

DT-AVR

DT-AVR *Application Note*

AN160 – *Graphic LCD*

Dengan Bahasa C

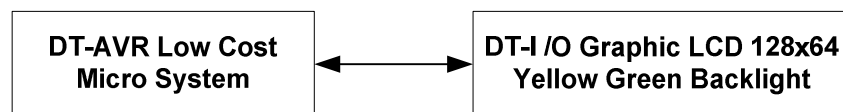
Oleh: Tim IE

Liquid Crystal Display (LCD) merupakan media tampilan yang sering kita jumpai sehari-hari, seperti pada telepon selular, kalkulator, dan jam. Berdasarkan tampilannya, LCD dibedakan menjadi 2 jenis yaitu *character LCD* dan *graphic LCD*. Aplikasi ini akan membahas tentang penggunaan *graphic LCD* 128x64 dots (titik) yang berbasis kontroler KS0108 untuk menampilkan tulisan dan gambar. Aplikasi ini dikembangkan dalam bahasa C menggunakan *editor & compiler* CodeVisionAVR versi evaluasi. Penggunaan bahasa pemrograman sekelas C akan mempermudah implementasi algoritma program pada aplikasi kali ini. Aplikasi ini menggunakan *file library* dari **Fabian Maximilian Thiele** (<http://www.apetech.de>) dengan sedikit modifikasi untuk penggunaan dengan CodeVisionAVR dan disesuaikan untuk keperluan aplikasi ini. *File library* ini digunakan untuk menangani rutin-rutin yang berhubungan dengan kontroler KS0108. Gambar yang akan ditampilkan pada layar *graphic LCD* memiliki ukuran 128 *pixel* x 64 *pixel*, warna hitam putih, dan berekstensi bmp. Proses konversi file *bitmap* gambar ke *file header* yang berupa data heksa menggunakan program BMP2HEX.exe yang dikembangkan dengan Borland Delphi 7.

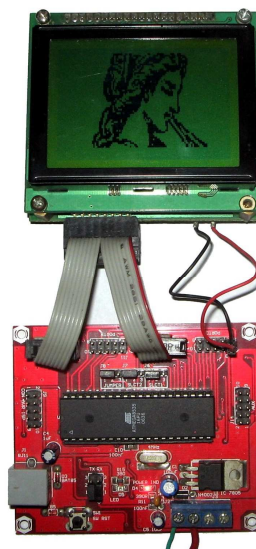
Modul-modul yang digunakan pada aplikasi ini adalah:

- 1 unit modul DT-AVR Low Cost Micro System (DT-AVR LCMS).
- 1 unit modul DT-I/O Graphic LCD 128x64 Yellow Green Backlight.

Blok diagram sistem secara keseluruhan adalah sebagai berikut:



Gambar 1
Blok Diagram AN160



Gambar 2
Contoh Tampilan Gambar Pada Layar *Graphic LCD*

Hubungan antar modul adalah sebagai berikut:

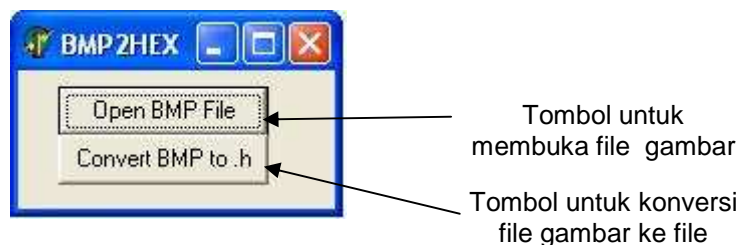
DT-AVR Low Cost Micro System	DT-I/O Graphic LCD 128x64 Yellow Green Backlight
VCC (J13 pin 2)	+5VDC (J1 pin 1)
GND (J13 pin 1)	GND (J1 pin 2)
PA.0 (J10 pin 3)*	CS1 (J2 pin 9)
PA.1 (J10 pin 4)*	CS2 (J2 pin 10)
PA.2 (J10 pin 5)*	D/I (J2 pin 11)
PA.3 (J10 pin 6)*	R/W (J2 pin 12)
PA.4 (J10 pin 7)*	E (J2 pin 13)
PA.5 (J10 pin 8)*	RST (J2 pin 15)
PC.4 (J12 pin 7)*	DB4 (J2 pin 5)
PC.5 (J2 pin 8)*	DB5 (J2 pin 6)
PC.6 (J2 pin 9)*	DB6 (J2 pin 7)
PC.7 (J2 pin 10)*	DB7 (J2 pin 8)

*penggunaan pin ini tidak mutlak, dapat diganti dengan cara modifikasi program.

Tabel 1
Hubungan DT-AVR LCMS DT-I/O Graphic LCD 128x64 Yellow Green Backlight.

Hubungkan catu daya +12 VDC pada J2 DT-AVR Low Cost Micro System, pastikan polaritas tegangan yang terpasang tidak terbalik. Selain itu pastikan catu daya yang digunakan mampu menyediakan kebutuhan arus kedua modul. Setelah semua rangkaian dan catu daya terhubung dengan benar, programlah `glcd.hex` ke dalam DT-AVR Low Cost Micro System menggunakan *In-System Programmer* seperti DT-HiQ AVR USB ISP (melalui *port* USB) atau DT-HiQ AVR In-System Programmer (melalui *port* paralel) atau divais *programmer* lain yang memiliki konektor ISP 5x2 standar ATMEL. Selanjutnya aturlah kontras DT-I/O Graphic LCD 128x64 Yellow Green Backlight dengan cara memutar VR 1 hingga didapatkan tampilan yang jelas pada layar *graphic LCD*. Aplikasi pada AN ini juga dapat digunakan untuk *graphic LCD* lain yang memiliki kontroler jenis KS0108 seperti DT-I/O Graphic LCD 128x64 Blue STN Backlight.

Program BMP2HEX.exe digunakan untuk mengubah *file bitmap* gambar yang berukuran 128 *pixel* x 64 *pixel* yang berekstensi `.bmp` menjadi *file* heksa yang berekstensi `.h`. File hasil konversi yang berupa data heksa (berekstensi `.h`) terletak pada direktori yang sama dengan direktori *file bitmap* gambar.



Gambar 3
Program BMP2HEX.exe

Jika ingin mengganti tampilan gambar lakukan langkah-langkah sebagai berikut:

- Pastikan gambar yang akan ditampilkan merupakan *file bitmap* gambar dengan ukuran 128 *pixel* x 64 *pixel* dan berwarna hitam putih.

- Jalankan program BMP2HEX.exe dan bukalah *file bitmap* gambar yang akan digunakan dengan cara memilih “Open BMP File”. Kemudian pilih “Convert BMP to .h” untuk mengkonversi *file bitmap* gambar menjadi *file* heksa berekstensi .h.
- *File* heksa hasil konversi memiliki nama *file* yang sama dengan nama *file bitmap* gambar tetapi memiliki ekstensi .h. *File* hasil konversi terletak pada direktori yang sama dengan direktori *file bitmap* gambar.
- Selanjutnya *copy*-kan *file* tersebut (berekstensi .h) ke dalam *folder* yang sama dengan *file* glcd.prj (hasil ekstrak dari *file* AN160.zip).
- Kemudian bukalah *file* glcd.prj menggunakan program CodeVisionAVR. Kemudian tambahkan *file* tersebut ke dalam *project* dengan perintah `#include “<nama file>.h”` seperti berikut ini :

```

25 | *****/
26 | #include "ks0108.h"
27 | #include "Monalisa.h"
28 |
29 | #define tampil " DT-I/O GLCD 128X64 "
30 |
31 | //prototype fungsi
32 | void tampil_IE(void);
33 | void tampil_font(void);
34 | void tampil_gambar(void);

```

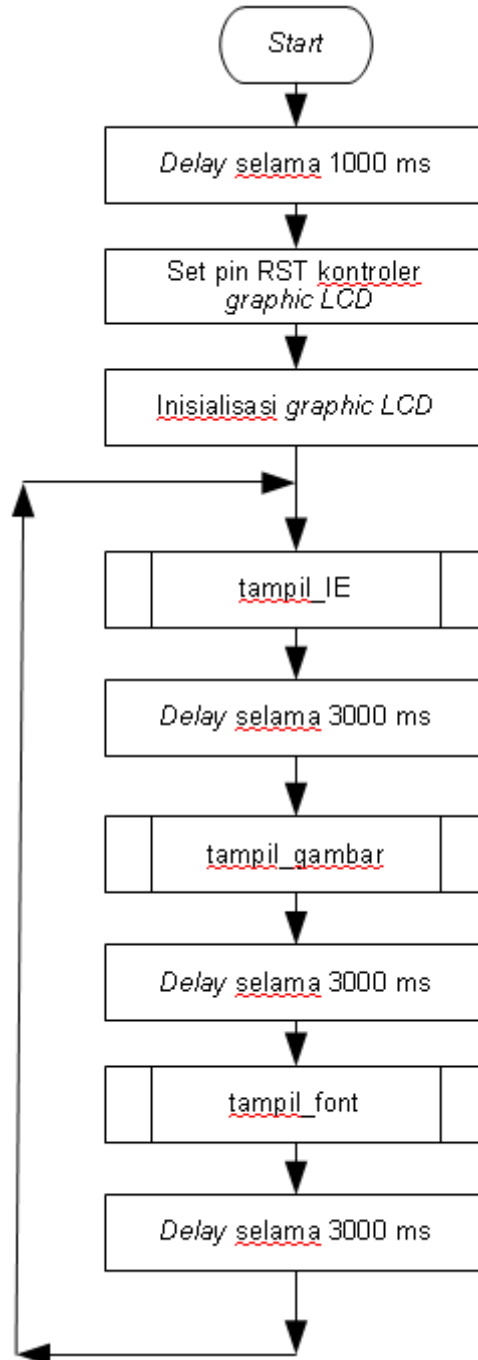
Penambahan file header hasil konversi dari BMP2HEX.exe

Gambar 4
Penambahan File Hasil Konversi ke Program Utama

- Proses penampilan gambar ke layar *graphic LCD* menggunakan rutin “LoadBitmap(<nama variabel file>)”. Misalnya jika nama variabel penyimpanan gambar adalah Monalisa maka penulisan rutusnya adalah “LoadBitmap(Monalisa)”.
- Kemudian lakukan *compile* dengan menekan shortcut F9.

File gambar yang telah diubah menjadi data heksa disimpan pada *flash memory* mikrokontroler. Hal ini dilakukan karena kapasitas *file* gambar yang cukup besar dan kurang memungkinkan jika disimpan dalam *memory* RAM atau EEPROM. Setelah catu daya dan kedua modul terhubung dengan benar, programlah *file* glcd.hex ke DT-AVR LCMS menggunakan perangkat *In-System Programmer* seperti DT-HiQ AVR USB ISP atau DT-HiQ AVR In-System Programmer. Kemudian tekanlah tombol *reset* pada DT-AVR LCMS maka pada layar *graphic LCD* akan muncul *teks* dan gambar secara bergantian.

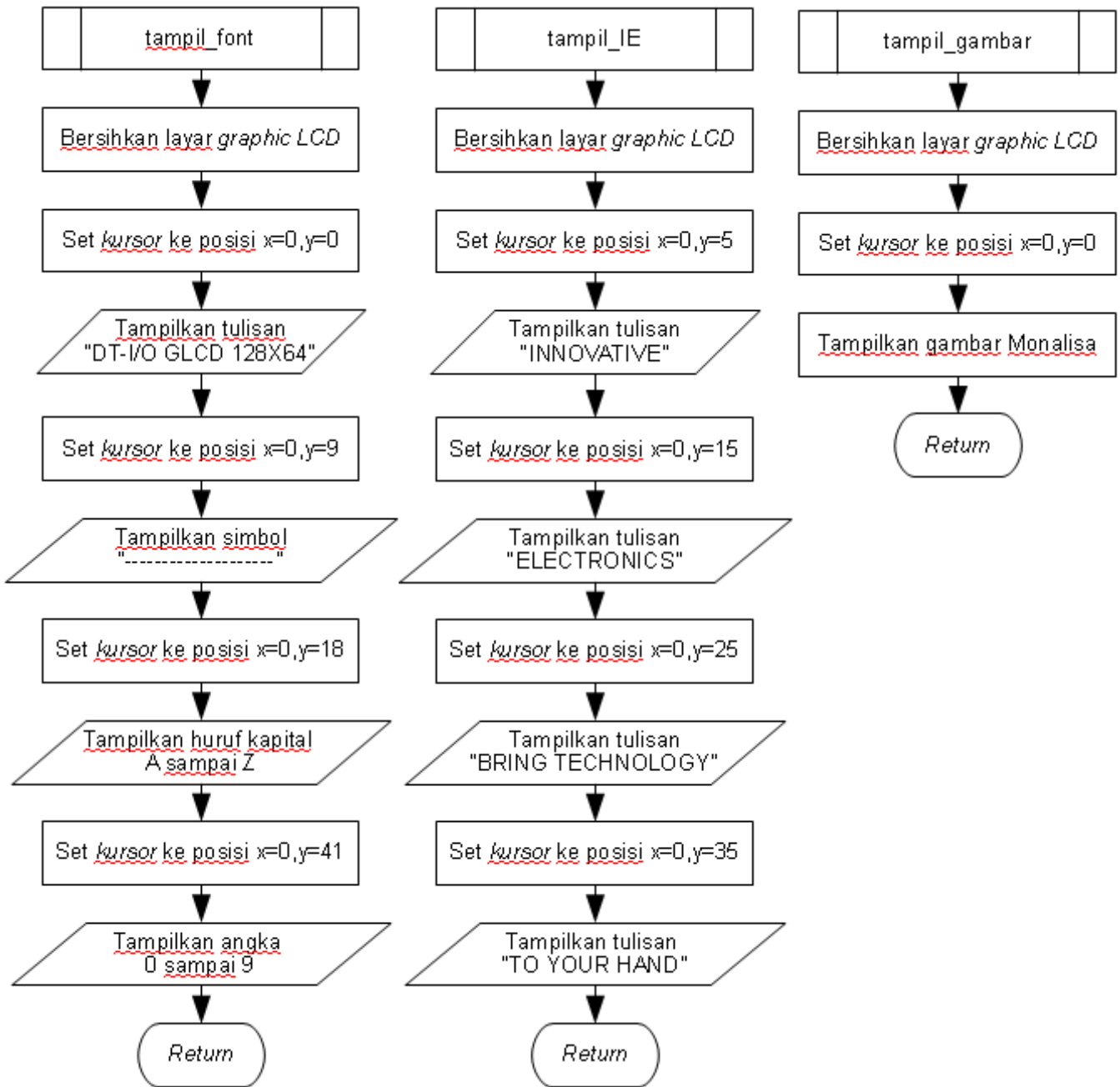
Flowchart program glcd.c secara garis besar adalah sebagai berikut:



Gambar 5
Flowchart Program Utama (glcd.c)

Program glcd.c akan bekerja sebagai berikut:

1. Pertama kali program akan melakukan *delay* selama 1000 ms. Hal ini dilakukan untuk memberikan waktu sejenak kepada rangkaian *graphic LCD* (khususnya kontroler) agar stabil setelah siklus *power-up*.
2. Selanjutnya PA.5 ATmega8535 pada DT-AVR LCMS diatur sebagai *output* (DDRA.5=1) kemudian nilainya diset menjadi 1 (PORTA.5=1).
3. Program melakukan inisialisasi *graphic LCD* melalui beberapa langkah sebagai berikut:
 - Set nilai x (ks0108Coord_x)=0, nilai y (ks0108Coord_y)=0, dan page (ks0108Coord_page)=0.
 - Definisi nilai dot yang aktif berwarna hitam.



Gambar 6
Flowchart Rutin tampil_IE(), tampil_gambar(), dan tampil_font()

7. Program kembali ke nomor 4 dan seterusnya.

Listing program aplikasi ini terdapat pada **AN160.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.*