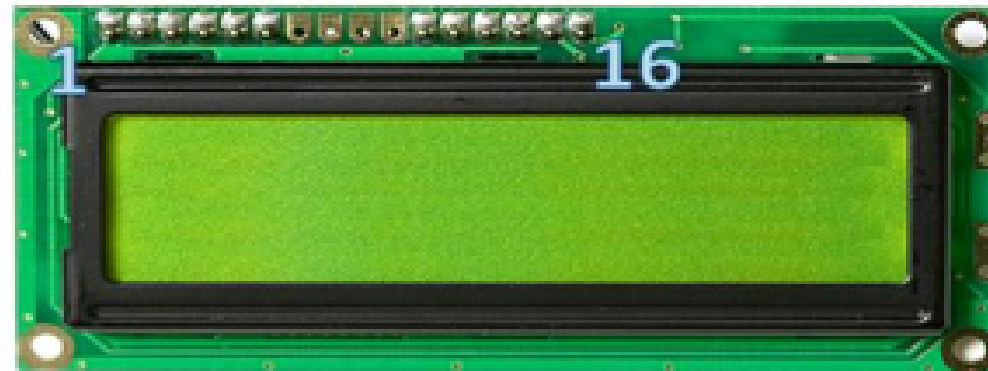


**MIKROKONTROLER ATMEGA
BERBASIS
CODEVISION AVR
(LCD 16X2 KARAKTER)**

LCD 16x2 Karakter HD44780

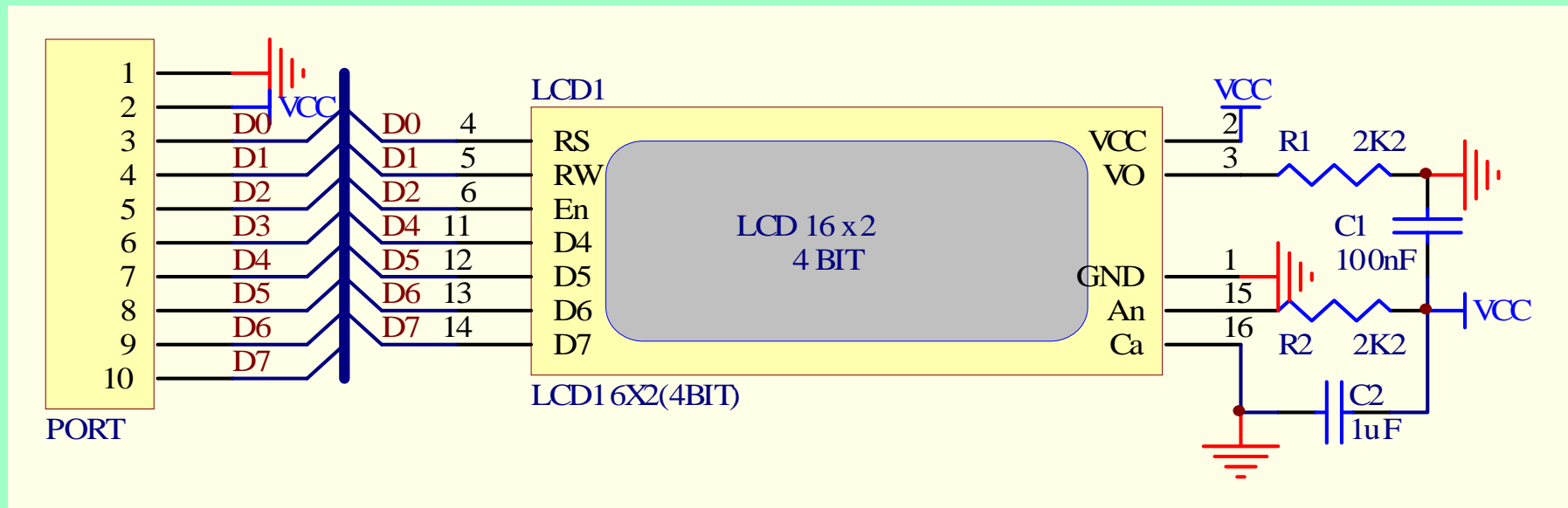
HD44780 - 16x2 LCD
Pin Description



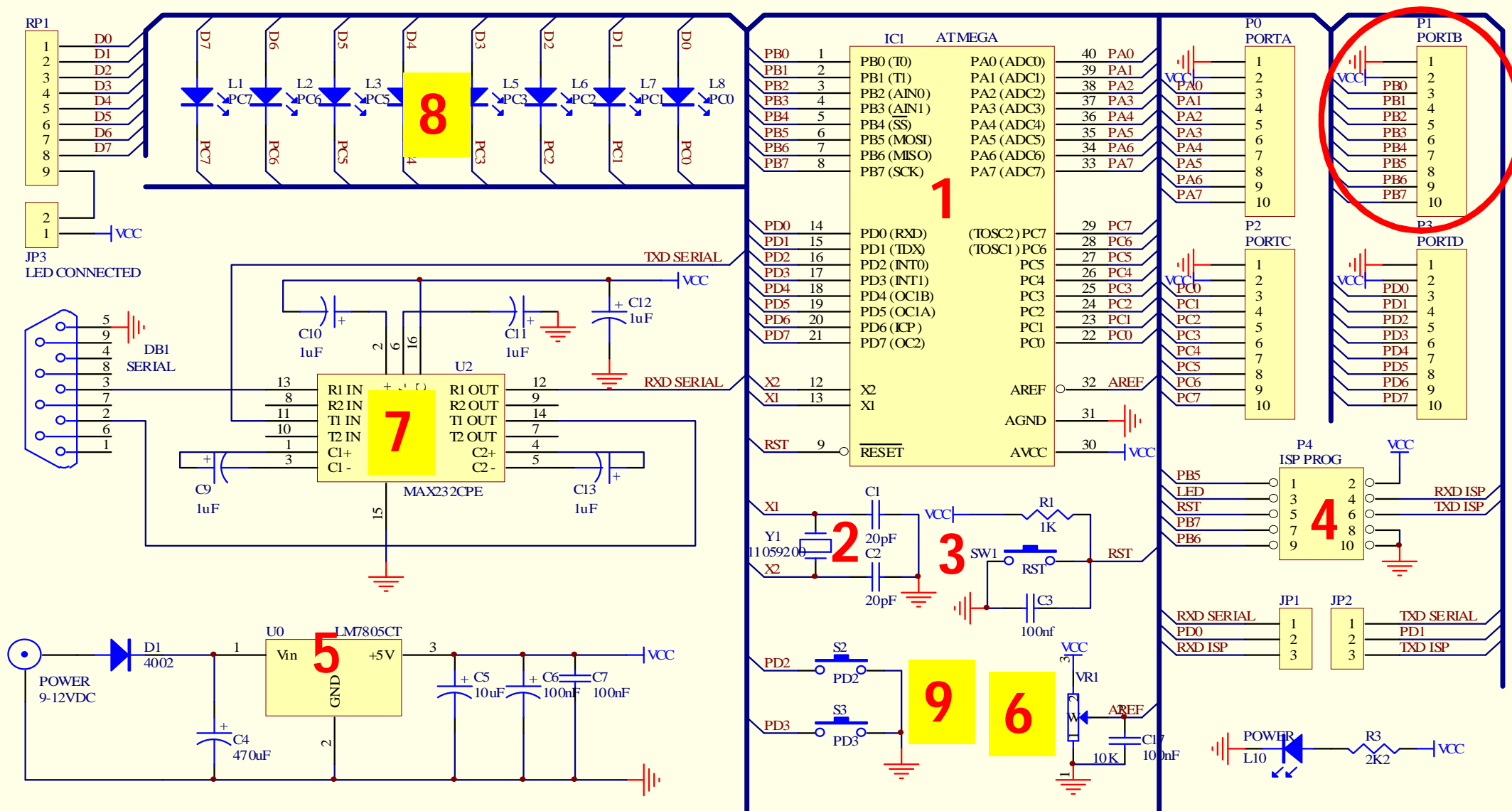
PIN	Description	
GND	GND connection	
VCC	$V_{CC} = 3.3 - 5\text{ V}$	
VEE	LCD Character Dimmer	
RS	Control Signals	Selects between writing a command or data
R / W		Read or Write operation. This pin is usually wired to GND
E		Falling edge will trigger an operation
D0	Data Lines	Least significant bites, usually not connected since LCD is used in 4-bit operation.
D1		
D2		
D3		
D4		Most significant bites. Data and commands are written to these pins 1 nibble at a time, starting with the most significant nibble (MSN).
D5		
D6		
D7		
An(+)	Power supply for the back light, V_{CC}	
Ca(-)	GND connection for the backlight, GND	

Rangkaian Antarmuka 4-Bit LCD

- PORT = Koneksi ke mikrokontroler

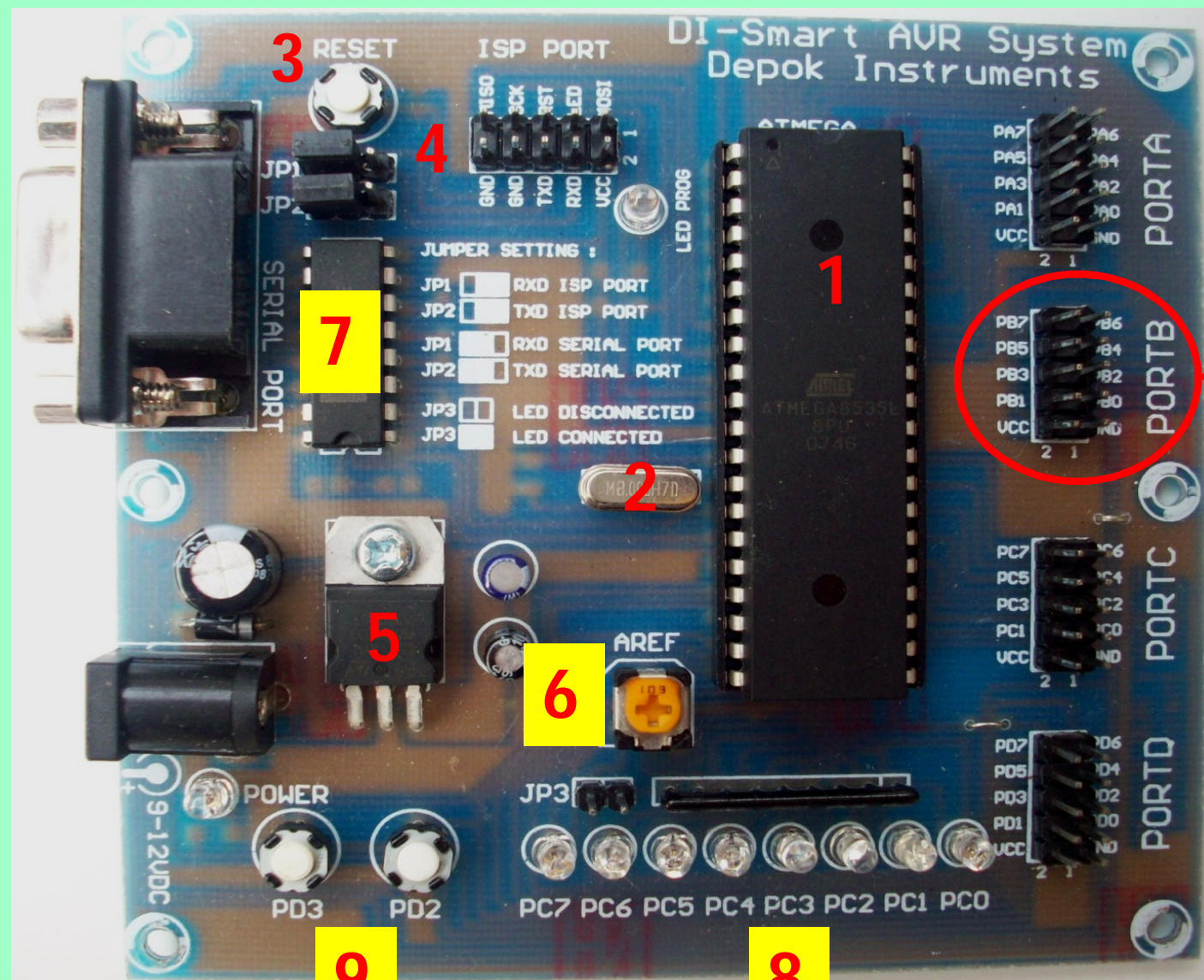


DI-Smart AVR.16 System (Sistem Plus μ C. ATmega16A)



DI-Smart AVR.16 System

1. ATmega16A
2. Crystal
3. Reset
4. ISP
5. Power
6. Vref ADC
7. RS232
8. LED Array
9. Push Button

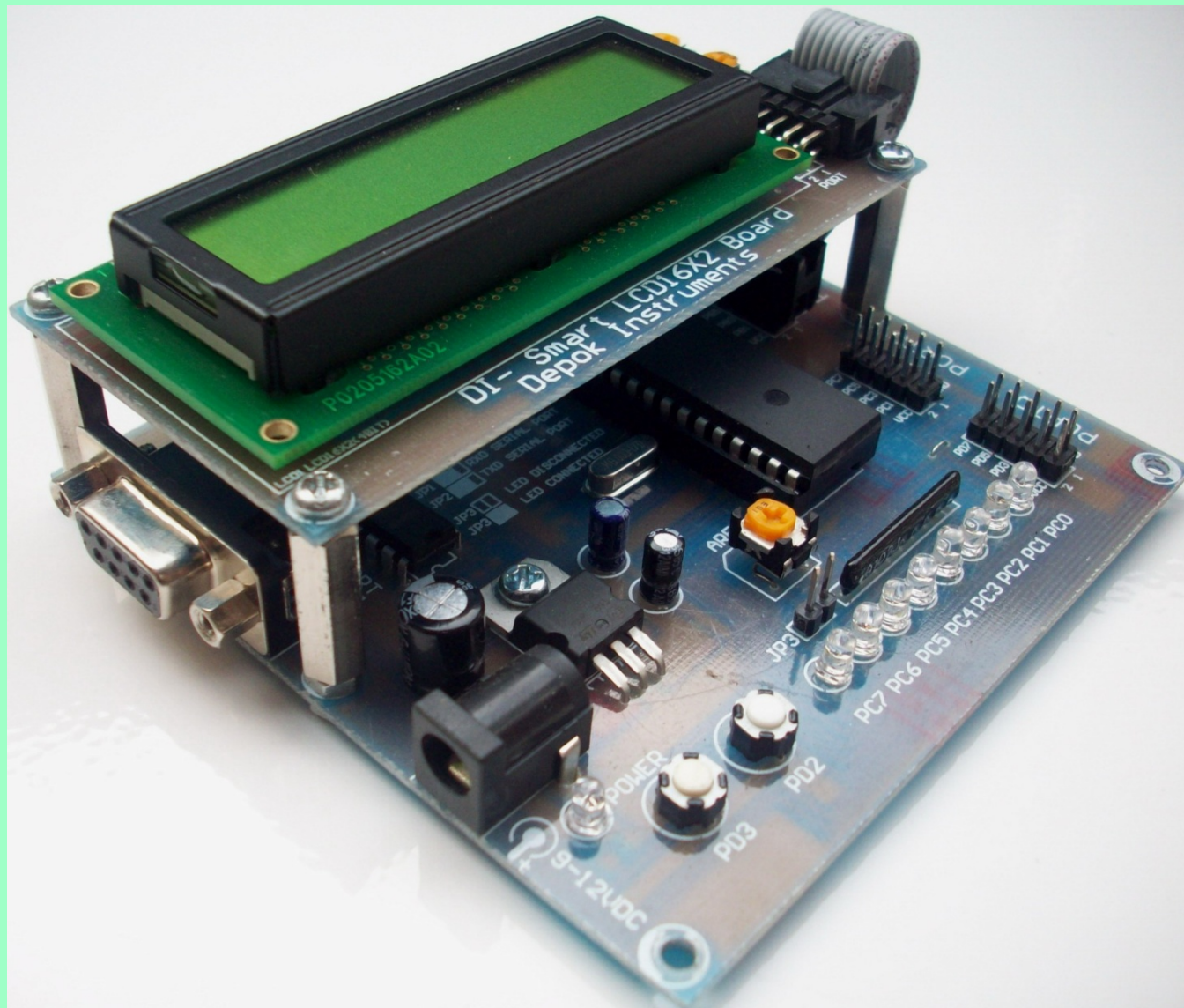


PORTB

9

8

DI-Smart AVR.16 System + DI-Smart LCD16x2 Board



Pemrograman Tampilan LCD (Setup "Wizard")

1

USART Analog Comparator ADC SPI
I2C 1 Wire TWI (I2C)
Alphanumeric LCD
Bit-Banged Project Information
Chip Ports External IRQ Timers

Chip: ATmega16
Clock: 11.059200 MHz

Check Reset Source

Program Type:
Application

2

USART Analog Comparator ADC SPI
I2C 1 Wire TWI (I2C)
Bit-Banged Project Information
Chip Ports External IRQ Timers

Alphanumeric LCD

Enable Alphanumeric LCD Support

Characters/Line: 16

Connections

LCD Module AVR

RS	PORTB	Bit: 0
RD	PORTB	Bit: 1
EN	PORTB	Bit: 2
D4	PORTB	Bit: 4
D5	PORTB	Bit: 5
D6	PORTB	Bit: 6
D7	PORTB	Bit: 7

3

File Program Edit Help

Generate program, save and exit

Pemrograman Tampilan LCD (Hasil *Setup "Wizard"*)

```
D:\PROGRAM_C\AVR\PO2_LCD.c
Notes P02_LCD.c
133
134 // Alphanumeric LCD initialization
135 // Connections specified in the
136 // Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
137 // RS - PORTB Bit 0
138 // RD - PORTB Bit 1
139 // EN - PORTB Bit 2
140 // D4 - PORTB Bit 4
141 // D5 - PORTB Bit 5
142 // D6 - PORTB Bit 6
143 // D7 - PORTB Bit 7
144 // Characters/line: 16
145 lcd_init(16);
146
147 while (1)
148 {
149     // Place your code here
150
151 }
152
153
```

`lcd_init(16)`
= Ada 16 kolom
yang dapat
diisi karakter.

Pemrograman Tampilan LCD (Sintaks Program LCD)

- `lcd_init([char]);` //Inisialisasi jumlah kolom di LCD
 - ▣ `lcd_init(16);` //Jumlah kolom = 16
- `lcd_clear();` //Membersihkan tampilan LCD dan `cursor = 0,0`
 - ▣ `lcd_clear();`
- `lcd_gotoxy([char], [char]);` //Memposisikan `cursor = x,y`
 - ▣ `lcd_gotoxy(0,1);` //Mulai dari nol
- `lcd_putchar([char]);` // = Put Char-Byte (0-255)
 - ▣ `lcd_putchar('A');` // = `lcd_putchar(0x41)`
- `lcd_putsf([string]);` // = Put String on Flash
 - ▣ `lcd_putsf("-SELAMAT DATANG-");`
- `lcd_puts([*RAM]);` //

Pemrograman Tampilan LCD (L1=Kelap-Kelip "-SELAMAT DATANG-") [1/2]

```
□ #include <mega16.h>
□ #include <delay.h>
□ #include <alcd.h>
□ void main(void)
□ {
  □ // RS - PORTB Bit 0
  □ // RD - PORTB Bit 1
  □ // EN - PORTB Bit 2
  □ // D4 - PORTB Bit 4
  □ // D5 - PORTB Bit 5
  □ // D6 - PORTB Bit 6
  □ // D7 - PORTB Bit 7
  □ // Characters/line: 16
  □ lcd_init(16);
```

Pemrograman Tampilan LCD

(L2=Kelap-Kelip "-PEJUANG BANGSA-") [2/2]

```
□ while(1)
  ■ {
  ■ lcd_putsf("-SELAMAT DATANG-");
  ■ lcd_gotoxy(0,1);
  ■ lcd_putsf("-PEJUANG BANGSA-");
  ■ delay_ms(250);
  ■ lcd_clear();
  ■ delay_ms(250);
  ■ }
□ }
```

Pemrograman Tampilan LCD (Sintaks "sprintf")

- `sprintf([char *str, char flash *fmtstr [, arg1, arg2, ...]]);`
 - `#include <stdio.h> //Library serial dan konversi karakter`
 - `#include <alcd.h>`
 - `unsigned char cdata=0, cstr[16];`
 - `void main(void)`
 - {
 - `sprintf(cstr,"Nilai Data: %04d",cdata++);`
 - `lcd_puts(cstr);`
 - }
- `cdata` = Alamat di RAM yang dialokasikan untuk menyimpan data sementara.
- `cstr[16]` = Alamat di RAM yang tersusun secara berurutan yang dialokasikan sebagai hasil konversi dari sintaks "sprintf".
- `%d` = Decimal; `%x` = Hexadesimal; `%c` = Character; `%s` = Strings

Pemrograman Tampilan LCD (*Up-Counter* di LCD) [1/2]

```
□ #include <mega16.h>
□ #include <delay.h>
□ #include <stdio.h>
□ #include <alcd.h>
□ unsigned char cdata=0, cstr[16];
□ void main(void)
□ {
  □ // RS - PORTB Bit 0
  □ // RD - PORTB Bit 1
  □ // EN - PORTB Bit 2
  □ // D4 - PORTB Bit 4
  □ // D5 - PORTB Bit 5
  □ // D6 - PORTB Bit 6
  □ // D7 - PORTB Bit 7
  □ // Characters/line: 16
```

Pemrograman Tampilan LCD

(Up-Counter di LCD) [2/2]

```
□ lcd_init(16);  
□ while(1)  
  ■ {  
  ■ sprintf(cstr,"Nilai Data: %04d",cdata++);  
  ■ lcd_puts(cstr);  
  ■ delay_ms(250);  
  ■ lcd_clear();  
  ■ delay_ms(250);  
  ■ }  
□ }
```

E-BOOK DINS

- <http://depokinstruments.com/category/3-e-book/>