

DT-I/O

DT-I/O

Application Note

AN193 – Pressure and Temperature Monitoring via TCP/IP

Oleh: Tim IE

Antarmuka menggunakan protokol komunikasi TCP/IP mungkin masih banyak dihindari oleh para pengembang aplikasi *embedded system* sederhana. Hal tersebut dikarenakan mekanisme dan prosedur TCP/IP relatif lebih rumit apabila dibandingkan dengan komunikasi menggunakan jalur *parallel* ataupun UART. Oleh karena itu, aplikasi kali ini akan membahas cara mudah mengkomunikasikan dua buah mikrokontroler menggunakan protokol TCP/IP dengan bantuan modul DT-I/O TCP/IP to UART Converter. Modul tersebut berfungsi untuk melakukan konversi data serial yang diterima dari mikrokontroler ke dalam paket data TCP/IP sehingga siap dikirimkan melalui jalur *ethernet*. Dengan ini, proses pertukaran data antar mikrokontroler tidak lagi perlu memikirkan protokol TCP/IP itu sendiri, karena hal tersebut ditangani sepenuhnya oleh DT-I/O TCP/IP to UART Converter.

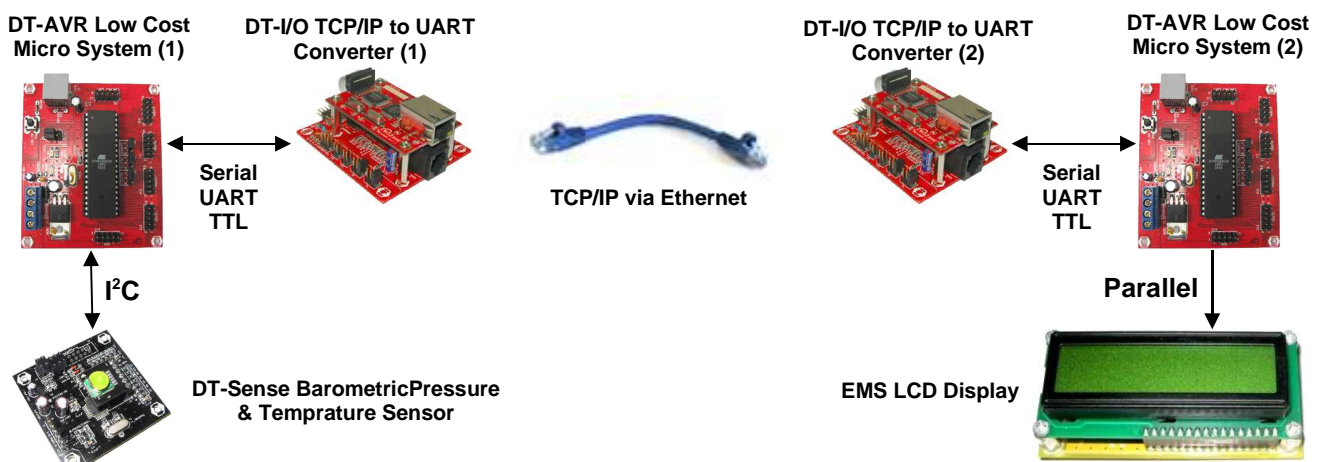
Berikut adalah perlengkapan yang diperlukan dalam aplikasi ini :

- 2x DT-AVR Low Cost Micro System
- 2x DT-I/O TCP/IP to UART Converter
- 1x DT-SENSE Barometric Pressure & Temperature Sensor
- 1x EMS LCD Display
- 1x Kabel LAN tipe *cross*
- Beberapa kabel *jumper*
- 2x *Power supply* 12V dan 2x *power supply* 5V
- Komputer (untuk melakukan konfigurasi DT-I/O TCP/IP to UART Converter)

Pengaksesan DT-SENSE Barometric Pressure & Temperature Sensor dilakukan oleh mikrokontroler ATmega8535 melalui protokol I^2C , sedangkan pengaksesan EMS LCD Display dilakukan secara *parallel*. Proses pertukaran data antara ATmega8535 dengan DT-I/O TCP/IP to UART dilakukan melalui jalur UART.

Pada aplikasi kali ini terdapat dua buah program mikrokontroler. Kedua program tersebut dikembangkan dengan bahasa C menggunakan bantuan *compiler* CodeVisionAVR. Program pertama (yaitu *jds.hex*) digunakan pada mikrokontroler ATmega8535 yang terhubung langsung dengan DT-SENSE Barometric Pressure & Temperature Sensor. Sedangkan program kedua (yaitu *display.hex*) akan digunakan pada mikrokontroler ATmega8535 yang terhubung langsung dengan EMS LCD Display.

Adapun blok diagram dari aplikasi ini adalah sebagai berikut :



Gambar 1
Blok Diagram AN193

Hubungan antar modul adalah sebagai berikut :

DT-AVR Low Cost Micro System (1 & 2)	DT-I/O TCP/IP to UART Converter (1 & 2)
GND (J13 pin 1)	GND (J2 pin 1)
RXD (J13 pin 3)	TXD (J2 pin 3)
TXD (J13 pin 4)	RXD (J2 pin 4)

Tabel 1
Hubungan DT-AVR Low Cost Micro System dengan DT-I/O TCP/IP to UART Converter

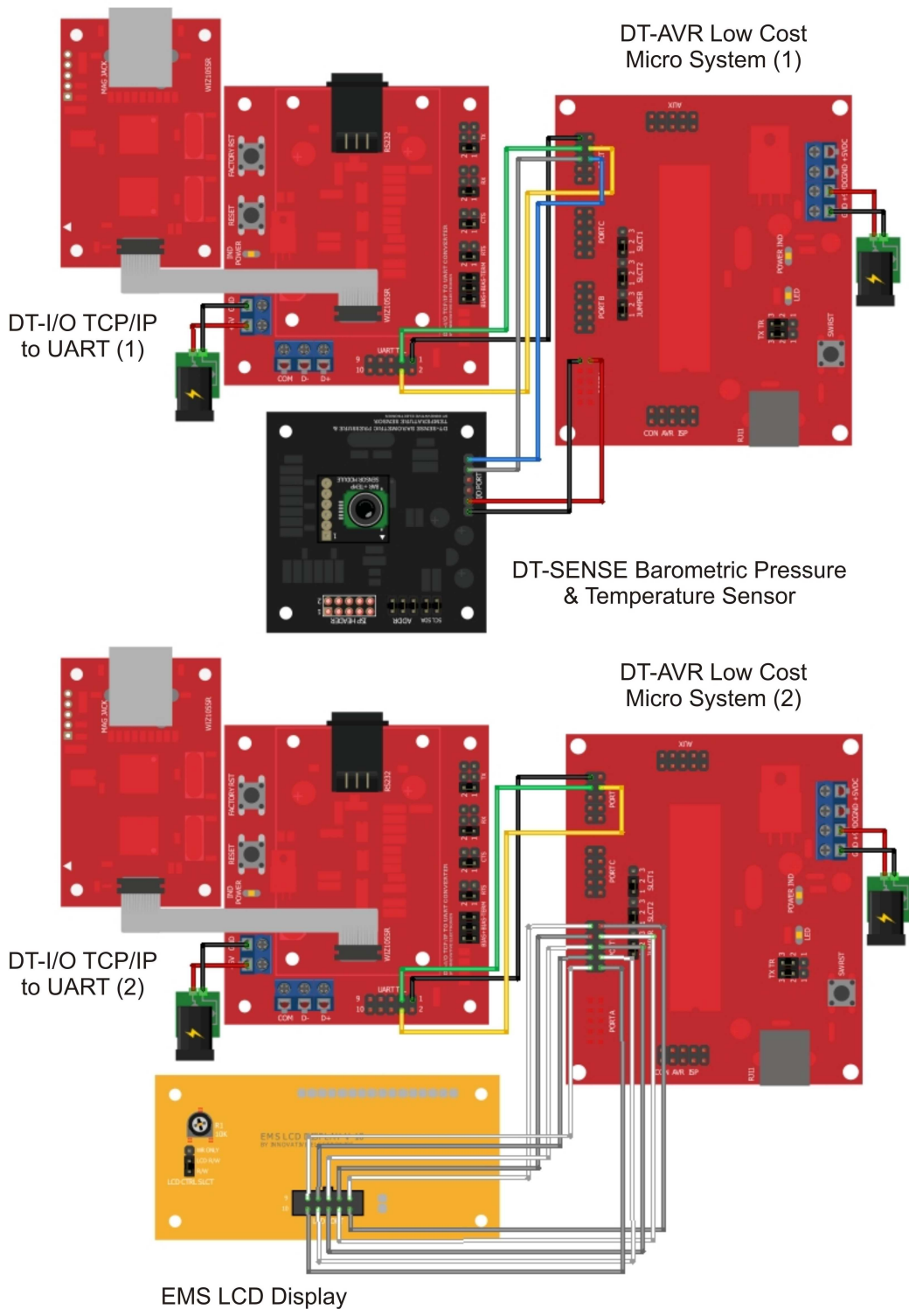
DT-AVR Low Cost Micro System (1)	DT-SENSE Barometric Pressure & Temperature Sensor
GND (J10 pin 1)	GND (J1 pin 1)
VCC (J10 pin 2)	VCC (J1 pin 2)
PD.2 (J13 pin 6)	MAIN SDA (J1 pin 5)
PD.3 (J13 pin 5)	MAIN SCL (J1 pin 6)

Tabel 2
Hubungan DT-AVR Low Cost Micro System dengan DT-SENSE Barometric Pressure & Temperature Sensor

DT-AVR Low Cost Micro System (2)	EMS LCD Display
GND (J7 pin 1)	GND (J3 pin 1)
VCC (J7 pin 2)	+5V (J3 pin 2)
PB.0 (J7 pin 3)	RS (J3 pin 3)
PB.1 (J7 pin 4)	R/W (J3 pin 4)
PB.2 (J7 pin 5)	E (J3 pin 5)
PB.3 (J7 pin 6)	BL (J3 pin 6)
PB.4 (J7 pin 7)	DB4 (J3 pin 7)
PB.5 (J7 pin 8)	DB5 (J3 pin 8)
PB.6 (J7 pin 9)	DB6 (J3 pin 9)
PB.7 (J7 pin 10)	DB7 (J3 pin 10)

Tabel 3
Hubungan DT-AVR Low Cost Micro System dengan EMS LCD Display

Setelah menghubungkan modul – modul tersebut menggunakan kabel jumper, lakukan pengecekan kembali menggunakan *multimeter*, apakah koneksi antar modul sudah benar atau tidak. Pastikan juga bahwa tidak terjadi hubungan singkat antara VCC dan GND sebelum memberikan catu daya. Ilustrasi koneksi antar modul terdapat pada **Gambar 2**.



Gambar 2
Hubungan antar modul pada AN193

Modul – modul diatas perlu dikonfigurasi terlebih dahulu agar dapat bekerja sesuai dengan yang diharapkan. Berikut ini adalah langkah – langkah konfigurasi yang perlu dilakukan :

- **DT-AVR Low Cost Micro System (1 & 2)**

- Mikrokontroler ATmega8535 pada DT-AVR Low Cost Micro System menggunakan osilator eksternal berupa *crystal* dengan nilai frekuensi 4 MHz. Silahkan melakukan pengaturan *fusebit* pada ATmega8535 terlebih dahulu agar dapat bekerja dengan osilator eksternal 4 MHz. Informasi lebih detail mengenai pengaturan *fusebit* terdapat pada AN177.
- Komunikasi antara ATmega8535 dengan DT-I/O TCP/IP to UART dilakukan menggunakan komunikasi serial UART TTL. Maka dari itu perlu dilakukan pengaturan jumper J4 dan J5 pada posisi 2-3.

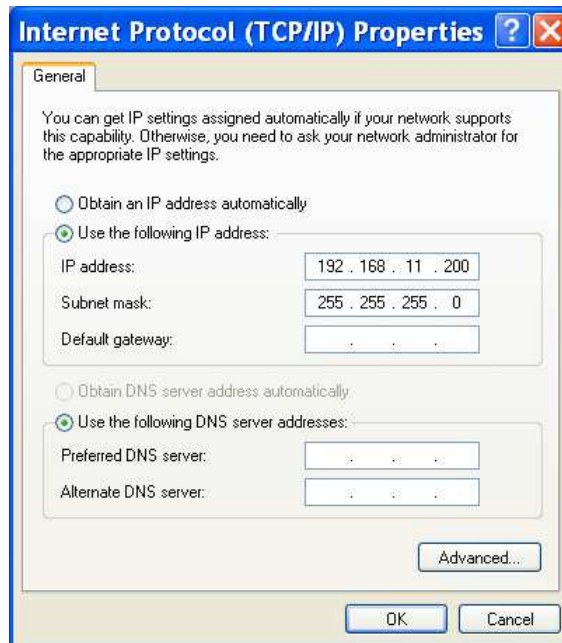
1	2	3	J4
1	2	3	J5

Gambar 3
Pengaturan Jumper J4 dan J5 pada DT-AVR Low Cost Micro System

- **Komputer**

- Proses konfigurasi DT-I/O TCP/IP to UART Converter dilakukan menggunakan bantuan komputer. Agar komputer dapat berkomunikasi dengan DT-I/O TCP/IP to UART Converter, perlu dilakukan konfigurasi pada IP address yang digunakan sebagai berikut :

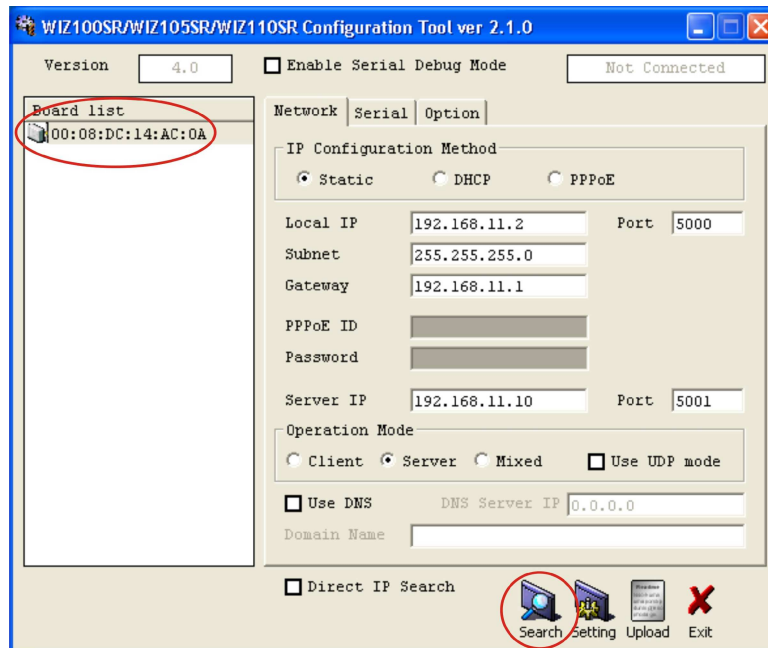
- IP address : 192.168.11.200
- Subnet mask : 255.255.255.0



Gambar 4
Pengaturan IP pada komputer

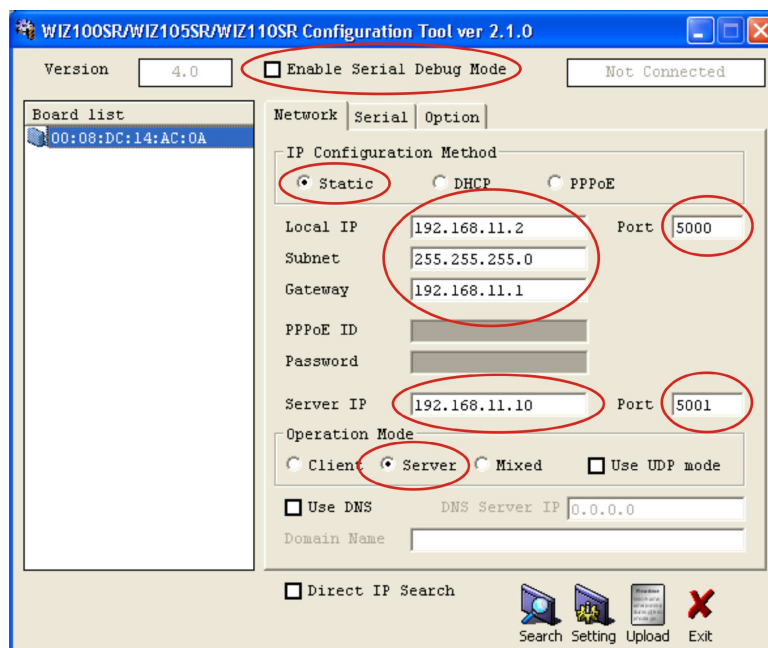
- **DT-I/O TCP/IP to UART Converter (1)**

- Berikan catu daya +5V pada terminal biru J8. Perhatikan polaritas catudaya, agar tidak terjadi pemasangan yang terbalik.
- Hubungkan DT-I/O TCP/IP to UART (1) dengan komputer menggunakan kabel LAN tipe *cross*.
- Jalankan program WIZ1x0_CFG.exe, yang dapat ditemukan pada DVD yang disertakan pada paket penjualan produk tersebut.
- Tekan tombol search untuk memulai pencarian modul.



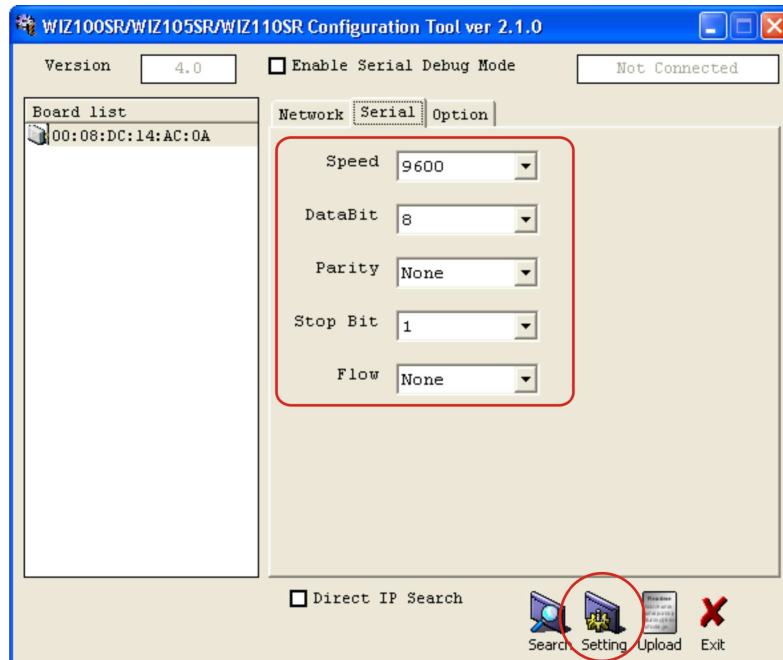
Gambar 5
Pencarian Modul DT-I/O TCP/IP to UART Converter

- Lakukan pengaturan IP dan *baudrate* pada modul DT-I/O TCP/IP to UART Converter (1) sebagai berikut :
 - Bagian Network :
 - Hilangkan tanda centang pada “Enable Serial Debug Mode”
 - Pilih IP Static
 - Local IP : 192.168.11.2 dengan nilai Port : 5000
 - Subnet : 255.255.255.0
 - Gateway : 192.168.11.1
 - Server IP : 192.168.11.10 dengan nilai Port : 5001
 - Pilih mode Server.



Gambar 6
Pengaturan DT-I/O TCP/IP to UART Converter (1) pada bagian Network

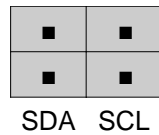
- Bagian Serial :
 - Speed : 9600
 - DataBit : 8
 - Parity : None
 - Stop Bit : 1
 - Flow : None
 - Tekan tombol Setting (di sebelah kanan tombol Search) untuk melakukan upload pengaturan ke DT-I/O TCP/IP to UART Converter (1).



Gambar 7
Pengaturan DT-I/O TCP/IP to UART Converter (1) pada bagian Serial

- **DT-I/O TCP/IP to UART Converter (2)**
 - Lakukan pengaturan IP dan *baudrate* pada modul DT-I/O TCP/IP to UART Converter (2) menggunakan cara yang sama dengan pengaturan DT-I/O TCP/IP to UART Converter (1). Adapun parameter pengaturannya adalah sebagai berikut :
 - Bagian Network :
 - Berikan tanda centang pada "Enable Serial Debug Mode"
 - Pilih IP Static
 - Local IP : 192.168.11.100 dengan nilai Port : 5000
 - Subnet : 255.255.255.0
 - Gateway : 192.168.11.1
 - Server IP : 192.168.11.2 dengan nilai Port : 5000
 - Pilih mode Client.
 - Bagian Serial :
 - Speed : 9600
 - DataBit : 8
 - Parity : None
 - Stop Bit : 1
 - Flow : None
 - Tekan tombol Setting (di sebelah kanan tombol Search) untuk melakukan upload pengaturan ke DT-I/O TCP/IP to UART Converter (2).

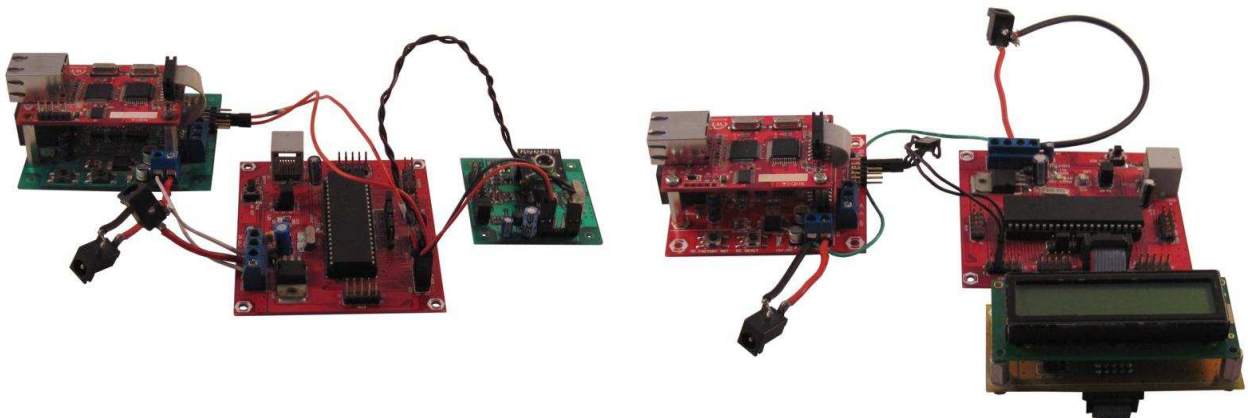
- **DT-SENSE Barrometric Pressure & Temperature Sensor**
 - Pasang semua jumper pada J4 untuk mengaktifkan resistor *pull-up*.



Gambar 8
Pengaturan Jumper J4 pada DT-SENSE Barometric Pressure & Temperature Sensor

Apabila konfigurasi di atas telah selesai dilakukan, silahkan mengikuti langkah – langkah berikut ini :

1. Hubungkan *programmer* mikrokontroler AVR yang mendukung fitur ISP dengan DT-AVR Low Cost Micro System, seperti DT-HiQ AVR In System Programmer, DT-HiQ AVR USB ISP, atau *programmer* lainnya.
2. Berikan catu daya +9V - +12V pada kedua buah modul mikrokontroler melalui terminal biru.
3. Download file dengan ekstensi .hex (jds.hex) yang berada di (AN193\Program Baca Sensor\Exe\jds.hex) pada DT-AVR Low Cost Micro System yang akan dihubungkan dengan DT-SENSE Barometric Pressure & Temperature Sensor.
4. Download file dengan ekstensi .hex (display.hex) yang berada di (AN193\Program Display\Exe\display.hex) pada DT-AVR Low Cost Micro System yang akan dihubungkan dengan EMS LCD Display.
5. Hubungkan kedua buah modul DT-I/O TCP/IP to UART Converter menggunakan kabel LAN tipe *cross*.
6. Berikan catu daya +5V pada modul DT-I/O TCP/IP to UART Converter (1 & 2) melalui terminal biru J8.



Gambar 9
Rangkaian antar modul pada AN193

Apabila tidak terdapat kesalahan pada langkah – langkah di atas, seharusnya akan keluar tampilan data suhu dan tekanan dari DT-SENSE Barometric Pressure & Temperature Sensor pada EMS LCD Display seperti gambar berikut.



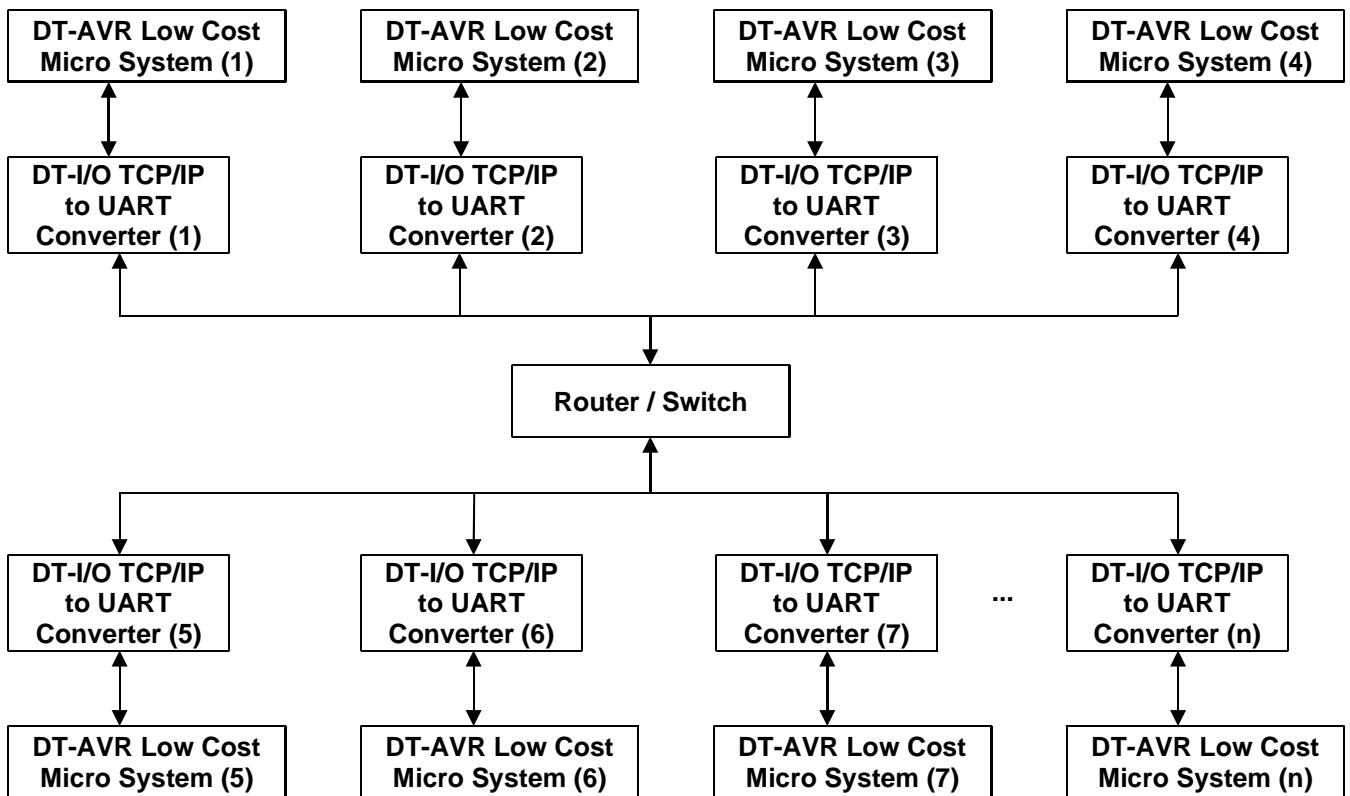
Gambar 10
Tampilan data suhu dan tekanan pada EMS LCD Display

Berikut adalah tampilan EMS LCD Display saat koneksi antar DT-I/O TCP/IP to UART Converter terputus.



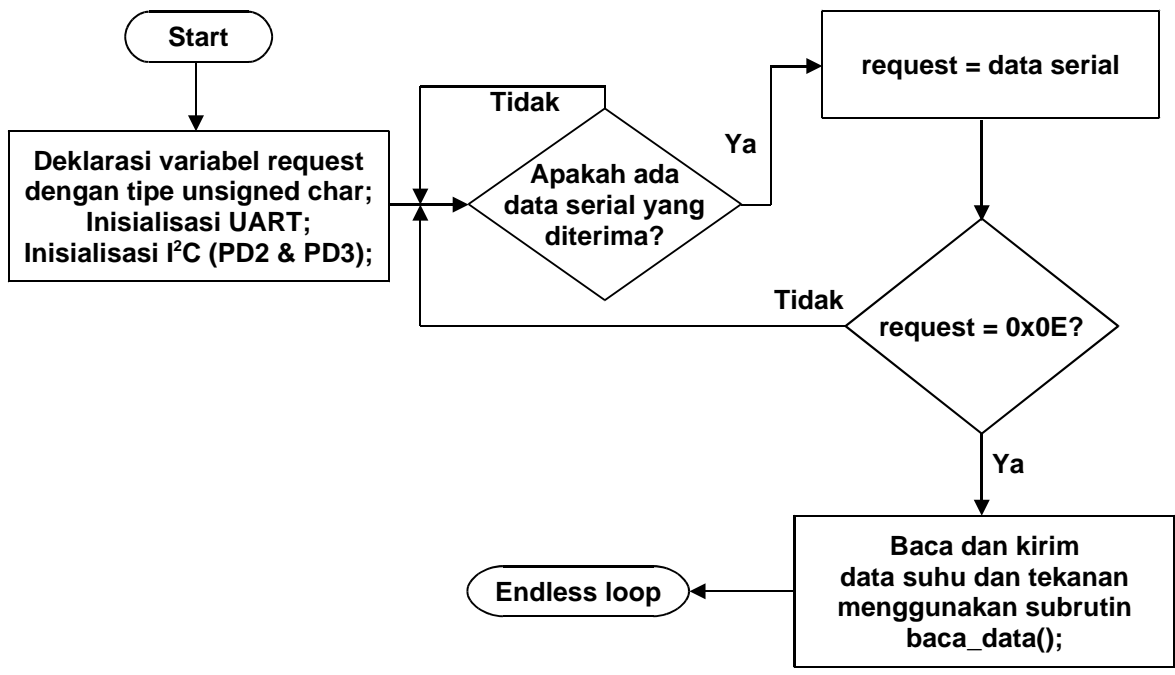
Gambar 11
Tampilan EMS LCD Display pada saat koneksi antar DT-I/O TCP/IP to UART Converter terputus

Aplikasi ini dapat dikembangkan menjadi sistem yang lebih kompleks, yaitu komunikasi jaringan antara banyak mikrokontroler seperti pada **Gambar 12**. Hal tersebut dapat dimanfaatkan untuk sistem telemetri (pengukuran / pelaporan informasi jarak jauh) ataupun sistem kontrol jarak jauh.



Gambar 12
Salah satu pengembangan lebih lanjut dari AN193

Adapun alur program dari jds.hex adalah sebagai berikut :



Gambar 13
Diagram alir program pada DT-AVR Low Cost Micro System (1)

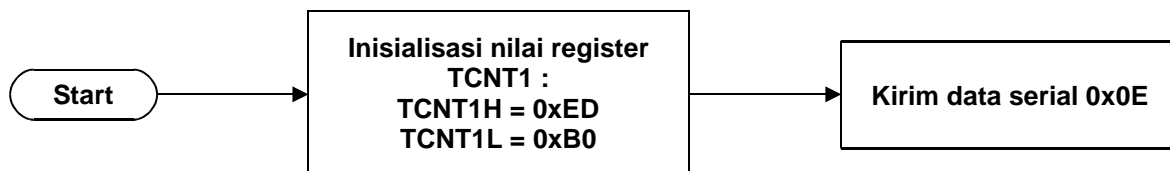
Penjelasan urutan kerja dari program diatas adalah sebagai berikut :

1. Program melakukan deklarasi variabel request
2. Program melakukan inisialisasi register UART (asynchronous, 9600 bps, 8 data bit, 1 stop bit, no parity)
3. Program melakukan inisialisasi protokol I²C (PD2 = SDA & PD3 = SCL)
4. Mikrokontroler akan melakukan proses *pooling* untuk mendeteksi apakah ada data serial yang diterima. Jika ada data yang diterima, data tersebut akan dimasukkan ke dalam variabel request
5. Nilai variabel request akan diperiksa, apakah bernilai 0x0E (nilai dalam hexa) atau tidak. Apabila nilai tersebut sama dengan 0x0E, maka program akan memanggil subutin *baca_data()*. Apabila nilai tersebut tidak sama dengan 0x0E, maka program akan kembali ke langkah 4 untuk melakukan proses *pooling* data serial berikutnya
6. Subrutin *baca_data()* akan melakukan pembacaan data dari DT-SENSE Barometric Pressure & Temperature Sensor menggunakan protokol komunikasi I²C. Langkah dari pembacaan data tersebut dijelaskan pada manual DT-SENSE Barometric Pressure & Temperature Sensor (disertakan pada DVD paket penjualan produk). Karena hasil pembacaan data memiliki tipe integer, data tersebut harus terlebih dahulu dikonversi ke dalam tipe char agar dapat dikirimkan melalui komunikasi UART. Proses konversi tersebut dilakukan menggunakan fungsi *itoa()* yang terdapat pada library *stdlib.h*. Fungsi *itoa()* dipanggil oleh fungsi *baca_data()*
7. Karena satu paket data yang dikirimkan oleh DT-AVR Low Cost Micro System (1) memiliki informasi mengenai suhu dan tekanan, maka perlu diberikan data *header* untuk memisahkan keduanya. Data *header* yang digunakan memiliki nilai 0x0D. Sedangkan sebagai penanda bahwa paket data yang dikirim berasal dari DT-AVR Low Cost Micro System (1), akan diberikan data *header* dengan nilai 'x'. Untuk menandakan bahwa pengiriman paket data telah selesai, DT-AVR Low Cost Micro System (1) akan mengirimkan data dengan nilai 0x0A. Berikut adalah gambar dari paket data yang dikirimkan :

Data Suhu	0x0D	Data Tekanan	'x'	0x0A
-----------	------	--------------	-----	------

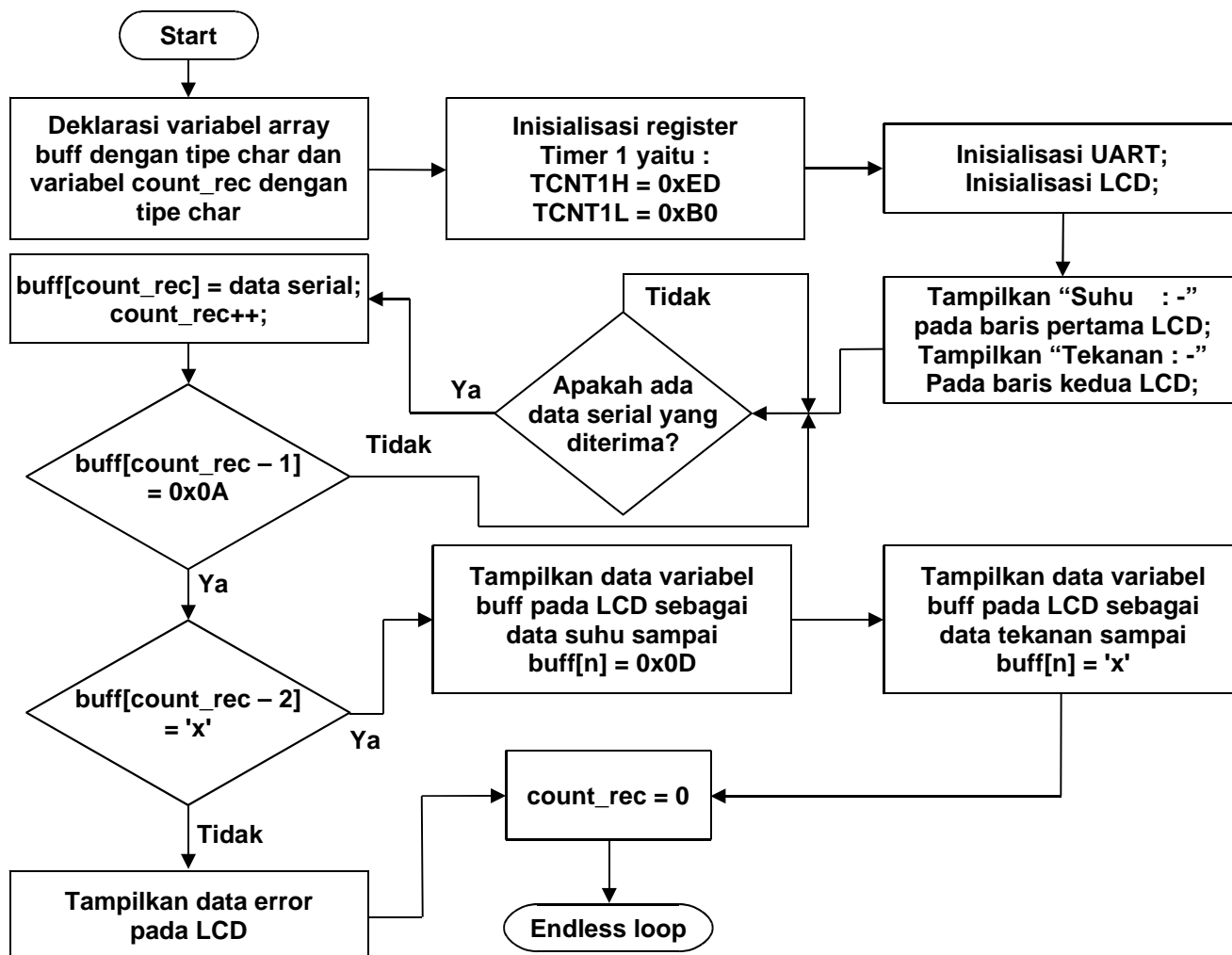
Gambar 14
Paket data yang dikirimkan oleh DT-AVR Low Cost Micro System (1)

DT-AVR Low Cost Micro System (2) akan mengirim data serial 0x0E pada DT-AVR Low Cost Micro System (1) yaitu perintah untuk melakukan pembacaan data suhu dan tekanan. Pengiriman data tersebut dilakukan menggunakan bantuan *peripheral timer* yang ada pada ATmega8535. Adapun *Timer* yang digunakan adalah Timer 1 dengan waktu tunda 300 mS. Apabila Timer 1 *overflow* akan terjadi interupsi, dimana program utama akan dihentikan sementara untuk menjalankan rutin yang terdapat pada fungsi interupsi tersebut. Proses pengaturan nilai waktu tunda dari Timer 1 dilakukan dengan cara memberi nilai 0xED pada register TCNT1H dan 0xB0 pada TCNT1L. Adapun alur program interupsi Timer 1 tersebut adalah sebagai berikut :



Gambar 15
Diagram alir rutin interupsi Timer 1

Berikut ini adalah alur program dari display.hex :



Gambar 16
Diagram alir program pada DT-AVR Low Cost Micro System (2)

Penjelasan urutan kerja dari program diatas adalah sebagai berikut :

1. Program melakukan deklarasi variabel array buff dan variabel count_rec
2. Program melakukan inialisasi Timer 1 dengan nilai TCNT1H = 0xED dan TCNT1L = 0xB0
3. Program melakukan inialisasi register UART (asynchronous, 9600 bps, 8 data bit, 1 stop bit, no parity)
4. Program melakukan inialisasi LCD
5. Pada LCD baris pertama akan ditampilkan tulisan "Suhu : -" dan baris kedua akan ditampilkan tulisan "Tekanan : -" sebagai pembeda data sensor yang akan ditampilkan
6. Mikrokontroler akan melakukan proses *pooling* apakah ada data serial yang diterima atau tidak. Apabila mikrokontroler menerima data serial, data tersebut akan ditampung ke dalam variabel buff
7. Jika data yang diterima tidak sama dengan 0x0A, maka mikrokontroler akan kembali ke langkah 6 untuk melakukan proses *pooling* data serial kembali. Apabila data yang diterima memiliki nilai 0x0A, hal tersebut menandakan bahwa mikrokontroler telah menerima 1 paket data serial yang utuh. Proses berjalan ke langkah selanjutnya
8. Program akan memeriksa apakah data pada buff[count_rec – 2] = 'x' atau tidak. Apabila ya, maka 1 paket data serial tersebut adalah data yang diterima dari DT-AVR Low Cost Micro System (1). Nilai 'x' telah ditentukan sebelumnya sebagai penanda untuk membedakan bahwa data serial yang diterima berasal dari DT-AVR Low Cost Micro System (1), bukan DT-I/O TCP/IP to UART Converter. Proses dilanjutkan ke langkah 9.
Apabila data pada buff[count_rec – 2] tidak sama dengan 'x', maka data tersebut merupakan data dari DT-I/O TCP/IP to UART Converter, dan data tersebut dianggap sebagai data *error*. Data *error* ini akan digunakan sebagai indikator putus atau tidaknya koneksi antara kedua DT-I/O TCP/IP to UART Converter. Proses dilanjutkan ke langkah 10.
9. Mikrokontroler akan menampilkan data suhu dan tekanan pada LCD kemudian melakukan inialisasi ulang nilai variable count_rec. Data suhu dan tekanan tersebut dibedakan oleh data 0x0D. Proses kembali ke langkah 6.
10. Mikrokontroler akan menampilkan data *error* seperti pada **Gambar 11** kemudian akan melakukan inialisasi ulang nilai variable count_rec. Proses kembali ke langkah 6.

Listing program aplikasi ini terdapat pada **AN193.ZIP**

Selamat berinovasi!

*All trademarks, company names, product names and trade names are the property of their respective owners.
All softwares are copyright by their respective creators and/or software publishers.*