAM or CG RAM)
1	1	DR read, internal operation (DD RAM or CG RAM)

- If BF=1, then module is performing internal operation and next instruction will not be accepted

Interfacing LCD (8 bit data length)



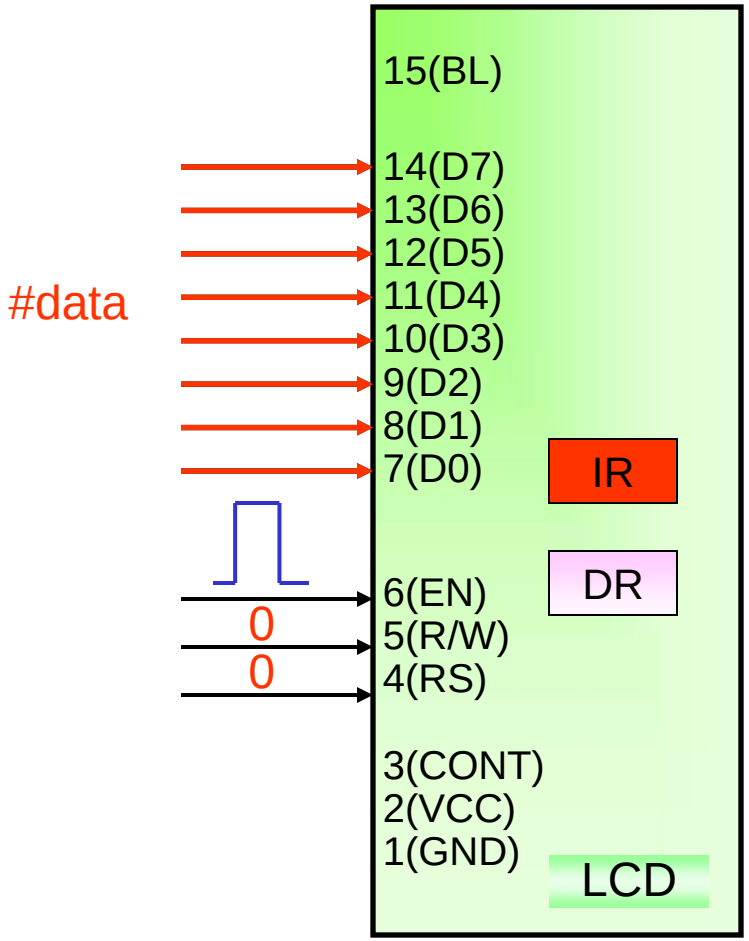
Interfacing LCD (4 bit data length)



Write Operation (IR)

To write data into IR

1. Select instruction Reg. (RS = 0)
2. Select write (R/W = 0)
3. Transfer instruction code to Port (Via Data bus)
4. Enable signal to start write operation (EN = 1)



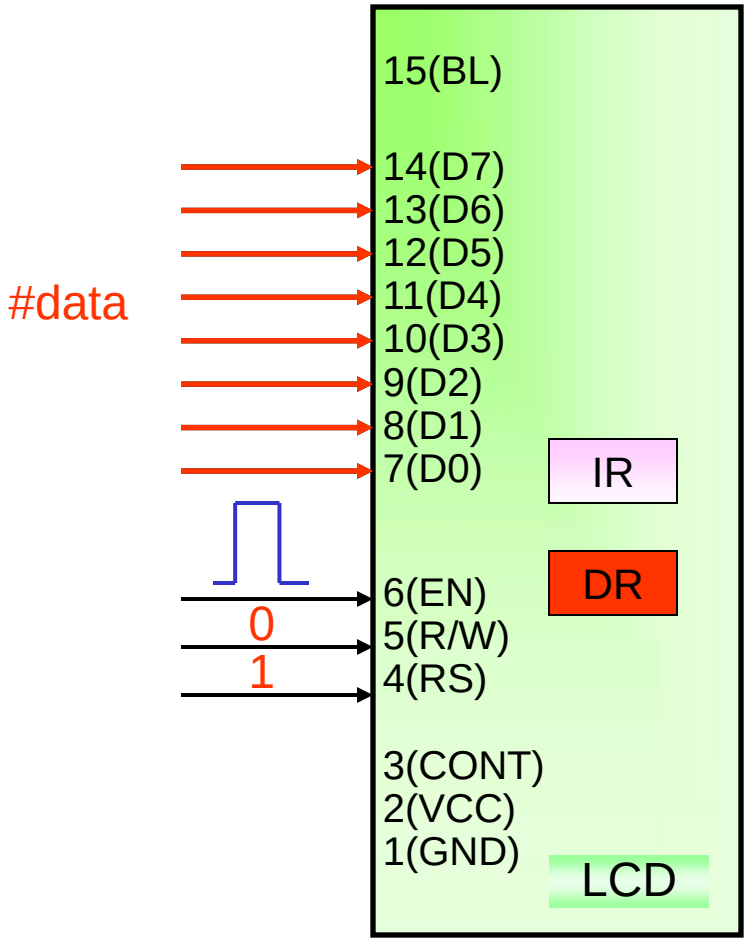
Sub program to write data into IR

```
CLR  RS
CLR  RW
MOV  Port, # data
SETB  EN
ACALL delay_1ms
CLR  EN
ACALL  delay_5ms
```

Write Operation (DR)

To write data into DR

1. Select Data Reg. (RS = 1)
2. Select write (R/W = 0)
3. Transfer ASCII code of display character to Port
4. Enable signal to start write operation (EN = 1)



Sub program to write data into DR

```
SETB  RS
CLR  RW
MOV  Port, # data
SETB  EN
ACALL delay_1ms
CLR  EN
ACALL  delay_5ms
```

LCD instructions (8 bit instr. code)

1. Function Set:
   - DL = Data Length (If DL=1, 8 bit & If DL=0, 4 bit)
   - N = No. of display lines (If N=0, 1 Line & If N=1, 2 Line)
   - F = Character Font (If F=0, 5x7 dot character font & F=1, 5x10
2. Display ON/OFF Control:
   - D = 0 ; Display OFF & D = 1 ; Display ON
   - C = 0 ; Cursor OFF & C = 1 ; Cursor ON
   - B = 0 ; Blink OFF & B = 1 ; Blink ON
3. Clear Display
4. Entry Mode Set:
   - I/D = 1 ; Increment Mode (Increment DD RAM address by 1)
   - S = 0 ; No display shift operation
5. Set DD RAM address:
   - Sets the address counter to the DD RAM address AAAAAAA. Data is then written / read to from the DD RAM.
   - For 2 line display module

     AAAAAAA = 00H to 27H for the first line &

     40H to 67H for the second line.

LCD instructions…

1. Function Set

D7	D6	D5	D4	D3	D2	D1	D0
0	0	1	DL	N	F	X	X

DL: Sets interface data length

If DL = 1, data is sent or received in 8 bit length (D7 – D0).

if DL = 0, data is sent or received in 4 bit length (D7 – D4). When 4 bit length is selected data must be sent or received twice.

N: Sets the number of lines

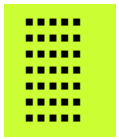
If N = 0 ; 1 line display

If N = 1 ; 2 line display

F: Sets character font

If F = 1 ; 5 x 10 dots

If F = 0 ; 5 x 7 dots



e.g. For 8 bit data length, 2 line display, 5 x 7 dots.

Function Set = 0011 1000 = 38H

LCD instructions…

2. Display ON/OFF Control

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	1	D	C	B

D : If D = 1, display is ON. If D = 0, display is OFF

C : If C = 1, cursor is displayed. If C = 0, cursor is not displayed.

B : If B = 1, Cursor blink is ON, If B = 0, Cursor blink is OFF.

e.g. For Display ON, Cursor display & blinking.

Display ON = 0000 1111 = 0FH

3. Clear Display

D7	D6	D5	D4	D3	D2	D1	D0	
0	0	0	0	0	0	0	1	= **01H**

Writes the space code “20H” into all addresses of DD RAM.

LCD instructions…

4. Entry Mode Set

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	1	I/D	S

Sets the increment / decrement and shift modes to the desired settings.

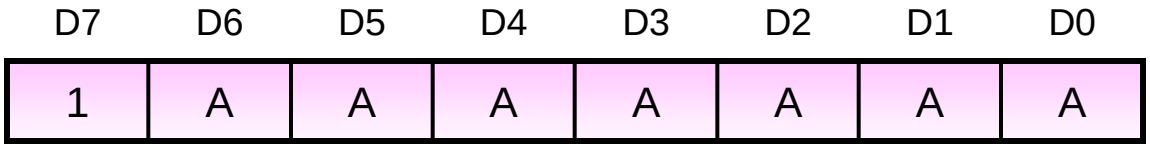
I/D : Increments (I/D = 1) or decrements (I/D = 0) the DD RAM address by 1 when a character code is written or read from the DD RAM. The cursor moves to the right when incremented by +1

S : Shift the entire display either to the right or to the left when S = 1. If I/D = 1, shift to the right. If I/D = 0, shift to the left.

e.g. DD RAM address should be incremented, No display shift.
Entry Mode Set = 0000 0110 = 06H

LCD instructions…

5. Set DD RAM Address



D7	D6	D5	D4	D3	D2	D1	D0
1	A	A	A	A	A	A	A

Sets the address counter to the DD RAM address AAAAAAA. Data is then written / read to from the DD RAM.
For 2 line display module AAAAAAA = 00H to 27H for the first line &
40H to 67H for the second line.

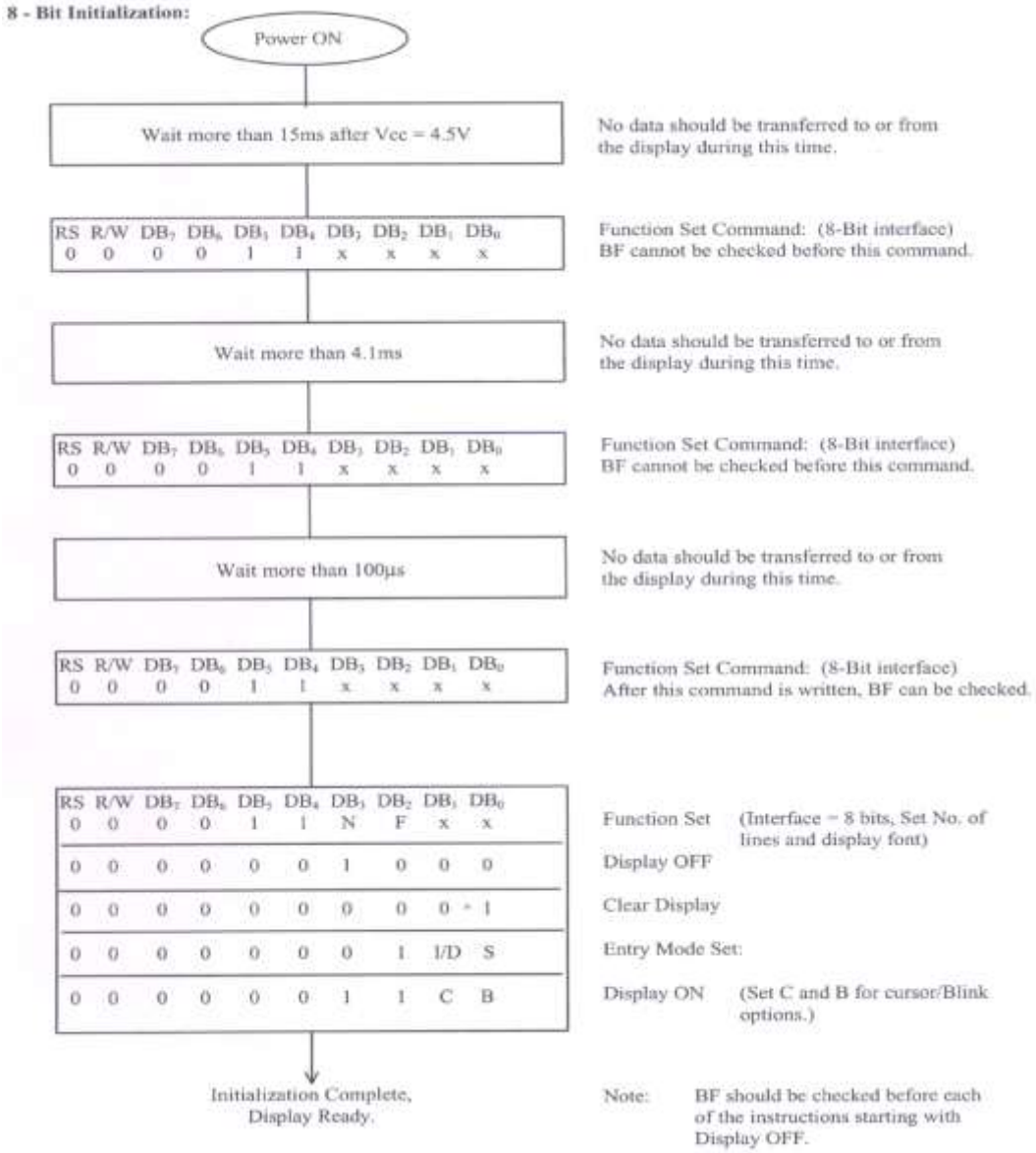
e.g. **To display characters in the first line**
Set DD RAM address = 1000 0000 = 80H

e.g. **To display characters in the second line**
Set DD RAM address = 1100 0000 = C0H

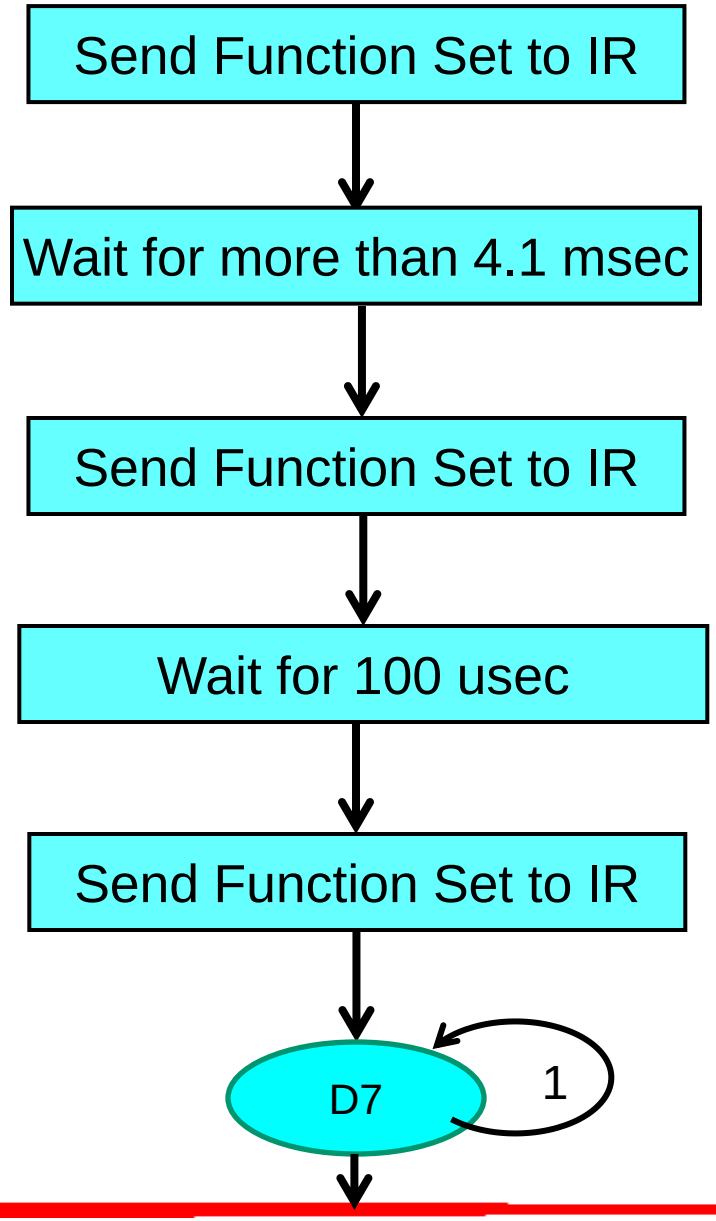
Summary of LCD instructions

Instruction	RS	R/W	D7	D6	D5	D4	D3	D2	D1	D0
Function Set	0	0	0	0	1	DL	N	F	X	X
Display ON/OFF	0	0	0	0	0	0	1	D	C	B
Clear Display	0	0	0	0	0	0	0	0	0	1
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S
Set DD RAM address	0	0	1	A	A	A	A	A	A	A
Write data to DD RAM	1	0	Write data							
Read data from DD RAM	1	1	Read data							

Initialization of LCD (8 bit)



Initialization Flowchart



; Initialize LCD
; Send Function Set
; instruction 3 times

; wait for 5 msec

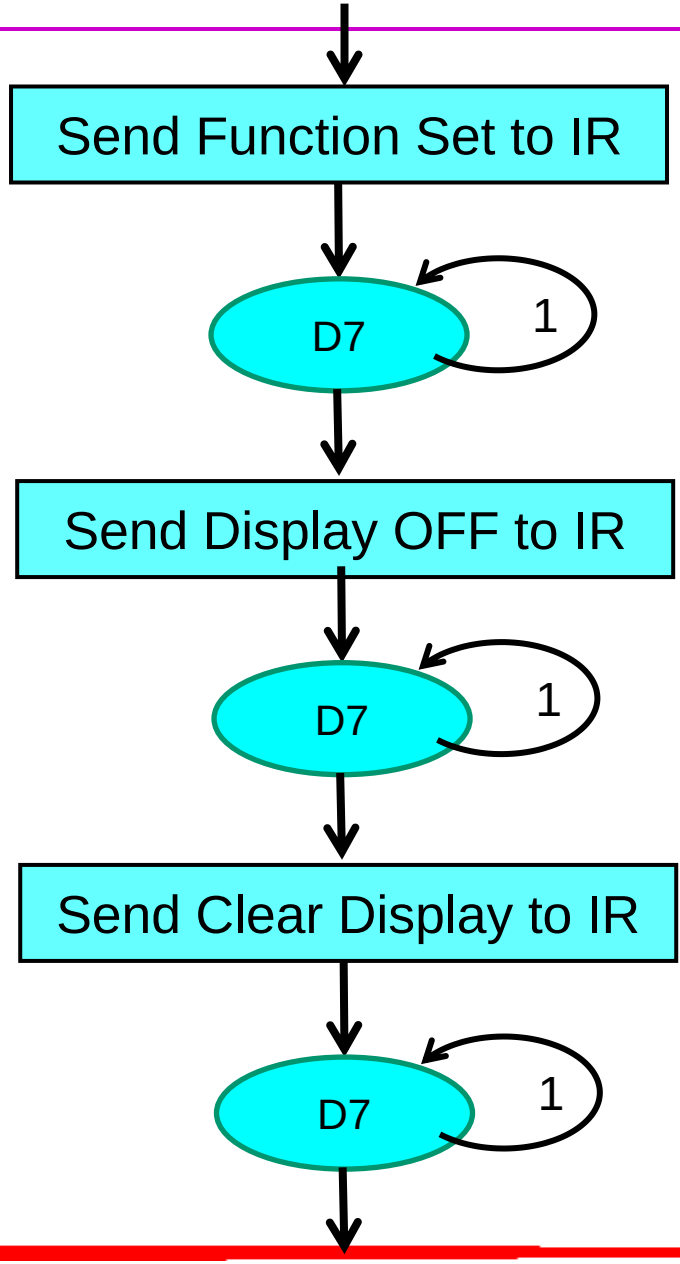
;Send Function Set instruction

; wait for 5 msec

;Send Function Set Instruction

; Read & Check Busy Flag

Initialization Flowchart …



; ;Send Function Set Instruction

; Check busy flag (D7=0)

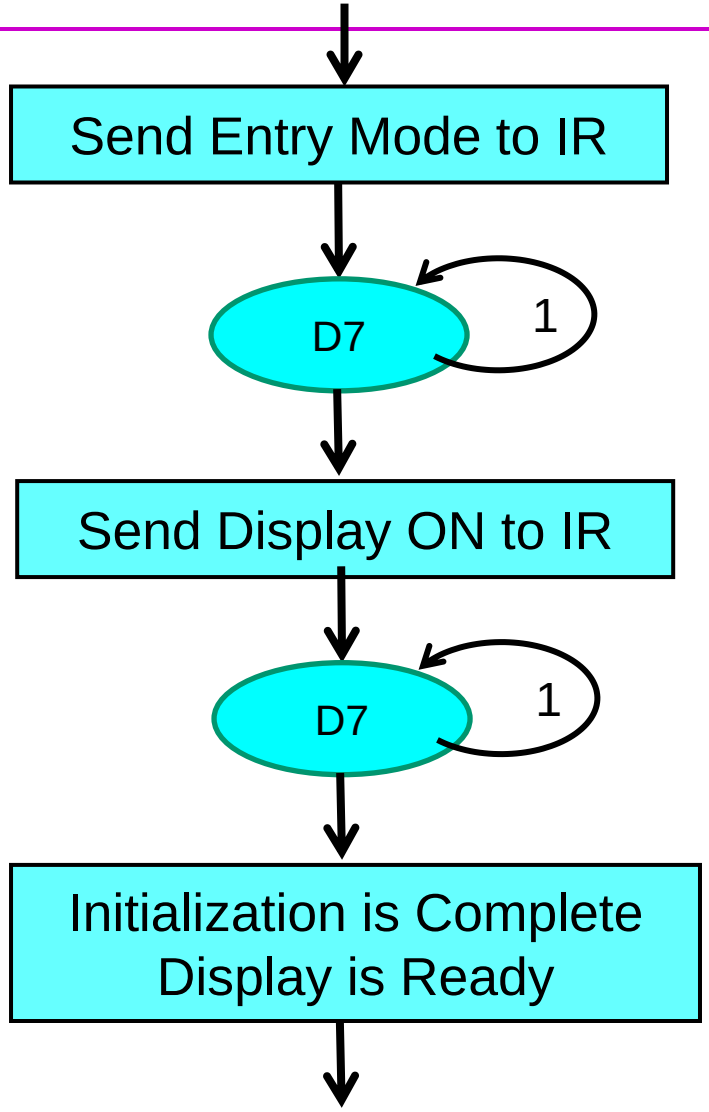
; Display OFF instruction

; Check busy flag (D7=0)

; Clear Display Instruction

; Check busy flag (D7=0)

Initialization Flowchart …



; ;Send Entry Mode Instruction

; Check busy flag (D7=0)

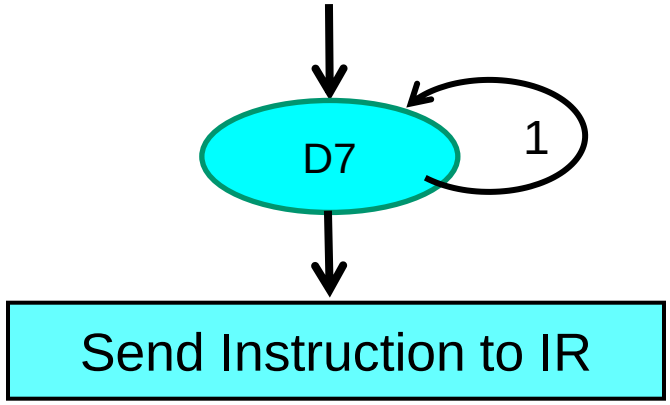
; Display ON instruction

; Check busy flag (D7=0)

; Initialization Done

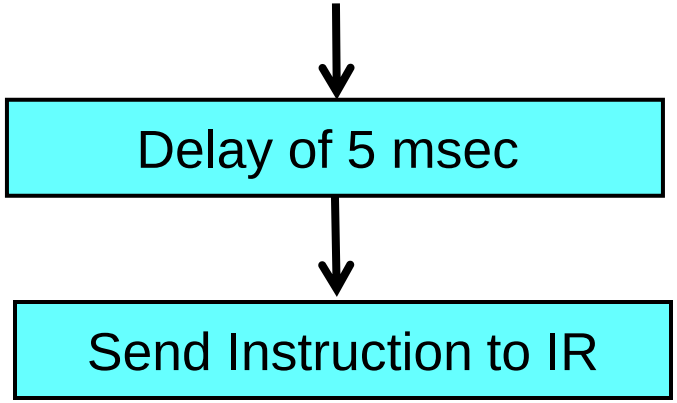
; Send Display RAM Address
; and data

Check Busy Flag OR Delay



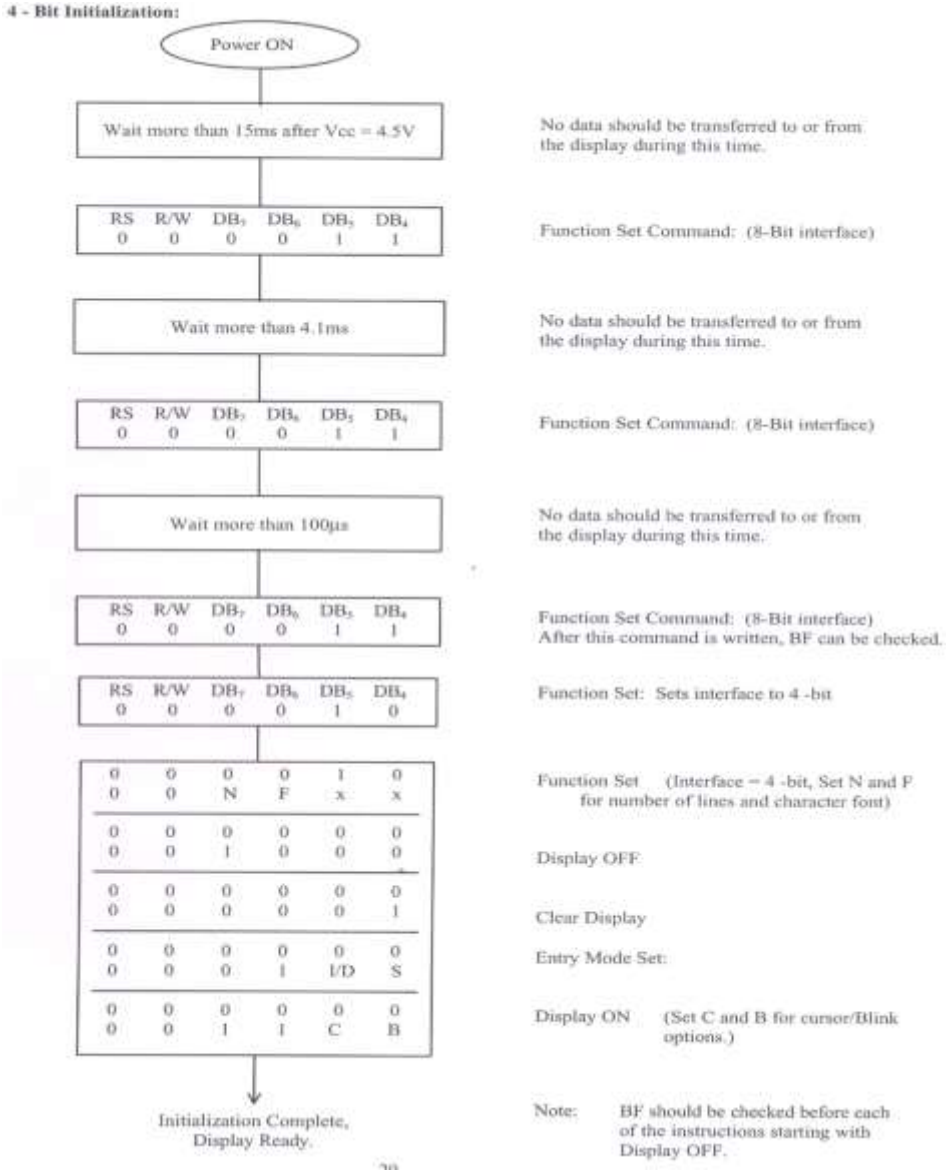
;Before sending instruction / Command you can check Busy Flag Status (D7=0)

OR



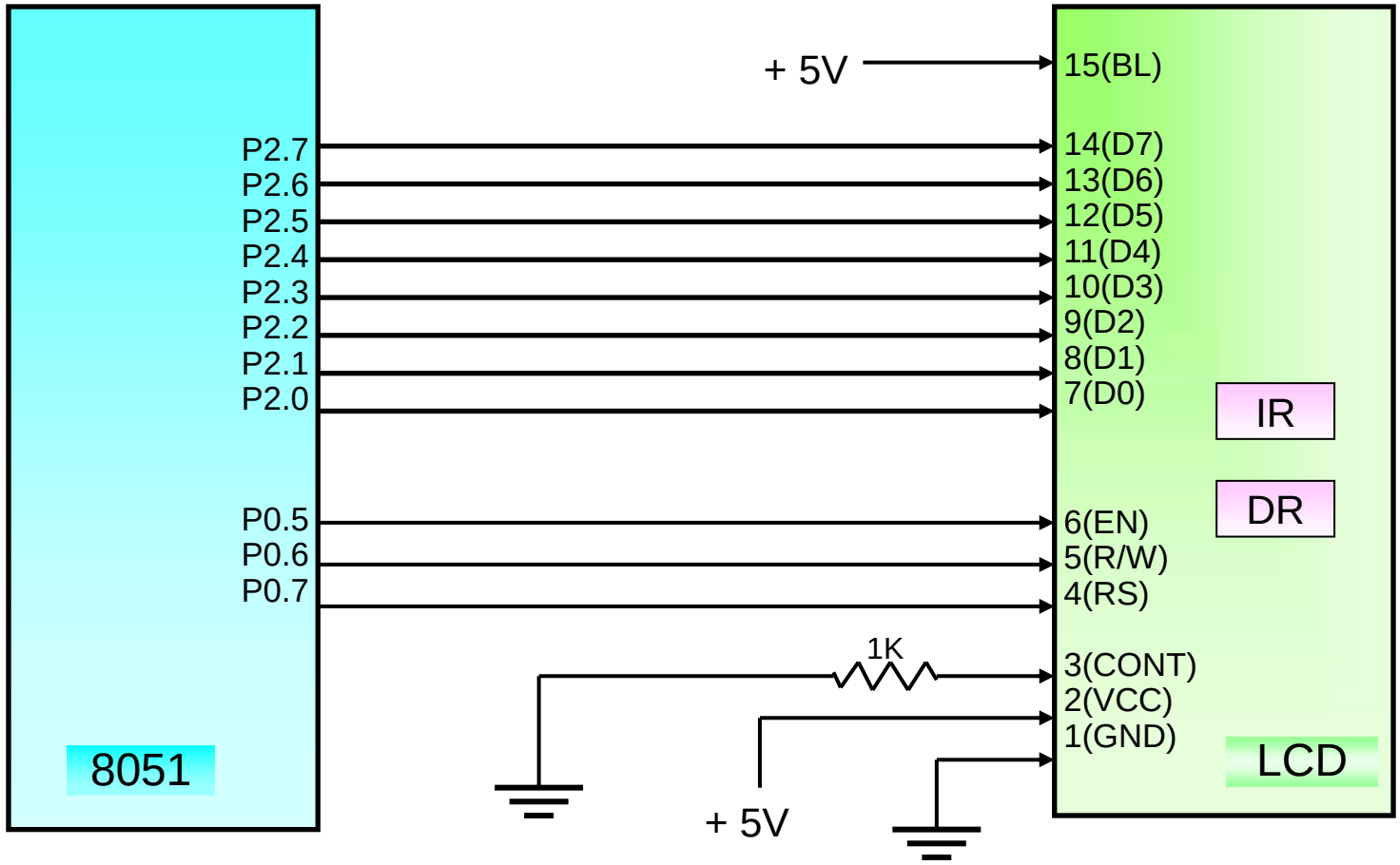
;Before sending instruction / Command you can give delay of 5 msec

Initialization of LCD (4 bit)



LCD Program #1

- Interface 2X16 LCD with 8051.Use 8 bit data length and write program to display “Y” in first line. Use P2 for data pins and P0 for control pins



LCD instructions Setting

1. Function Set

D7	D6	D5	D4	D3	D2	D1	D0
0	0	1	DL	N	F	X	X

DL = 1 ; data is sent or received in 8 bit length (D7 – D0).
N = 1 ; 2 line display
F = 0 ; 5 x 7 dots

Function Set

0	0	1	1	1	0	0	0

= 38H

LCD instructions Setting

2. Display ON/OFF Control

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	1	D	C	B

D = 1 ;display is ON
C = 1 ;cursor is displayed
B = 1 ;Cursor blink

Display ON Control

0	0	0	0	1	1	1	1

= 0FH

Display OFF Control

0	0	0	0	1	0	0	0

= 08H

3. Clear Display

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	1

= **01H**

LCD instructions Setting

4. Entry Mode Set

D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	1	I/D	S

I/D = 1 ;Increments the DD RAM address by 1 when a character code is
;written or read from the DD RAM.
;The cursor moves to the right when incremented by +1
S = 0 ; No display shift

Entry Mode Set

0	0	0	0	0	1	1	0

= 06H

LCD instructions Setting

5. DD RAM Address

DD RAM Address

80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF

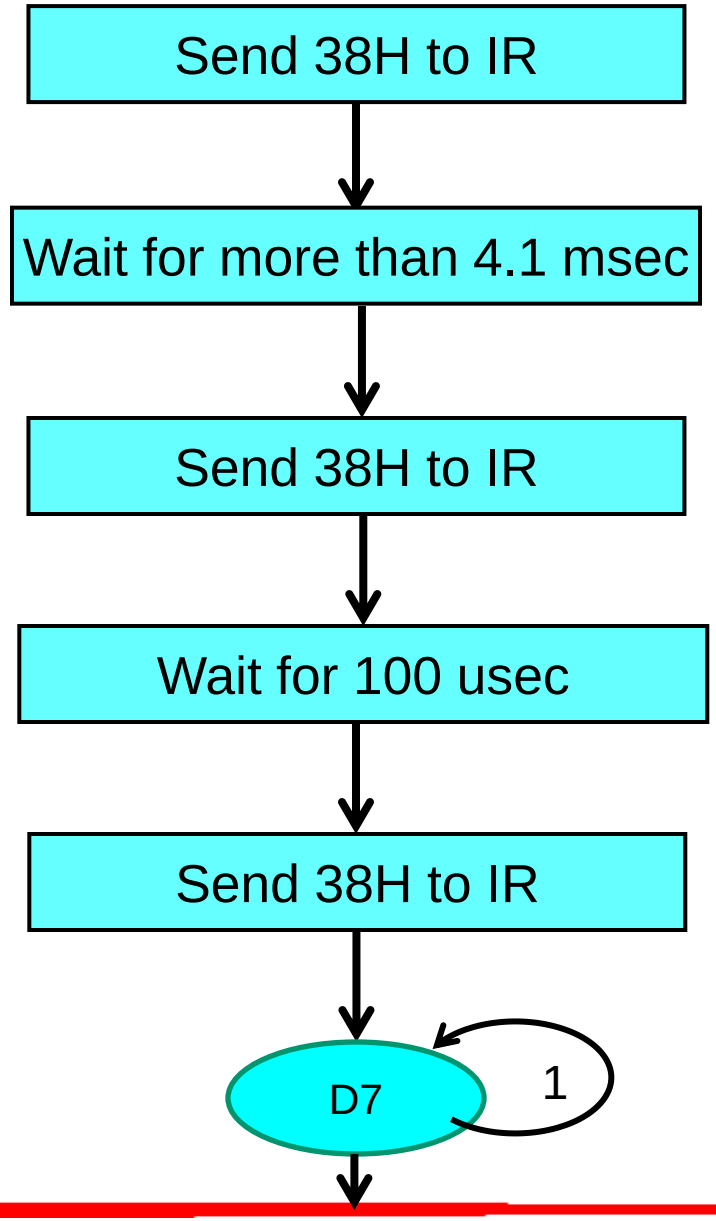
LCD display

Y															

e.g. To display characters in the first line, first position

Set DD RAM address = 1000 0000 = 80H

Flowchart



; Initialize LCD
; Send Function Set
; instruction 3 times

; wait for 5 msec

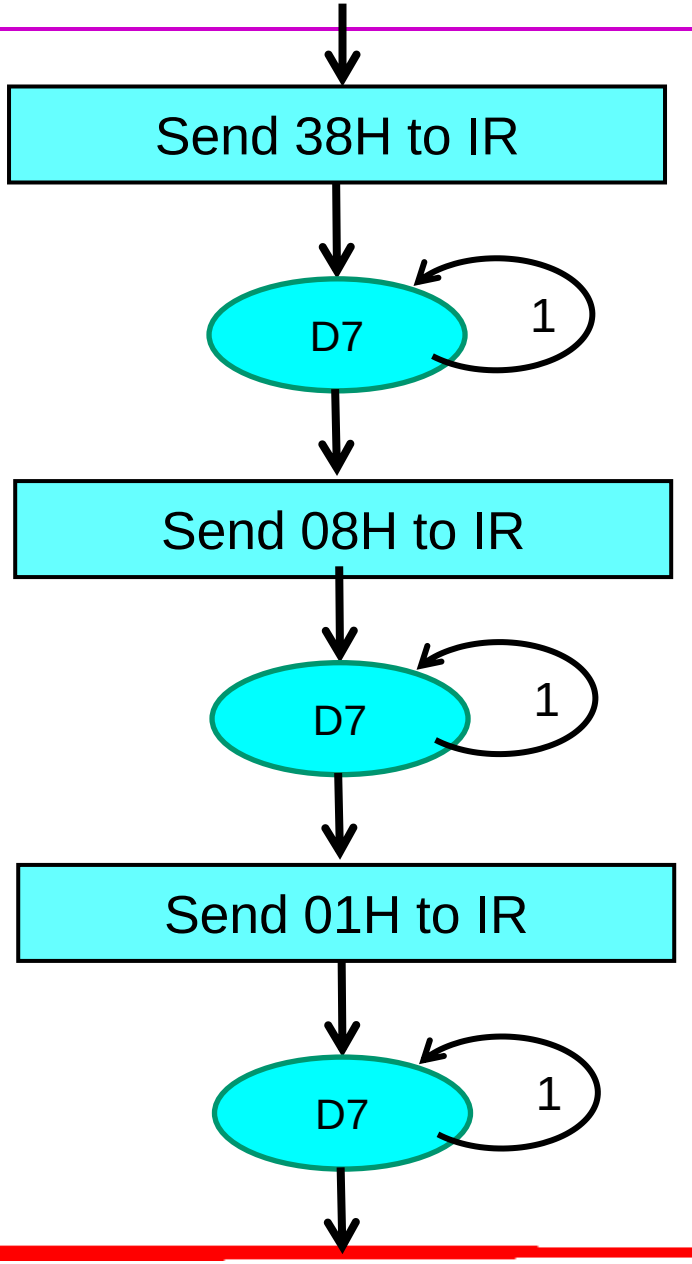
;Send Function Set instruction

; wait for 5 msec

;Send Function Set Instruction

; Read & Check Busy Flag

Flowchart …



; ;Send Function Set Instruction

; Check busy flag (D7=0)

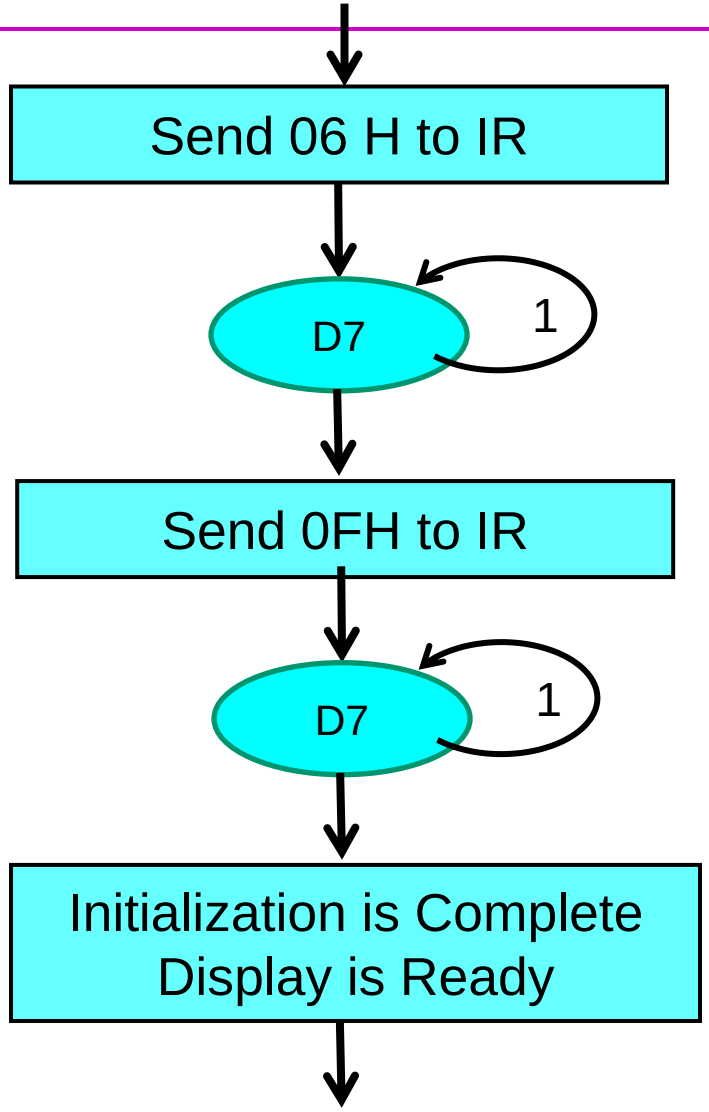
; Display OFF instruction

; Check busy flag (D7=0)

; Clear Display Instruction

; Check busy flag (D7=0)

Flowchart …



; ;Send Entry Mode Instruction

; Check busy flag (D7=0)

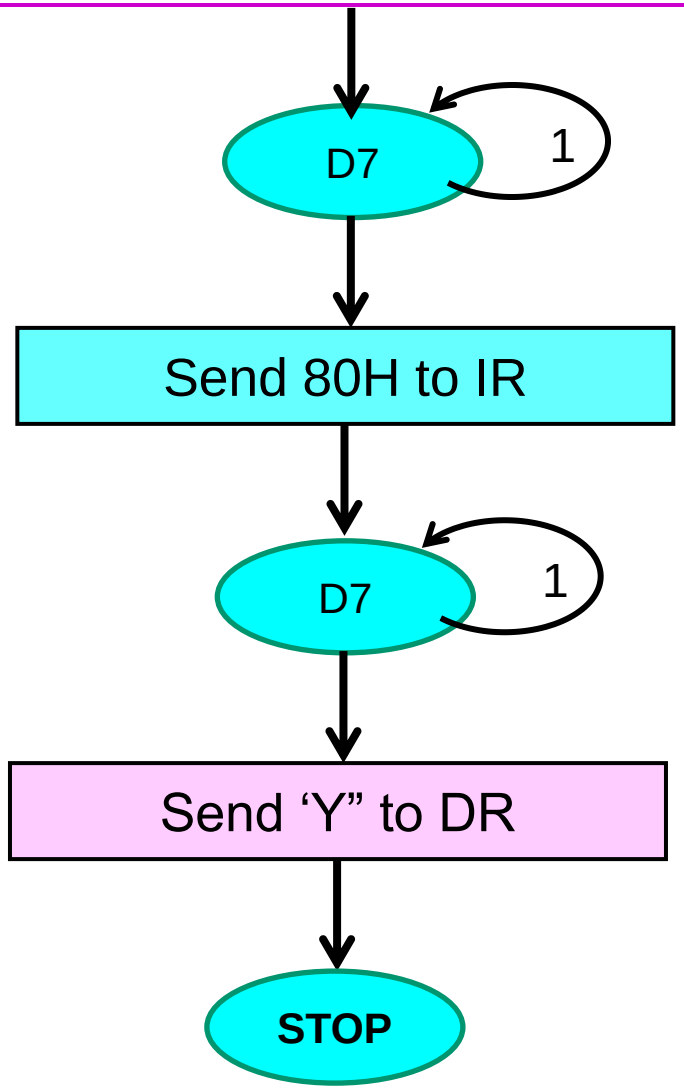
; Display ON instruction

; Check busy flag (D7=0)

; Initialization Done

; Send Display RAM Address
; and data

Flowchart…



; Check busy flag (D7=0)

; DD RAM Address
: First Line, first position

; Check busy flag (D7=0)

; ASCII CODE OF ‘Y’

Program #1

```
MOV A, #38H          ; Function Set : 0011 NFXX, N=1: 2Line, F=0: 5x7 font
ACALL COMM
ACALL delay_5ms
MOV A, #38H          ; Function Set : 0011 NFXX, N=1: 2Line, F=0: 5x7 font
ACALL COMM
ACALL delay_5ms
MOV A, #38H          ; Function Set : 0011 NFXX, N=1: 2Line, F=0: 5x7 font
ACALL COMM
ACALL delay_5ms

ACALL  CHK_BUSY      ; Check Busy Flag (D7 =0) then send
MOV A, #38H          ; Function Set : 0011 NFXX, N=1: 2Line, F=0: 5x7 font
ACALL COMM

ACALL  CHK_BUSY
MOV A, #08H          ; Display OFF (0000 1 D C B)
ACALL COMM

ACALL  CHK_BUSY
MOV A, #01H          ; Clear Display
ACALL COMM
```

Program #1 …

```
        ACALL  CHK_BUSY
        MOV A, #06H          ; Entry mode: increment (0000 01 I/D S)
        ACALL COMM

        ACALL  CHK_BUSY
        MOV A, #0FH          ; Display ON (0000 1 D C B), CURSOR BLINK
        ACALL COMM

        ACALL  CHK_BUSY
        MOV A, #80H          ; DD RAM Address (First Line, first position)
        ACALL COMM

        ACALL  CHK_BUSY
        MOV A, #'Y'          ; ASCII CODE OF 'Y'
        ACALL DISP           ; Subroutine to send data to DR
L1:     SJMP L1              ; Terminate Program
```

Sub Programs #1

```
; Subroutine to check Busy Flag
CHK_BUSY:  CLR  P0.7    ; RS= 0, Register select (Instruction Register)
           SETB  P0.6   ; RW=1, READ IR
           MOV  A, P2   ; Read Data from IR (D7=Busy Flag) (D7=1, Busy)
    BUSY:  JB  ACC.7, BUSY      ; Check Busy Flag status
           RET                  ; Busy Flag = 0


; Subroutine to send data to IR (Instruction Register)
 COMM: CLR P0.5          ; EN=0
       CLR P0.7          ; RS= 0, Register select (IR)
       CLR P0.6          ; RW=0, WRITE
       MOV P2, A         ; Send Data to IR
       SETB  P0.5        ; EN=1
       NOP
       NOP
       NOP
       CLR  P0.5         ; EN=0
       RET
```

Sub Programs #1…

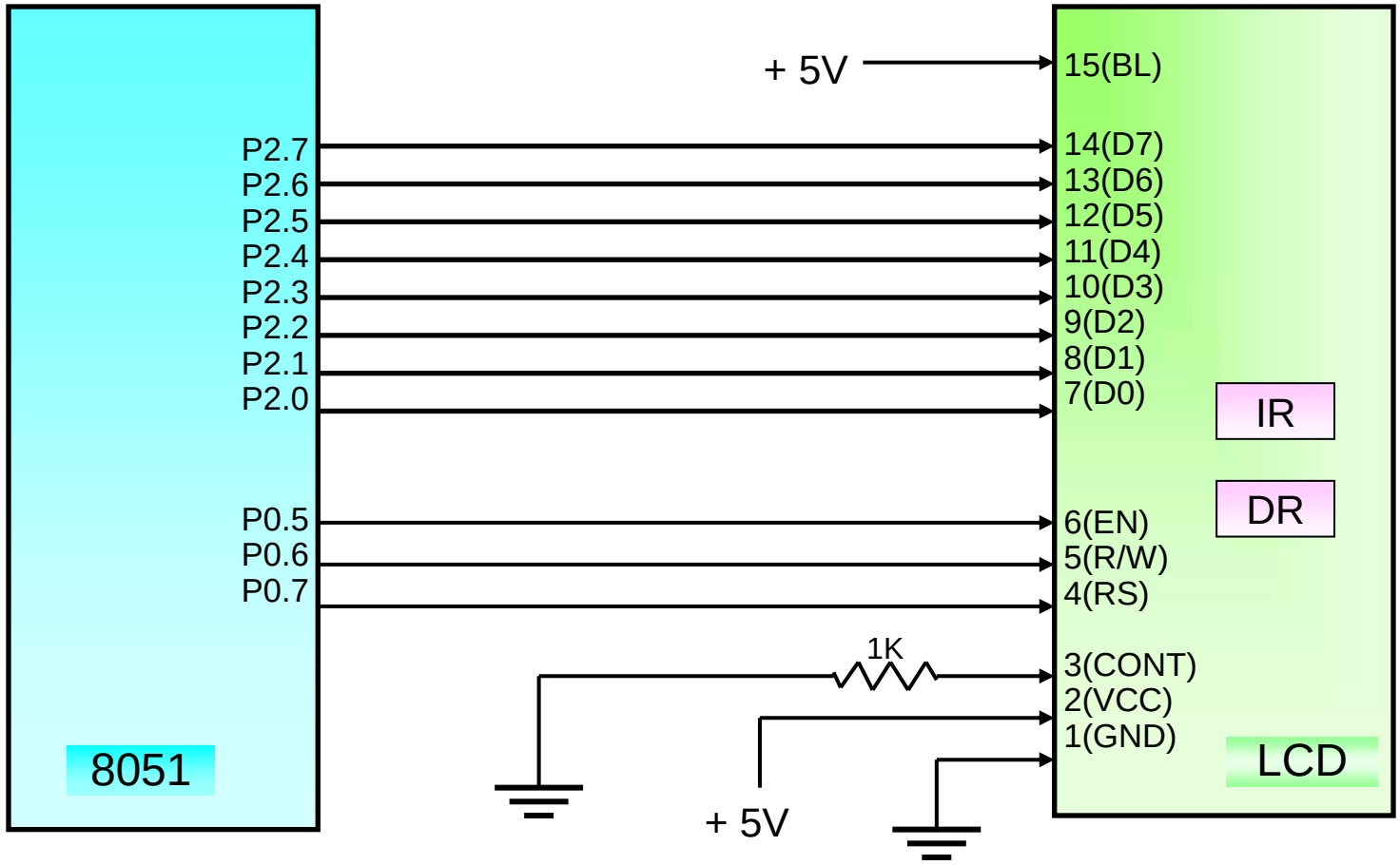
```
; Subroutine to send data to DR (Data Register)
DISP:   CLR  P0.5        ; EN=0
        SETB  P0.7       ; RS= 1, Register select (DR)
        CLR  P0.6        ; RW=0, WRITE
        MOV P2, A        ; Send Data to DR
        SETB  P0.5       ; EN=1
        NOP
        NOP
        NOP
        CLR  P0.5        ; EN=0
        RET

; Subroutine for Delay of 5 msec

delay_5ms:  MOV R0, #0AH        ; N = 10
        L2: MOV R1, #0FEH       ; N = 254
        L3: DJNZ R1, L3         ; Decrement R1& JNZ
            DJNZ R0, L2         ; Decrement R0& JNZ
            RET
```

LCD Program #2

- Interface 2X16 LCD with 8051.Use 8 bit data length and write program to display “HELLO” in first line. Use P2 for data pins and P0 for control pins



LCD instructions Setting

DD RAM Address

DD RAM Address

80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF

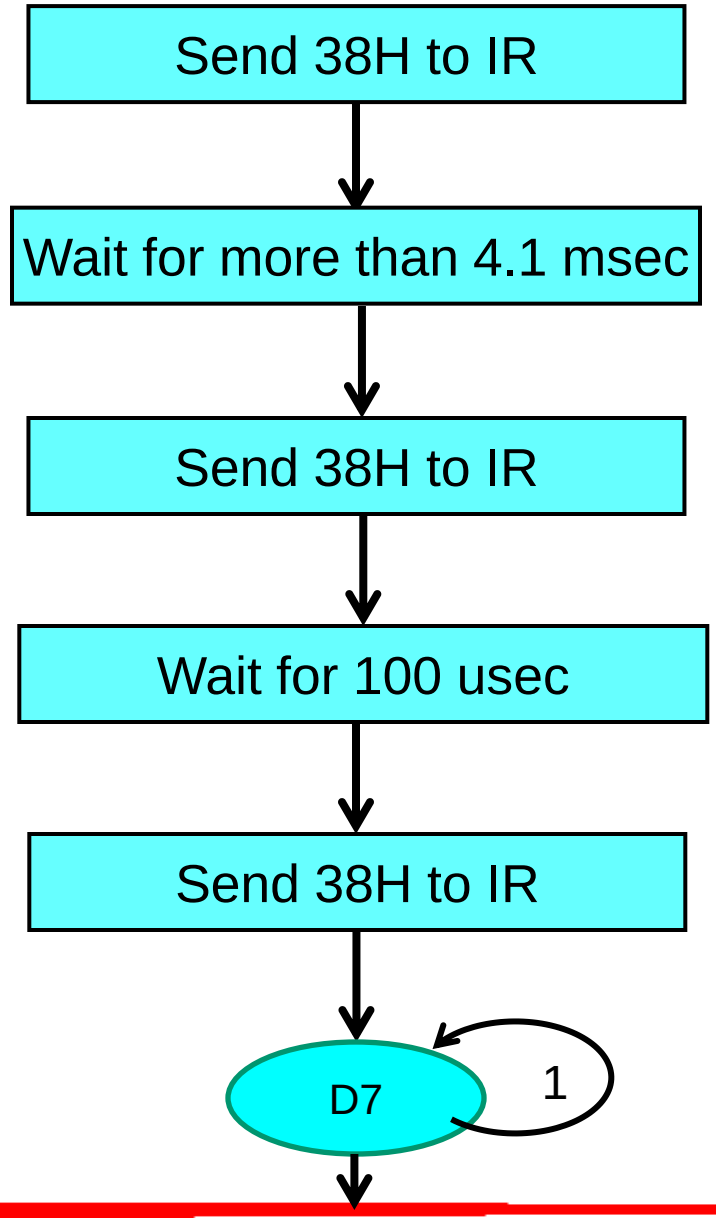
LCD display

H	E	L	L	O											

e.g. To display characters in the first line, first position

Set DD RAM address = 1000 0000 = 80H

Flowchart



; Initialize LCD
; Send Function Set
; instruction 3 times

; wait for 5 msec

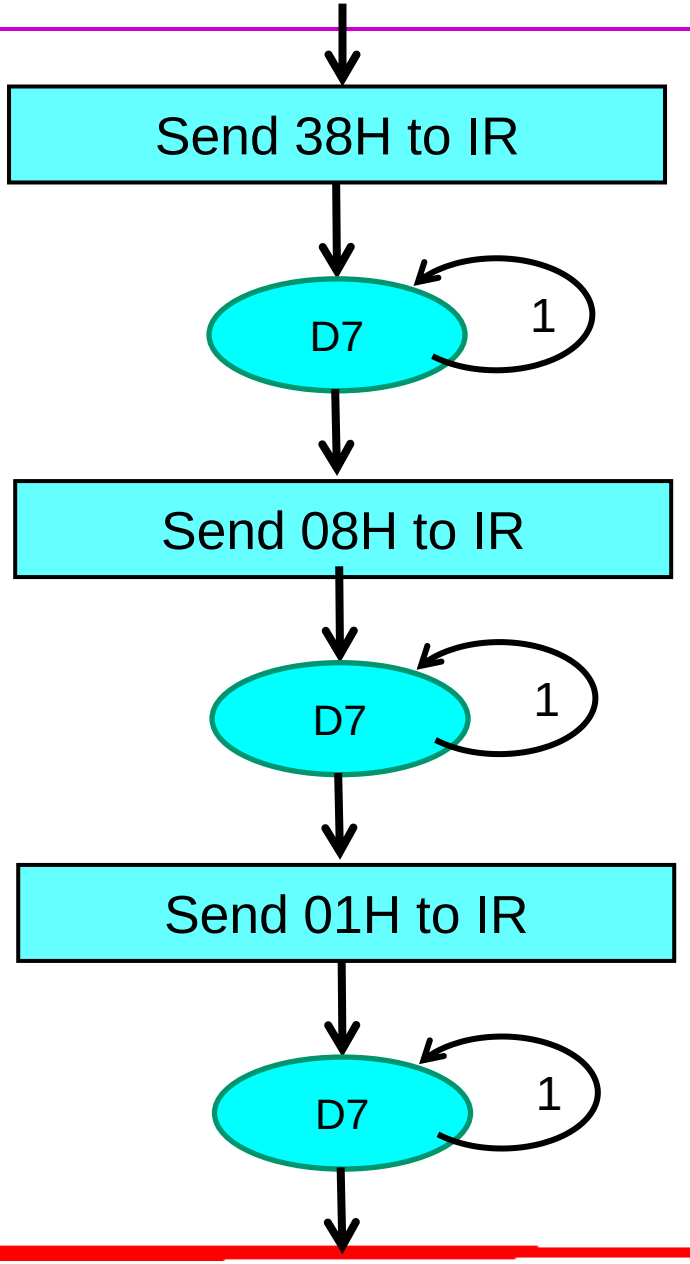
;Send Function Set instruction

; wait for 5 msec

;Send Function Set Instruction

; Read & Check Busy Flag

Flowchart …



; ;Send Function Set Instruction

; Check busy flag (D7=0)

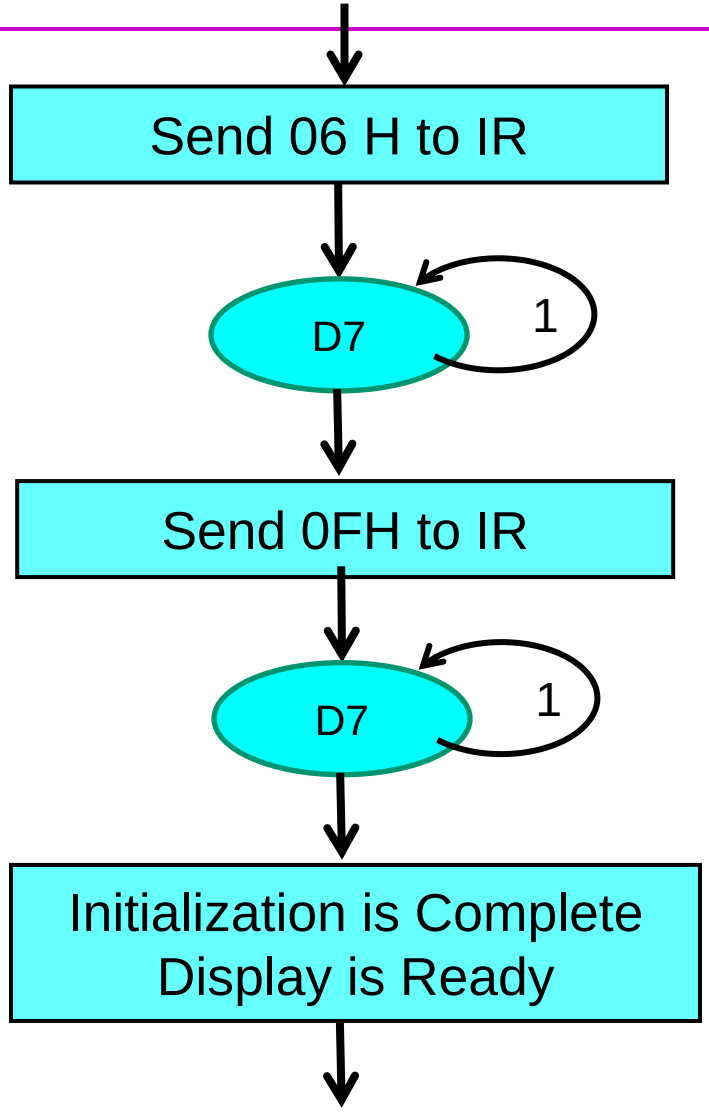
; Display OFF instruction

; Check busy flag (D7=0)

; Clear Display Instruction

; Check busy flag (D7=0)

Flowchart …



; ;Send Entry Mode Instruction

; Check busy flag (D7=0)

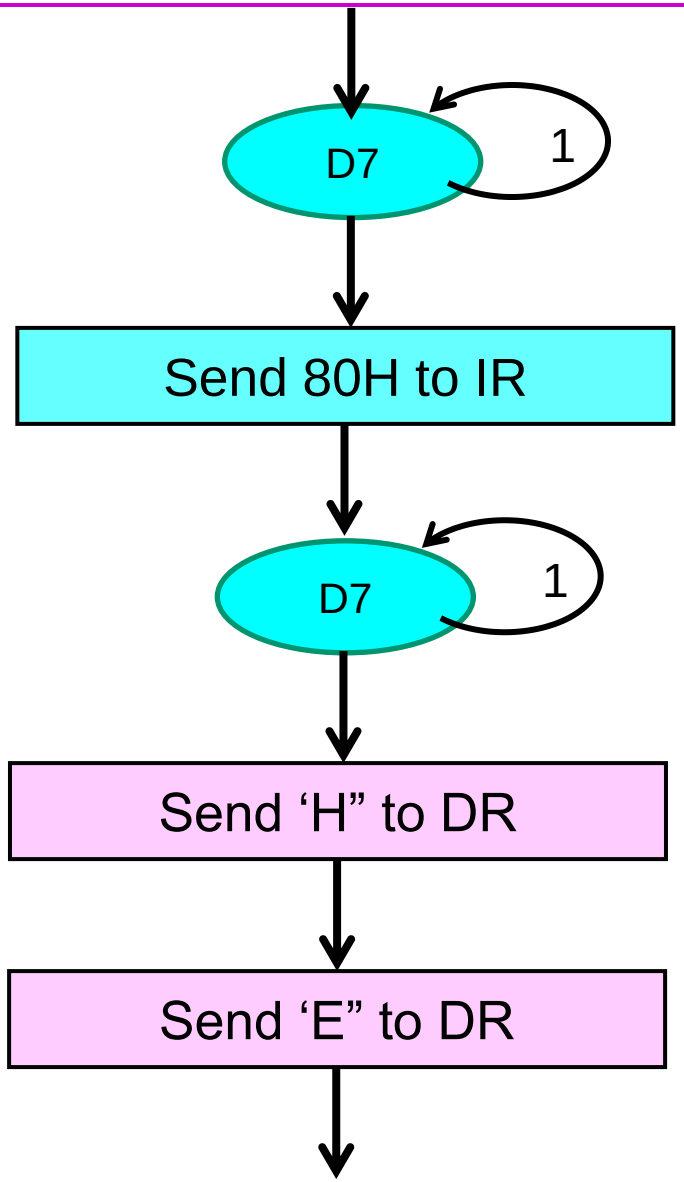
; Display ON instruction

; Check busy flag (D7=0)

; Initialization Done

; Send Display RAM Address
; and data

Flowchart…



; Check busy flag (D7=0)

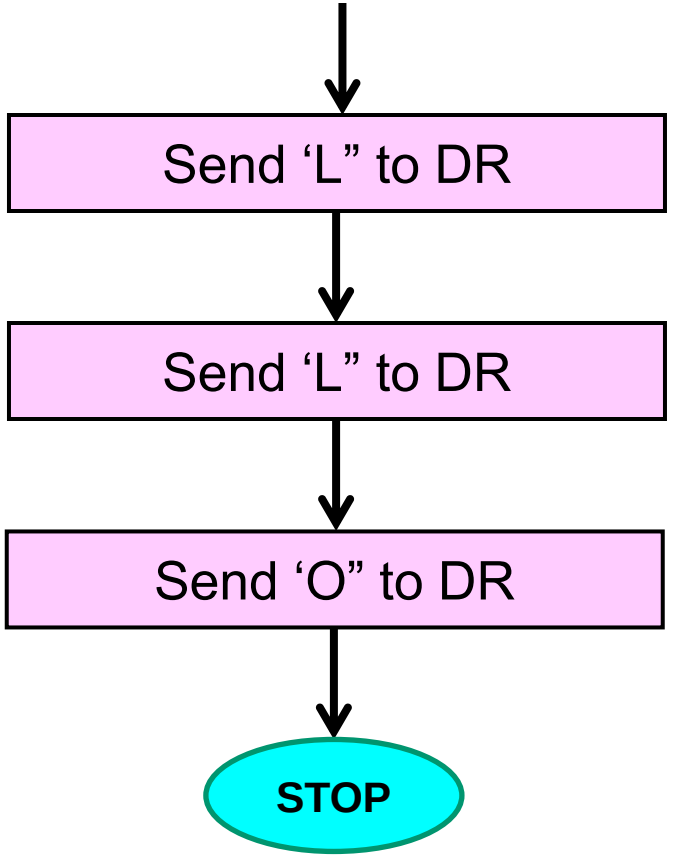
; DD RAM Address
: First Line, first position

; Check busy flag (D7=0)

; ASCII CODE OF ‘H’

; ASCII CODE OF ‘E’

Flowchart…



; ASCII CODE OF ‘L’

; ASCII CODE OF ‘L’

; ASCII CODE OF ‘O’

Program #2

```
MOV A, #38H          ; Function Set : 0011 NFXX, N=1: 2Line, F=0: 5x7 font
ACALL COMM
ACALL delay_5ms
MOV A, #38H          ; Function Set : 0011 NFXX, N=1: 2Line, F=0: 5x7 font
ACALL COMM
ACALL delay_5ms
MOV A, #38H          ; Function Set : 0011 NFXX, N=1: 2Line, F=0: 5x7 font
ACALL COMM
ACALL delay_5ms

ACALL  CHK_BUSY      ; Check Busy Flag (D7 =0) then send
MOV A, #38H          ; Function Set : 0011 NFXX, N=1: 2Line, F=0: 5x7 font
ACALL COMM

ACALL  CHK_BUSY
MOV A, #08H          ; Display OFF (0000 1 D C B)
ACALL COMM

ACALL  CHK_BUSY
MOV A, #01H          ; Clear Display
ACALL COMM
```

Program #2

```
ACALL  CHK_BUSY
MOV A, #06H              ; Entry mode: increment (0000 01 I/D S)
ACALL COMM
ACALL  CHK_BUSY
MOV A, #0FH              ; Display ON (0000 1 D C B), CURSOR BLINK
ACALL COMM
ACALL  CHK_BUSY
MOV A, #80H              ; DD RAM Address (First Line, first position)
ACALL COMM
        ACALL  CHK_BUSY
        MOV A, #'H'              ; ASCII CODE OF 'H'
        ACALL DISP               ; Subroutine to send data to DR
        MOV A, #'E'              ; ASCII CODE OF 'E'
        ACALL DISP               ; Subroutine to send data to DR
        MOV A, #'L'              ; ASCII CODE OF 'L'
        ACALL DISP               ; Subroutine to send data to DR
        MOV A, #'L'              ; ASCII CODE OF 'L'
        ACALL DISP               ; Subroutine to send data to DR
        MOV A, #'O'              ; ASCII CODE OF 'O'
        ACALL DISP               ; Subroutine to send data to DR
   L1:  SJMP L1                  ; Terminate Program
```

Sub Programs #2

```
; Subroutine to check Busy Flag
CHK_BUSY:  CLR  P0.7    ; RS= 0, Register select (Instruction Register)
           SETB  P0.6   ; RW=1, READ IR
           MOV  A, P2   ; Read Data from IR (D7=Busy Flag) (D7=1, Busy)
     BUSY: JB  ACC.7, BUSY      ; Check Busy Flag status
           RET                  ; Busy Flag = 0


; Subroutine to send data to IR (Instruction Register)
 COMM: CLR P0.5          ; EN=0
       CLR P0.7          ; RS= 0, Register select (IR)
       CLR P0.6          ; RW=0, WRITE
       MOV P2, A         ; Send Data to IR
       SETB  P0.5        ; EN=1
       NOP
       NOP
       NOP
       CLR  P0.5         ; EN=0
       RET
```

Sub Programs #2

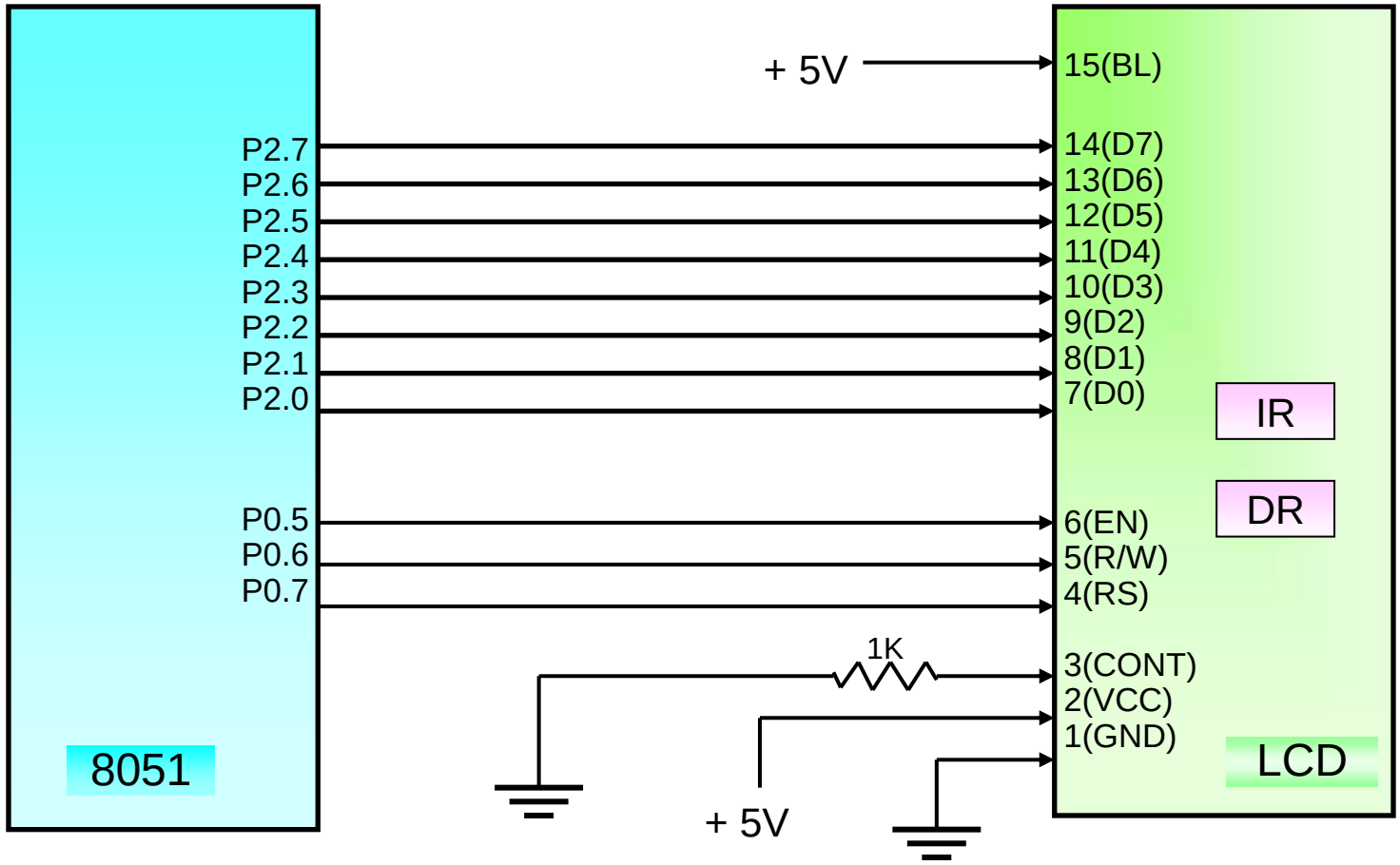
```
; Subroutine to send data to DR (Data Register)
DISP:   CLR  P0.5        ; EN=0
        SETB  P0.7       ; RS= 1, Register select (DR)
        CLR  P0.6        ; RW=0, WRITE
        MOV P2, A        ; Send Data to DR
        SETB  P0.5       ; EN=1
        NOP
        NOP
        NOP
        CLR  P0.5        ; EN=0
        RET

; Subroutine for Delay of 5 msec

delay_5ms:  MOV R0, #0AH        ; N = 10
        L2: MOV R1, #0FEH       ; N = 254
        L3: DJNZ R1, L3         ; Decrement R1& JNZ
            DJNZ R0, L2         ; Decrement R0& JNZ
            RET
```

LCD Program #3

- Interface 2X16 LCD with 8051.Use 8 bit data length and write program to display “WELCOME” in first line. Use P2 for data pins and P0 for control pins



LCD display

- DD RAM Address

DD RAM Address

80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF

LCD display

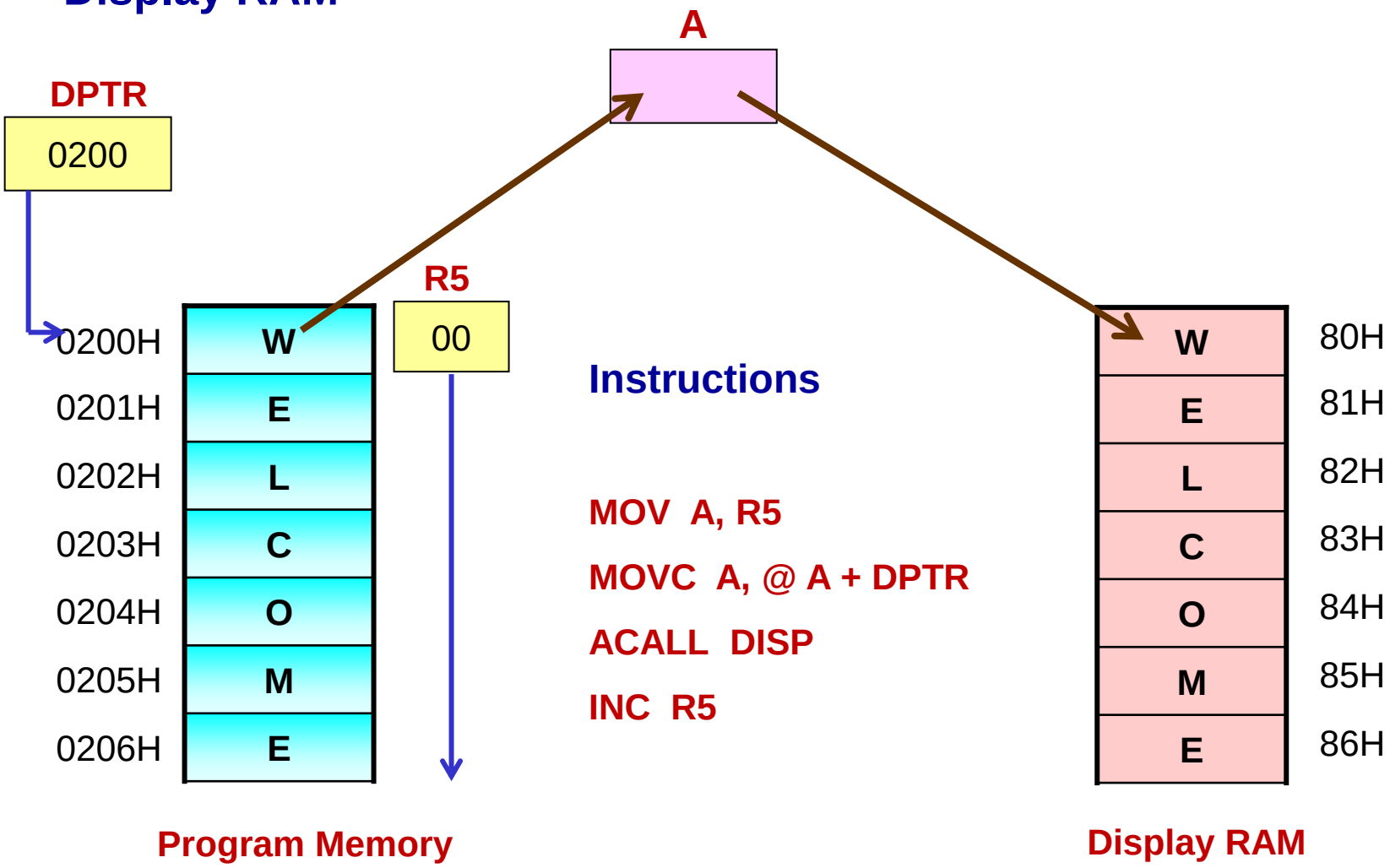
W	E	L	C	O	M	E									

e.g. To display characters in the first line, first position

Set DD RAM address = 1000 0000 = 80H

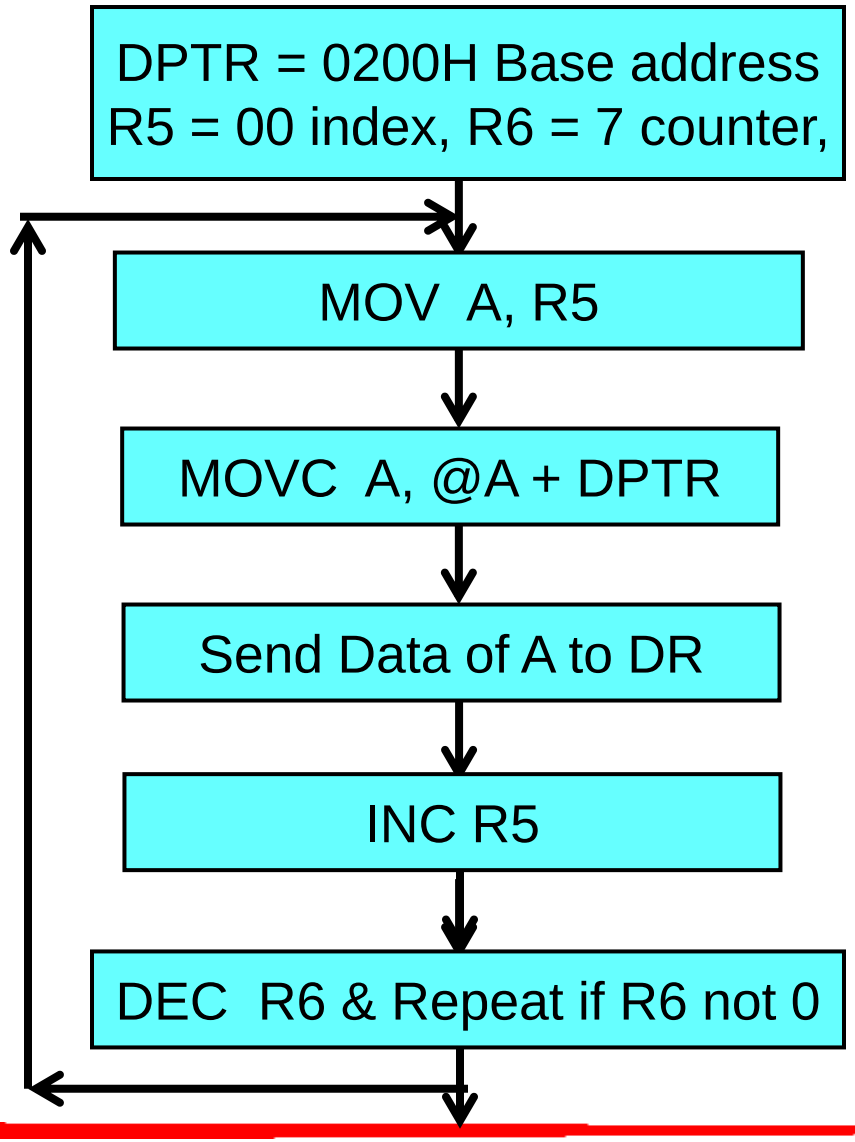
LCD Program #3

- **ASCII codes of characters are transferred from program memory to Display RAM**



LCD Program #3 …

- **Data Transfer Logic**



; Initialise Base Address,

; Index, Counter

; Transfer index to ACC

; Use MOVC to read data

; Send data to Display RAM

; Increment index for nest Char

;Decrement the Counter and

;repeat till Counter is 0

Program #3

```
MOV A, #38H          ; Function Set : 0011 NFXX, N=1: 2Line, F=0: 5x7 font
ACALL COMM
ACALL delay_5ms
MOV A, #38H          ; Function Set : 0011 NFXX, N=1: 2Line, F=0: 5x7 font
ACALL COMM
ACALL delay_5ms
MOV A, #38H          ; Function Set : 0011 NFXX, N=1: 2Line, F=0: 5x7 font
ACALL COMM
ACALL delay_5ms

ACALL  CHK_BUSY      ; Check Busy Flag (D7 =0) then send
MOV A, #38H          ; Function Set : 0011 NFXX, N=1: 2Line, F=0: 5x7 font
ACALL COMM

ACALL  CHK_BUSY
MOV A, #08H          ; Display OFF (0000 1 D C B)
ACALL COMM

ACALL  CHK_BUSY
MOV A, #01H          ; Clear Display
ACALL COMM
```

Program #3

```
ACALL  CHK_BUSY
MOV A, #06H              ; Entry mode: increment (0000 01 I/D S)
ACALL COMM
ACALL  CHK_BUSY
MOV A, #0FH              ; Display ON (0000 1 D C B), CURSOR BLINK
ACALL COMM
ACALL  CHK_BUSY
MOV A, #80H              ; DD RAM Address (First Line, first position)
ACALL COMM
        ACALL  CHK_BUSY
        MOV DPTR, #0200H        ; Base address of Look up table
        MOV R5, #00H            ; Index first element of message string
        MOV R6, #07H            ; R6 is counter for 7 characters to display
L4:     MOV A, R5               ; Index
        MOVC A, @A+DPTR         ; Get ASCII code of character in ACC
        ACALL DISP              ; Subroutine to send data to DR
        INC R5                  ; Increment Index
        DJNZ R6, L4             ; Decrement Counter
L5:     SJMP L5                 ; Terminate the Program
        ORG 0200H
        DB 'WELCOME'
```

Sub Programs #3

```
; Subroutine to check Busy Flag
CHK_BUSY:  CLR  P0.7    ; RS= 0, Register select (Instruction Register)
           SETB  P0.6   ; RW=1, READ IR
           MOV  A, P2   ; Read Data from IR (D7=Busy Flag) (D7=1, Busy)
    BUSY:  JB  ACC.7, BUSY      ; Check Busy Flag status
           RET                  ; Busy Flag = 0


; Subroutine to send data to IR (Instruction Register)
 COMM: CLR P0.5          ; EN=0
       CLR P0.7          ; RS= 0, Register select (IR)
       CLR P0.6          ; RW=0, WRITE
       MOV P2, A         ; Send Data to IR
       SETB  P0.5        ; EN=1
       NOP
       NOP
       NOP
       CLR  P0.5         ; EN=0
       RET
```

Sub Programs #3

```
; Subroutine to send data to DR (Data Register)
DISP:   CLR  P0.5        ; EN=0
        SETB  P0.7       ; RS= 1, Register select (DR)
        CLR  P0.6        ; RW=0, WRITE
        MOV P2, A        ; Send Data to DR
        SETB  P0.5       ; EN=1
        NOP
        NOP
        NOP
        CLR  P0.5        ; EN=0
        RET

; Subroutine for Delay of 5 msec

delay_5ms:  MOV R0, #0AH        ; N = 10
        L2: MOV R1, #0FEH       ; N = 254
        L3: DJNZ R1, L3         ; Decrement R1& JNZ
            DJNZ R0, L2         ; Decrement R0& JNZ
            RET
```

Thank You!!