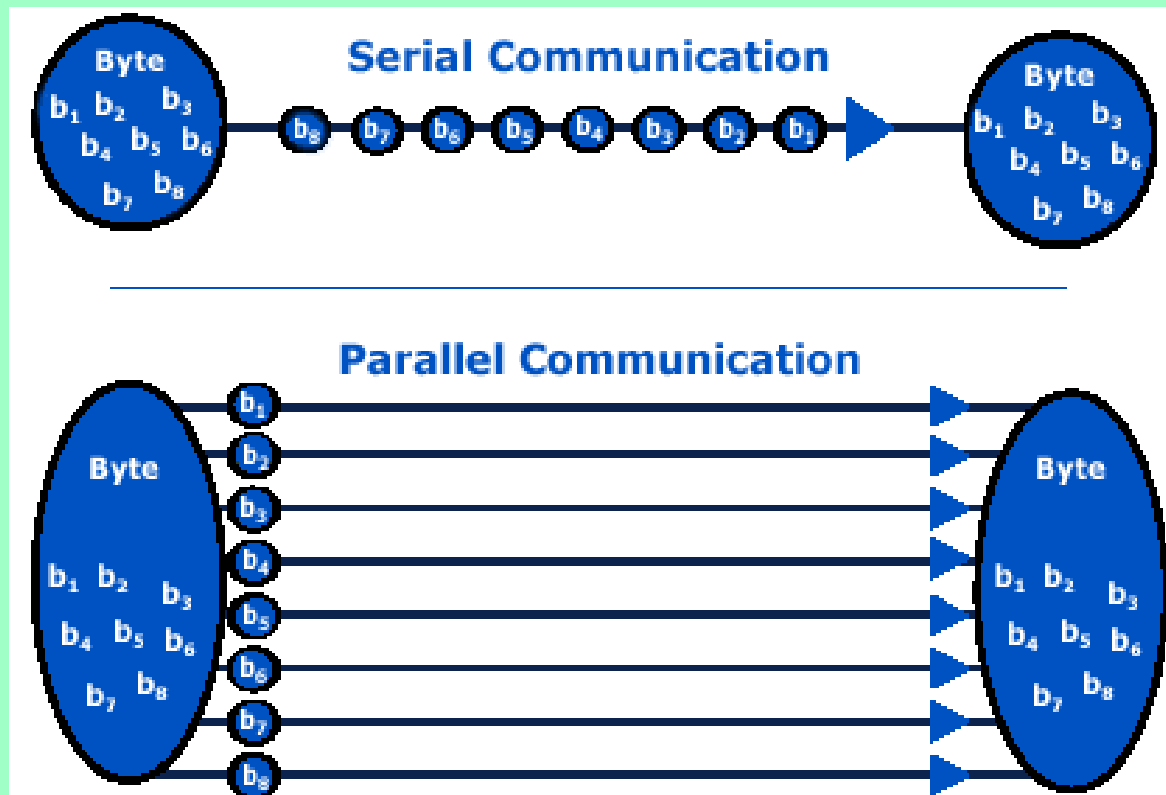


**MIKROKONTROLER ATMEGA
BERBASIS
CODEVISION AVR
(SERIAL U(S)ART)**

Teori U(S)ART

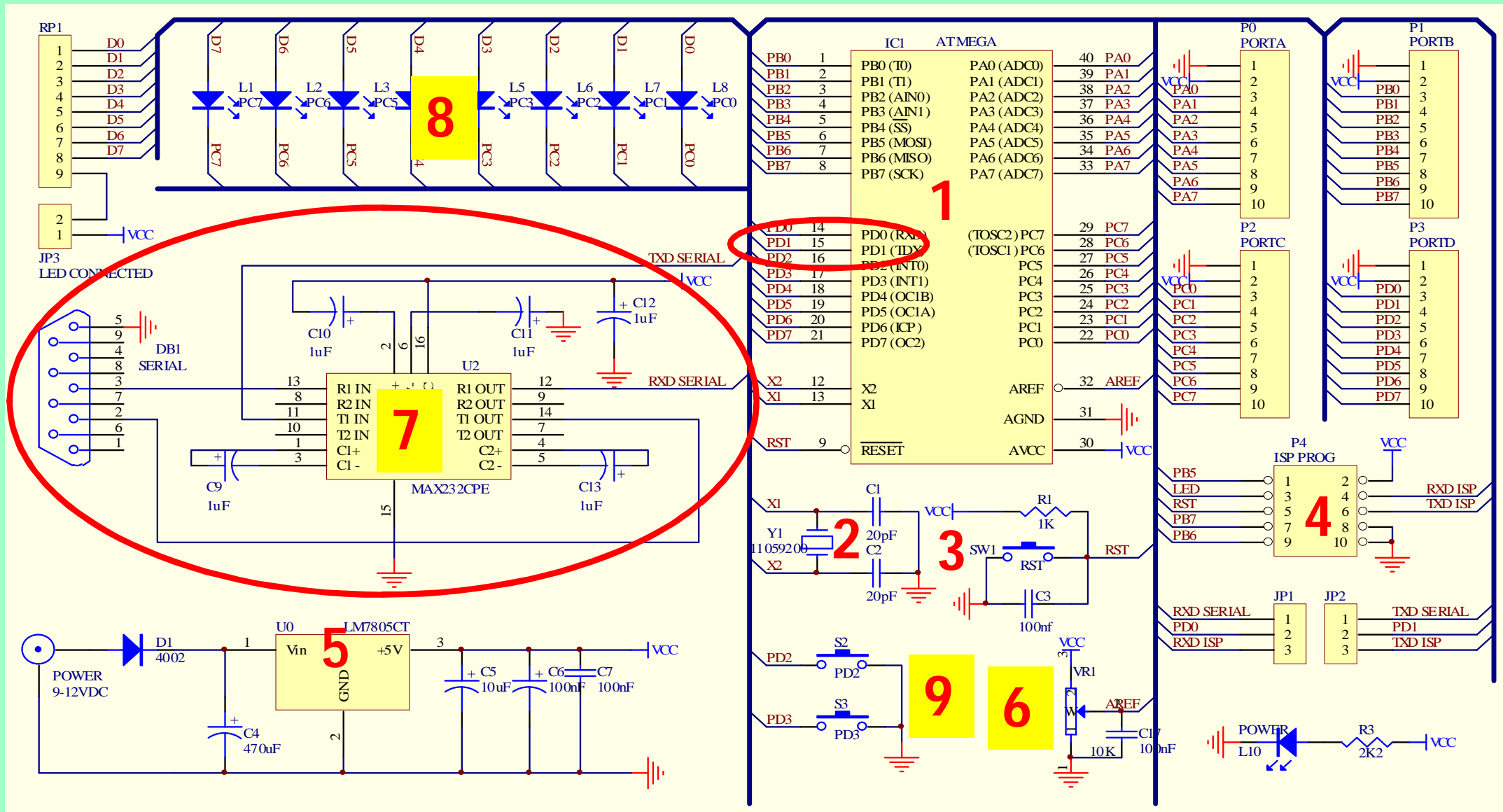
- U(S)ART = Universal (Synchronous) Asynchronous Rx Tx
 - ▣ adalah standar komunikasi serial
- Serial vs Parallel:



Fitur U(S)ART μ C. ATmega16A

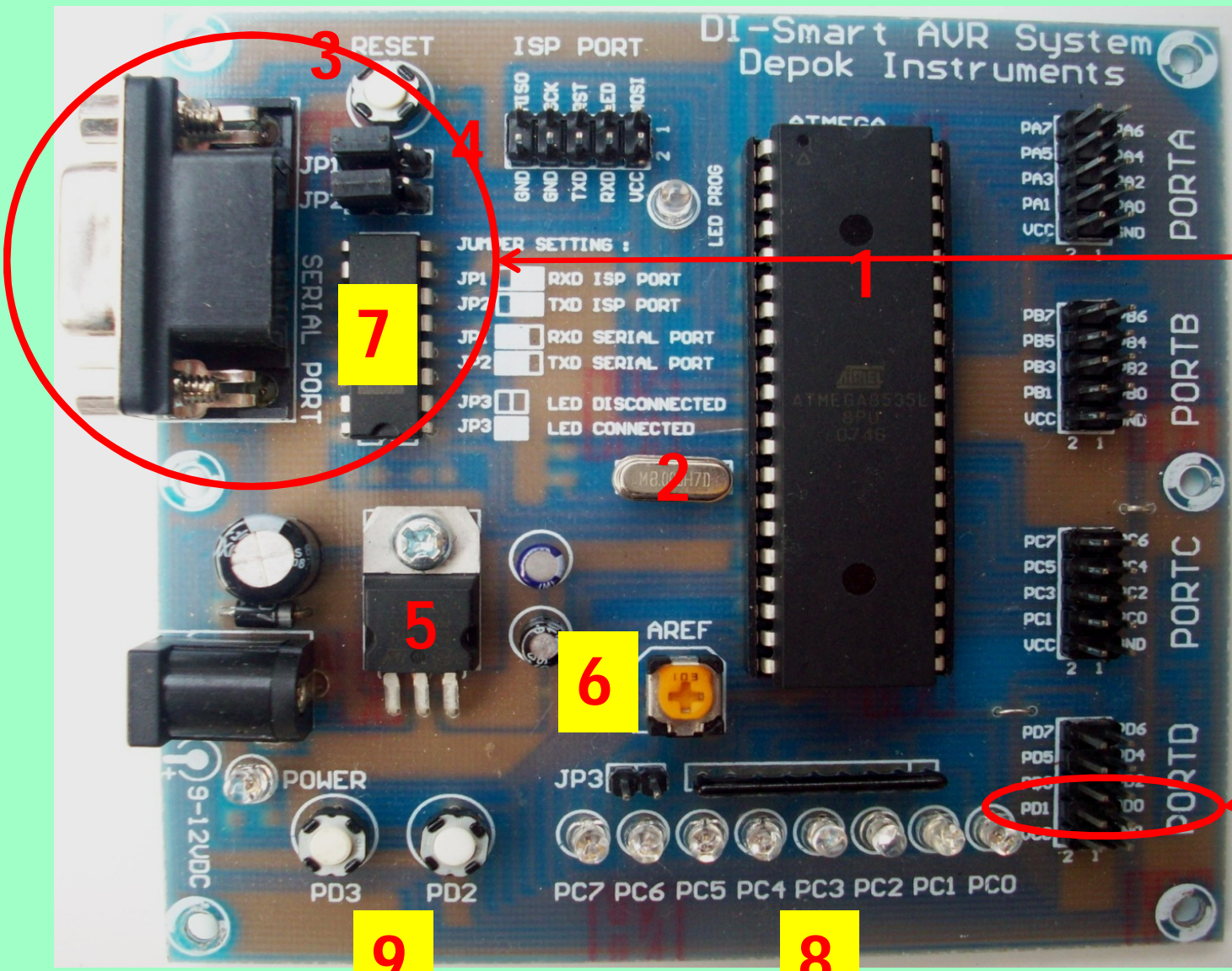
- PIND.0 = RXD
- PORTD.1 = TXD
- RS232 = Protokol standar tegangan untuk komunikasi berjarak sedang.
- USB TO RS232 = Konverter komunikasi USB ke RS232.

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



Serial RS232

PD0=RX
PD1=TX

9

8

Pemrograman U(S)ART pada CVAVR

- Inisialisasi "*Wizard*"
 - ▣ Aktivasi "*Receiver*"
 - ▣ Aktivasi "*Transmitter*"
 - ▣ Pemilihan "*Baud Rate*" yang akan digunakan
 - ▣ Pemilihan "*Communication Parameters*"
 - ▣ "*Mode*" Serial

Pemrograman U(S)ART (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:

Clock: MHz

Check Reset Source

Program Type:

2

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

Receiver Rx Interrupt

Transmitter Tx Interrupt

Baud Rate: x2



Baud Rate Error: 0.0%

Communication Parameters:

Mode:

3

File Program Edit Help

Generate program, save and exit

Pemrograman U(S)ART (Inisialisasi U(S)ART Hasil "Wizard")

```
D:\V\PROGRAM C\AVR\P04_USART.c
Notes P04_USART.c
106 MCUCR=0x00;
107 MCUCSR=0x00;
108
109 // Timer(s)/Counter(s) Interrupt(s) initialization
110 TIMSK=0x00;
111
112 // USART initialization
113 // Communication Parameters: 8 Data, 1 Stop, No Parity
114 // USART Receiver: On
115 // USART Transmitter: On
116 // USART Mode: Asynchronous
117 // USART Baud Rate: 9600
118 UCSRA=0x00;
119 UCSRB=0x18;
120 UCSRC=0x86;
121 UBRRH=0x00;
122 UBRRL=0x47;
123
124 // Analog Comparator initialization
125 // Analog Comparator: Off
126 // Analog Comparator Input Capture by Timer/Counter 1: Off
127 ACSR=0x80;
128 SFIOR=0x00;
```

Inisialisasi USART
sesuai dengan yang
diinputkan saat
Wizard.

Pemrograman U(S)ART

(Sintaks Program U(S)ART - RX)

- `getchar(void);` //Membaca *buffer* serial penerima.
 - ▣ `unsigned char char_data;`
 - ▣ `void main(void)`
 - `{char_data = getchar();}`
- `scanf(char flash *fmtstr,...);` //Membaca *buffer* serial penerima.
 - ▣ `unsigned char char_data;`
 - ▣ `void main(void)`
 - `{scanf("%d",&char_data);}`
- `char_data` = Alamat di RAM yang dialokasikan untuk menyimpan data sementara.

Pemrograman U(S)ART (Sintaks Program U(S)ART - TX)

- `putchar([char]); // = Put Char-Byte (0-255).`
 - ▣ `putchar('A'); // = putchar(0x41)`
- `putsf([string]); // = Put String on Flash.`
 - ▣ `putsf("-SELAMAT DATANG-");`
- `puts([*RAM]); // Lihat berikutnya.`
- `printf([char flash *fmtstr [, arg1, arg2, ...]]); // Lihat berikutnya.`

Pemrograman U(S)ART (Sintaks "sprintf")

- `sprintf([char *str, char flash *fmtstr [, arg1, arg2, ...]]);`
- `#include <stdio.h> //Library serial dan konversi karakter`
- `unsigned char cdata=0, cstr[16];`
 - `void main(void)`
 - {
 - `sprintf(cstr,"Nilai Data: %04d",cdata++);`
 - `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 U(S)ART (Sintaks "printf")

- `printf([char flash *fmtstr [, arg1, arg2, ...]]);`
 - `#include <stdio.h> //Library serial dan konversi karakter`
 - `unsigned char cdata=0, cstr[16];`
 - `void main(void)`
 - `{`
 - `Printf("Nilai Data: %04d",cdata++);`
 - `}`
- `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 U(S)ART (Echo)

```
□ #include <mega16.h>
□ #include <stdio.h>
□ unsigned char char_data;
□ void main(void)
  □ {
  □ // USART initialization
  □ // 8D-1S-NoParity; RX: On; TX: On; Async; 9600 BPS
  □ UCSRA=0x00; UCSRB=0x18; UCSRC=0x86;
  □ UBRRH=0x00; UBRRL=0x47;
  □ while (1)
    ■ {
    ■ char_data = getchar(); //Receive
    ■ putchar(char_data); //Transmit
    ■ }
  □ }
```


Fungsi Transfer Sensor LM35DZ

- Tinjau-ulang dasar ADC:
 - ▣ **$\text{Data_ADC} = (\text{Vin}/\text{Vref}) * (\text{Maksimal_Data_Bit})$**
 - Data_ADC = Hasil konversi ADC internal ATmega16A
 - Vin = Tegangan LM35
 - Vref = Sesuai pengaturan di "Wizard" CVAVR = 5V
 - Maksimal_Data_Bit = 10 Bit = 1023
- Sensitivitas LM35 = 10 mV/ °C
- Temperatur terukur (T):
 - ▣ $T = (\text{Vin}/10 \text{ mV}) \text{ } ^\circ\text{C}$
 - ▣ **$T = ((\text{Data_ADC} * 5\text{V}/1023)/10 \text{ mV}) \text{ } ^\circ\text{C}$**

"Data Logger Temperatur berbasis LM35DZ" (Setup "Wizard") [1/2]

1

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

Chip:

Clock: MHz

Check Reset Source

Program Type:

2

I2C	1 Wire	TWI (I2C)	
Alphanumeric LCD			
Bit-Banged		Project Information	
Chip	Ports	External IRQ	Timers

USART	Analog Comparator	ADC	SPI
-------	-------------------	-----	-----

Receiver Rx Interrupt

Transmitter Tx Interrupt

Baud Rate: x2

Baud Rate Error: 0.0%

Communication Parameters:

Mode:

“Data Logger Temperatur berbasis LM35DZ” (Setup “Wizard”) [2/2]

3

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

ADC Enabled Use 8 bits
 Interrupt

Volt. Ref: AREF pin

Clock: 172.800 kHz

Auto Trigger Source: Free Running

4

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

5

File Program Edit Help



Generate program, save and exit

“Data Logger berbasis Sensor LM35DZ” (Skrip Program CVAVR) [1/3]

```
□ #include <mega16.h>
□ #include <delay.h>
□ #include <stdio.h>
□ #define ADC_VREF_TYPE 0x00
□ unsigned int read_adc(unsigned char adc_input)
□ {
□ ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
□ delay_us(10);
□ ADCSRA|=0x40;
□ while ((ADCSRA & 0x10)==0);
□ ADCSRA|=0x10;
□ return ADCW;
□ }
```

“Data Logger berbasis Sensor LM35DZ” (Skrip Program CVAVR) [2/3]

- `unsigned char T; //T = Temperatur`
- `unsigned int idata;`
- `float fdata; //Nilai pecahan agar perhitungan baik.`
- `void main(void)`
 - `{`
 - `UCSRA=0x00; UCSRB=0x18; UCSRC=0x86;`
 - `UBRRH=0x00; UBRRL=0x47;`
 - `ADMUX=ADC_VREF_TYPE & 0xff; ADCSRA=0xA6; SFIOR&=0x1F;`
 - `while (1)`
 - `{`
 - `idata = read_adc(0);`
 - `Printf("Data ADC: %04d",idata);`

"Data Logger berbasis Sensor LM35DZ" (Skrip Program CVAVR) [3/3]

```
■ fdata = (((float)idata*5)/1023)/0.01;  
■ T = (char)fdata;  
■ Printf("Temperatur: %02d C",T);  
■ delay_ms(250);  
■ }  
□ }
```

E-BOOK DINS

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