

SISTEM MIKROPROSESOR

ANTARMUKA LCD

Mochamad Fajar Wicaksono, S.Kom., M.Kom.

- Harga
- Kemampuan untuk menampilkan angka, karakter dan gambar.
- Mudah dalam pemrograman karakter dan grafik.

LCD PIN

Pin	Nama	Fungsi
1	VSS	Ground
2	VCC	+5V
3	VEE	Tegangan kontras
4	RS	Register Select 0 = Register instruksi 1 = Register data
5	R/W	Read/ Write, untuk memilih mode tulis atau baca ; 0 = mode tulis 1 = mode baca
6	E	Enable 0 = enable (mulai menahan data ke LCD) 1 = disable
7	DB0	Data bit 0, LSB
8	DB1	Data bit 1
9	DB2	Data bit 2
10	DB3	Data bit 3
11	DB4	Data bit 4
12	DB5	Data bit 5
13	DB6	Data bit 6
14	DB7	Data bit 7, MSB
15	BPL	Back Plane Light
16	GND	Ground

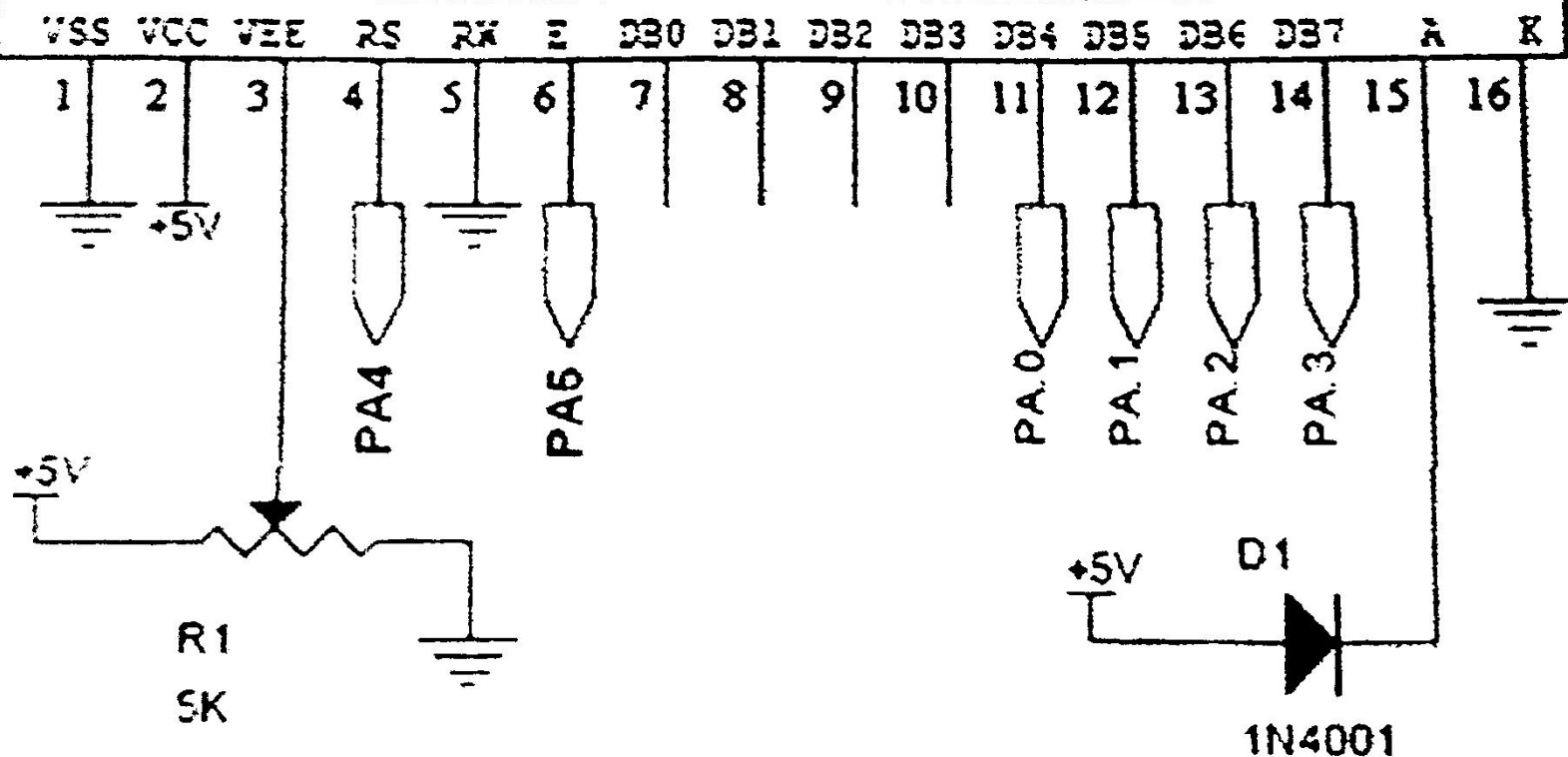
LCD PIN

Pin Enable (E)

- digunakan oleh LCD untuk menahan informasi yang masuk ke pin data
- Ketika data dikirim ke pin-pin data, sebuah pulsa *high-to-low* harus diberikan ke pin ini agar LCD dapat me-latch data yang ada di pin-pin data.
- Lebar pulsa ini minimal 450ns.

**LIQUID CRYSTAL DISPLAY
16 CHARACTER X 2 LINES**

CONTINUED ON NEXT PAGE



OPERASI PENULISAN/PEMBACAAN LCD16X2

- Modul LCD 16x2 sudah dilengkapi dengan sebuah kontroler yang memiliki 2 register 8bit.
- Register instruksi (IR) dan register data (DR).
- IR menyimpan kode instruksi seperti display clear, cursor shift dan informasi address untuk display data RAM (DDRAM) dan character generator (CGRAM)

RS	R/W	Operasi
0	0	Menulis instruksi ke IR, seperti display clear, cursor shift
0	1	Membaca busy flag (DB7) dan address counter (DB0 s/d DB7)
1	0	Menulis data ke DDRAM atau CGRAM
1	1	Membaca data dari DDRAM atau CGRAM

BUSY FLAG

- Busy flag = 1 saat kontroler sedang mengerjakan instruksi selama instruksi tersebut belum selesai dikerjakan maka kontroler tidak akan menerima instruksi apapun.
- Ketika $RS=0$ dan $R/W=1$ busy flag mengeluarkan logika 1 pada DB7. instruksi berikutnya akan siap diterima ketika busy flag = 0.

Function set

Tabel 14.4. Konfigurasi *function set*

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	1	DL	N	F	X	X

- **X** = don't care (0 atau 1)
- **DL** = mengatur lebar data
 - DL =1, lebar data 8 bit (DB7 s.d. DB0)
 - DL =0, lebar data 4 bit (DB3 s.d. DB0)
- **N** = pengaktifan baris
 - N = 0, satu baris N = 1, dua baris
- **F** = penentuan ukuran font karakter
 - F=0 , 5x7 F = 1, 5x8

ENTRY MODE SET

Tabel 9.4. Konfigurasi entry mode set

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	1	I/D	S

- **I/D** : increment/decrement alamat DDRAM dengan 1 ketika kode karakter dituliskan ke DDRAM

I/D : 0 , decrement

I/D : 1 , increment

- **S** : Geser keseluruhan tampilan ke kanan/ ke kiri

S : 1, geser ke kiri atau ke kanan tergantung I/D

S : 0, tampilan tidak bergeser

TAMPILAN KURSOR ON/OFF

Tabel 9.5. Konfigurasi kursor ON/OFF

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	1	D	C	B

- **D Mengatur tampilan**

D = 1, tampilan on D = 0, tampilan off

- **C Mengatur kursor**

C = 1, kursor ditampilkan

C = 0, kursor tidak ditampilkan

- **B Karakter ditunjukkan dengan kursor yang berkedip**

B = 1, kursor kedip B = 0, kursor tidak kedip

BERSIHKAN LAYAR

Tabel 9.6. Konfigurasi clear tampilan

[illegible]

GESER KURSOR DAN TAMPILAN

Tabel 9.7(a). Konfigurasi geser kursor dan tampilan

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	1	S/C	R/L	X	X

Tabel 9.7(b). Konfigurasi geser kursor dan tampilan

S/C	R/L	Keterangan
0	0	Geser posisi kursor ke kiri
0	1	Geser posisi kursor ke kanan
1	0	Geser seluruh tampilan ke ke kiri
1	1	Geser seluruh tampilan ke ke kiri

POSISI KURSOR

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Baris 1	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
Baris 2	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F

Tabel 9.8. Konfigurasi set alamat DDRAM

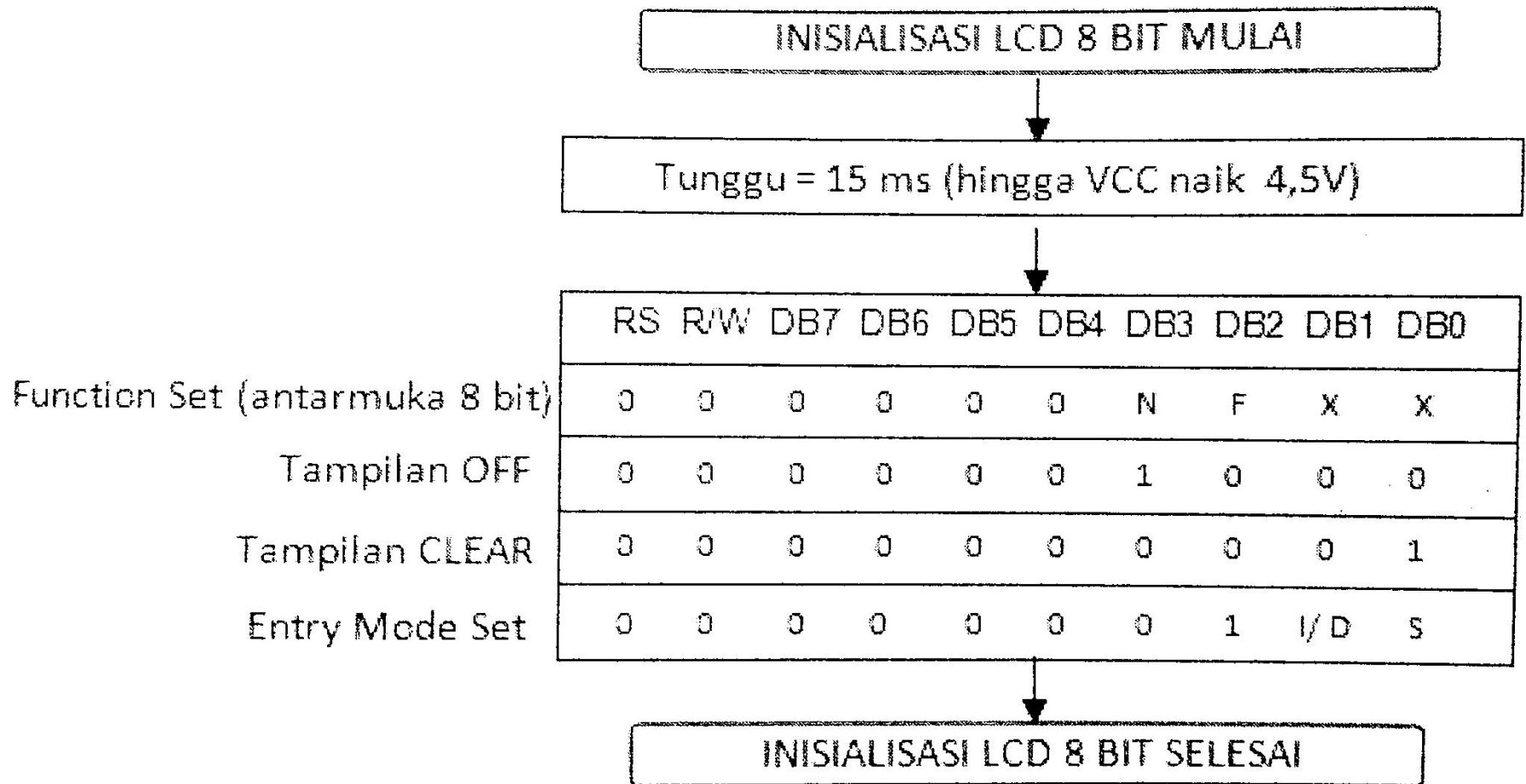
RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	1	A	A	A	A	A	A	A

Tabel 14.9. Rangkuman format perintah LCD HD44780

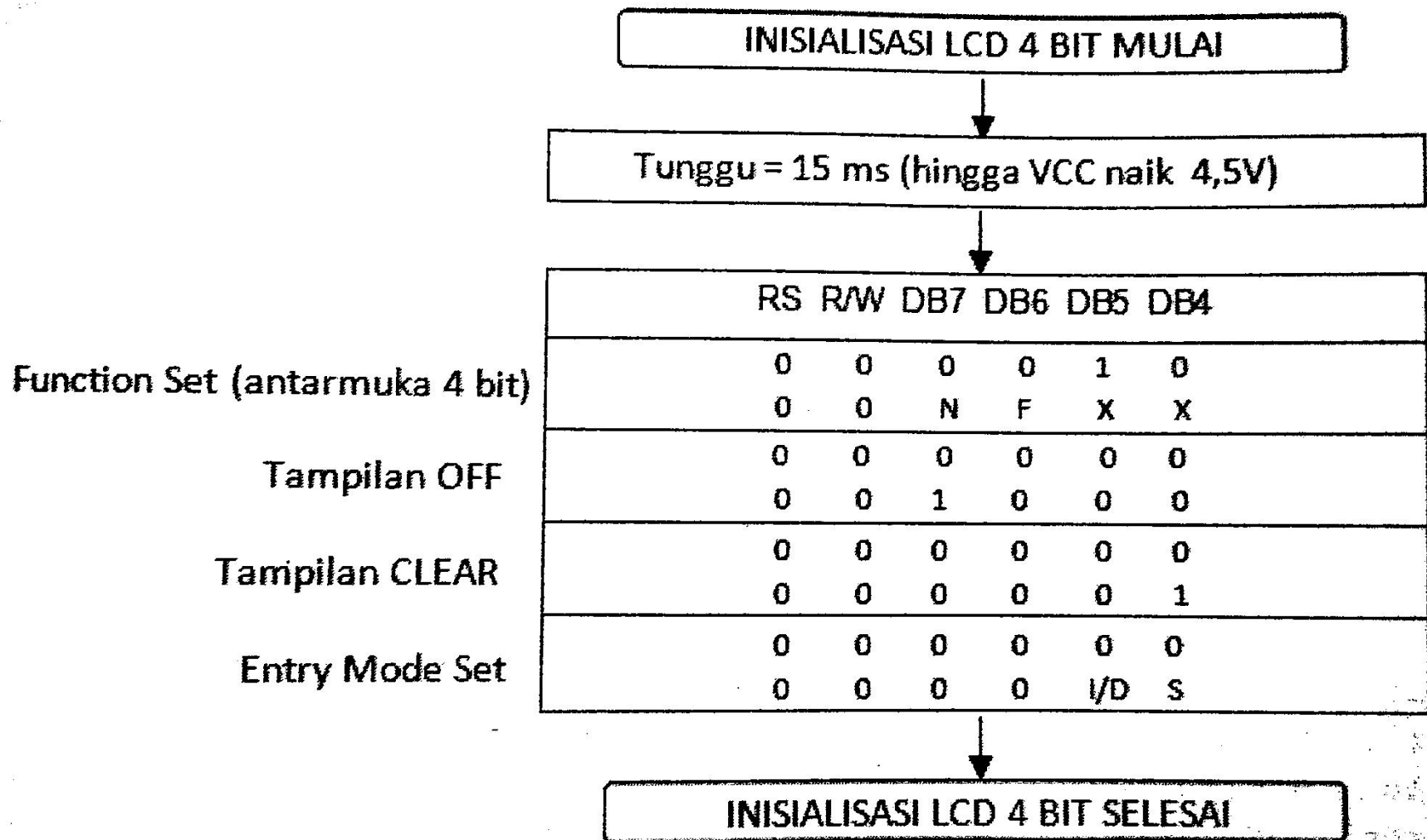
Operasi	D 7	D 6	D 5	D 4	D3	D2	D1	D 0	Hex
Hapus tampilan	0	0	0	0	0	0	0	1	01
Kursor ke awal <i>text</i>	0	0	0	0	0	0	1	Ø	02 – 03
<i>Mode entry</i>	0	0	0	0	0	1	I/D	S	04 – 07
<i>Display on/off</i> kursor	0	0	0	0	1	D	U	B	08 – 0F
Mode geser	0	0	0	1	D/C	R/L	Ø	Ø	10 – 1F
<i>Function set</i>	0	0	1	8/4	2/1	8/7	Ø	Ø	20 – 3F
Set alamat CGRAM	0	1	A	A	A	A	A	A	40 – FF
Set alamat tampilan	1	A	A	A	A	A	A	A	80 – FF

Keterangan:

- Ø = *don't care* (0 atau 1)
 D 1 = tampilan ON; 0 = OFF *
 S 1 = tampilan geser ON; 0 = OFF *
 U 1 = kursor underline ON; 0 = OFF *
 B 1 = kursor berkedip ON; 0 = OFF *
 I/D 1 = *increment* *; 0 = *decrement*
 D/C 1 = tampilan geser ; 0 = kursor bergerak
 R/L 1 = kanan ; 0 = kiri
 8/4 1 = bus data 8-bit *; 0 = bus data 4 bit
 2/1 1 = mode dua baris; 0 = mode satu baris *
 8/7 1 = jumlah dot 5x8 *; 0 = jumlah dot 5x7 *
 Tanda *) menunjukkan keadaan *default*

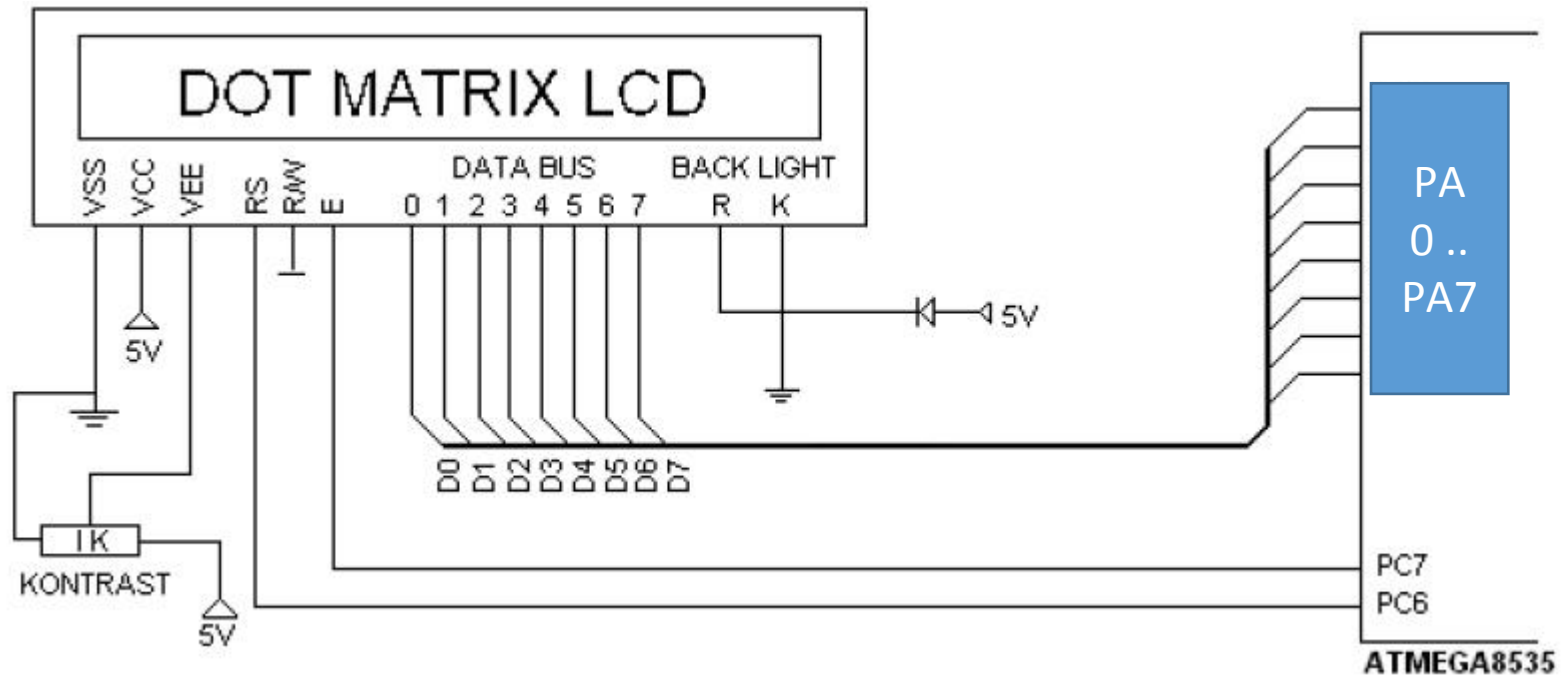


Gambar 14.3. Diagram alir inisialisasi LCD 16x2 mode antarmuka 8-bit



Gambar 14.4. Diagram alir inisialisasi LCD 16x2 mode antarmuka 4-bit

CONTOH



```

.include "m8535def.inc"                ; Definisi alamat register ATmega8535

.equ   LCD_DATA      = PORTA
.equ   LCD_CONTROL    = PORTB
.equ   LCD_EN        = 0
.equ   LCD_RS        = 1

.org 0x0000                            ; Mulai pengisian program pd alamat 0x0000
    rjmp    reset                      ; Lompat ke reset

reset:
    ; Pengaturan Stack Pointer
    ldi     r16, low(RAMEND)           ; Isikan r16 dgn nilai dari RAMEND(low)
    out     SPL, r16                  ; Isikan register SPL dgn nilai dari r16
    ldi     r16, high(RAMEND)         ; Isikan r16 dgn nilai dari RAMEND(high)
    out     SPH, r16                  ; Isikan register SPH dgn nilai dari r16

init:
    ; Pengaturan arah PORTA
    ldi     r16, 0xff                 ; Isikan r16 dgn nilai 1 semua (output)
    out     DDRA, r16                 ; Isikan register DDRA dgn nilai dari r16

    ; Pengaturan arah PORTC
    out     DDRC, r16                 ; Isikan register DDRC dgn nilai dari r16

    out     PORTA, r16
    out     PORTC, r16

```

```

main:
    rcall    LCD_Init
    ; Isikan register Z dengan alamat Tulisan
    ldi      zh, high(Tulisan<<1)
    ldi      zl, low(Tulisan<<1)

loop:
    lpm      r16, z+          ; Isikan r16 dgn nilai yg ditunjuk oleh Z
    cpi      r16, 0          ; Cek untuk null-terminator
    breq     fin             ; Jika iya selesai

    rcall    LCD_WriteData    ; Tulis data ke LCD
    rjmp     loop

fin:
    rjmp     PC              ; Loop ditempat

```

LCD_Init:

```
    ldi    r16, 0x38          ; 00110000, 8 bit, 1 baris
    rcall  LCD_WriteInst      ; Tulis instruksi ke LCD

    ldi    r16, 0x0c          ; 00001100, nyalakan LCD
    rcall  LCD_WriteInst      ; Tulis instruksi ke LCD

    ldi    r16, 0x06          ; 00000110, set kursor
    rcall  LCD_WriteInst      ; Tulis instruksi ke LCD

    ldi    r16, 0x01          ; 00000001, hapus layar
    rcall  LCD_WriteInst      ; Tulis instruksi ke LCD
    rcall  delay
    ret
```

LCD_WriteInst:

```
    cbi    LCD_CONTROL, LCD_RS ; RS=0 (Instruksi)
    out    LCD_DATA, r16        ; Siapkan data
    cbi    LCD_CONTROL, LCD_EN ; Turunkan EN
    rcall  delay                ; Panggil rutin delay
    sbi    LCD_CONTROL, LCD_EN ; Naikkan kembali EN
    ret                          ; Selesai rutin
```

LCD_WriteData:

```
    sbi    LCD_CONTROL, LCD_RS ; RS=1 (Data)
    out    LCD_DATA, r16        ; Siapkan data
    cbi    LCD_CONTROL, LCD_EN ; Turunkan EN
    rcall  delay                ; Panggil rutin delay
    sbi    LCD_CONTROL, LCD_EN ; Naikkan kembali EN
    ret                          ; Selesai rutin
```

delay:

Tulisan:

LATIHAN