

PROPELLER DISPLAY

BERBASIS MIKROKONTROLER ATMEGA16



Diajukan Oleh :

Dwi Nurul Saputro

D 400 080 036

JURUSAN TEKNIK ELEKTRO FAKULTAS TEKNIK

UNIVERSITAS MUHAMMADIYAH SURAKARTA

2012

HALAMAN PENGESAHAN

Karya Ilmiah dengan judul “*Propeller Display Berbasis Mikrokontroler ATmega16*” ini diajukan oleh :

Nama : **Dwi Nurul Saputro**

NIM : **D 400 080 036**

NIRM :

Guna memenuhi salah satu syarat untuk menyelesaikan program Sarjana jenjang Pendidikan Strata Satu (S1) pada Jurusan Teknik Elektro Fakultas Teknik Universitas Muhammadiyah Surakarta.

Telah diperiksa dan disetujui pada :

Hari :

Tanggal :

Menyetujui,

Dosen Pembimbing I



(Ir. Abdul Basith, MT)

Dosen Pembimbing II



(Dedi Ary Prasetya, ST)

**PROPELLER DISPLAY
BERBASIS MIKROKONTROLER ATMEGA16**

Dwi Nurul Saputro
FAKULTAS TEKNIK JURUSAN TEKNIK ELEKTRO
UNIVERSITAS MUHAMMADIYAH SURAKARTA
E-mail : dutcha.saputra@gmail.com

ABSTRAKSI

Pada umumnya dalam suatu alat display yang dapat menampilkan karakter dalam jumlah yang banyak dan bergerak, alat display tersebut membutuhkan jumlah LED yang banyak. Misalnya saja penampil dot matrik 5x7, dot matrik ini membutuhkan 35 buah LED untuk membentuk sebuah karakter. Apabila hendak menampilkan 40 karakter dalam sekali tampil, dot matrik ini membutuhkan paling tidak 1.400 buah LED. Jumlah yang cukup banyak, untuk itu dibutuhkan adanya sebuah teknologi display yang dapat menjawab permasalahan tersebut. Persistence Of Vision (POV) merupakan teori yang digunakan dalam propeller display, yang mengungkapkan ketidakmampuan mata manusia dalam melihat pergantian gambar dengan cepat. Teori ini juga sering disebut dengan ilusi atau tipuan mata.

Tujuan penelitian ini adalah membuat sebuah alat display berbasis mikrokontroler yang dapat menampilkan berbagai karakter huruf, angka dan simbol dengan prinsip POV (Persistence Of Vision), sehingga dapat menghemat penggunaan jumlah LED dan terlihat menarik. Propeller display bekerja dengan memutar 20 buah LED yang disusun secara vertikal yang dikontrol menggunakan mikrokontroler. Pada kecepatan tertentu dapat menimbulkan efek berupa tampilan karakter huruf, angka dan simbol. Kontrol putaran dan kontrol karakter dilakukan dengan menggunakan keyboard PS/2. Sedangkan untuk kontrol posisi digunakan optocoupler pada interrupt external. Penyuplaian sumber tegangan pada sistem yang berputar dilakukan dengan cara menggesekkan sumber tegangan dengan benda yang bersifat konduktor yang ditempatkan pada as motor yang berputar. Terdapat empat jalur yang dihantarkan dengan cara ini, yaitu tegangan 12 V, GND, clock keyboard dan data dari keyboard.

Panjang lengan propeller pada perancangan ini adalah 25cm dan kecepatan tertinggi yang dihasilkan motor pemutar adalah 1.320 RPM. Sehingga framerate yang dapat dihasilkan propeller mencapai 22 fps (frame per second). Dengan framerate secepat itu dan ketidak mampuannya mata manusia dalam melihat pergantian yang cepat (POV), maka seolah – olah yang terlihat oleh mata adalah gambar yang diam. Memiliki keliling tampilan 1.57m propeller display dengan 20 LED mampu menampilkan sebanyak 72 karakter 5x7 dalam sekali tampil. Penggunaan jumlah LED yang sangat hemat, bila dibandingkan dengan dot matrik yang harus menggunakan 2.520 buah LED.

Kata kunci : penampil, POV, Persistence Of Vision, LED, mikrokontroler, keyboard PS/2.

1. PENDAHULUAN

Pada umumnya dalam suatu alat *display* yang dapat menampilkan karakter dalam jumlah yang banyak dan bergerak, alat *display* tersebut membutuhkan jumlah LED yang banyak. Misalnya saja penampil dot matrik 5x7, dot matrik ini membutuhkan 35 buah LED untuk membentuk sebuah karakter. Apabila hendak menampilkan 40

karakter dalam sekali tampil, dot matrik ini membutuhkan paling tidak 1.400 buah LED. Jumlah yang cukup banyak, untuk itu dibutuhkan adanya sebuah teknologi *display* yang dapat menjawab permasalahan tersebut.

Propeller display merupakan teknik penampil atau *display* dari efek pancaran cahaya LED yang ditimbulkan dari putaran motor yang cepat. Teknik ini memanfaatkan

kedipan LED yang cepat dan bergantian hingga pada kecepatan tertentu dapat membentuk suatu karakter huruf, angka dan simbol sesuai dengan keinginan pengguna. Penggunaan teknik ini membuat *propeller display* mampu menghemat banyaknya LED yang digunakan. Untuk membentuk sebuah karakter 5x7, alat ini membutuhkan 7 buah LED. Apabila hendak menampilkan 40 karakter dalam sekali tampil, alat ini tetap hanya menggunakan 7 buah LED. Jumlah yang sangat sedikit bila dibandingkan dengan dot matrik yang harus menggunakan 1.400 buah LED.

Persistence Of Vision (POV) merupakan teori yang digunakan dalam *propeller display*. Peter Mark Reget merupakan seorang ilmuwan yang meneliti tentang kemampuan mata manusia dalam menangkap gerak, kemudian dinamakan dengan *Persistence Of Vision* (POV). *Persistence Of Vision* ini menjadi dasar kemampuan mata manusia menangkap gambar. Ketika sebuah seri gambar yang tersusun secara rapi dan ditampilkan secara sekilas dan berurutan, maka efek yang timbul pada otak manusia adalah seri gambar tersebut terlihat hidup.

Penelitian sebelumnya yang berhubungan dengan topic pembahasan dan dijadikan bahan untuk melakukan pengembangan penelitian ini adalah sebagai berikut :

a) Sheikh Rafik Manihar (2012), merupakan seorang mahasiswa Program elektronik dan rekayasa instrumentasi di Chhatrapati Shivaji Institute of Technology, Durg, Chhattisgarh, India. Pada jurnalnya dengan judul *The Power Saving Low Cost Rotating 8 Led Information Display*. Dijelaskan tentang *propeller display* berbasis mikrokontroler AT89C2051, dan 8 LED sebagai *display*-nya. Sheikh menggunakan *sliding contact* untuk jalur *supply* tegangan ke bagian lengan yang berputar. Serta pemanggilan rutin INT0 yang diaktifkan pada mode *rising edge* ketika dijalankan.

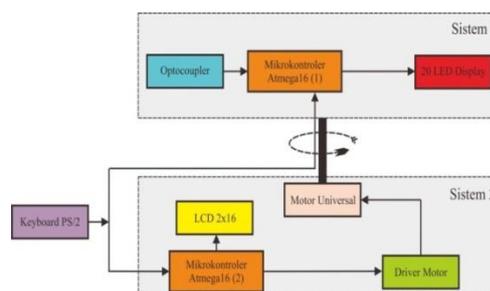
b) Yudha Adi Putra (2011), Jurusan Teknik Elektro Universitas Muhammadiyah Surakarta. Pada Tugas Akhirnya yang berjudul, “Tampilan Karakter Dot Matrik Melalui *Short Message Service* Berbasis Mikrokontroler AT89S52”, digunakannya mikrokontroler dalam mengolah data input yang kemudian menjadi data ASCII. Pengkonversian ini dilakukan untuk mempermudah dalam mengolah data serta dapat menghemat dalam penyimpanan *memory*.

2. METODE PENELITIAN

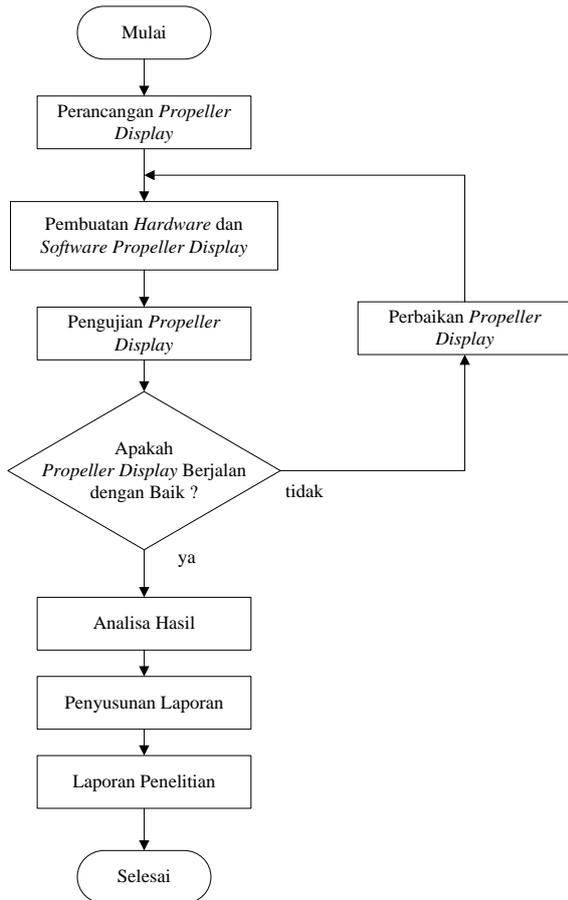
Agar dapat mempermudah penulis dalam melakukan perancangan *hardware* dan *software* pada *propeller display*, maka dibuatlah perancangan blok diagram sistem secara keseluruhan.

Perancangan sistem *propeller display* seperti yang terlihat pada Gambar 3.2. memiliki 2 sistem yang keduanya berbasis mikrokontroler ATmega16. Sistem pertama diletakkan pada bagian lengan yang ikut berputar dan sistem kedua diletakkan pada bagian *body* bawah *propeller display*. Antara sistem 1 dan sistem 2 tidak ada komunikasi yang menghubungkan keduanya. Mengingat sistem 1 yang terletak pada lengan yang berputar akan sulit untuk membuat jalur komunikasi ke sistem 2 yang berada di bawahnya.

Untuk membuat agar kedua sistem bekerja dengan bebarengan maka, *clock* sistem, program dan waktu tunda (*delay*) dalam program dibuat semirip mungkin dan hanya output pengontrolan sistem keduanya saja yang membedakannya.



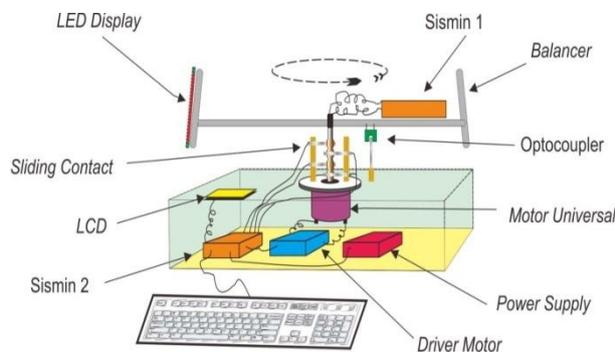
Gambar 1. Blok Diagram Sistem Kerja



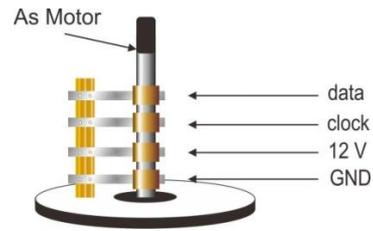
Gambar8. Flowchart Penelitian

2.1. Perancangan Hardware

Pipa aluminium dengan panjang 50cm dan berdiameter 1cm yang digunakan untuk menempatkan rangkaian mikrokontroler (1) dan optocoupler, dan aluminium kotak 1x2cm dengan panjang 14cm yang digunakan untuk menempatkan rangkaian 20 LED.



Gambar 2. Rancangan Mekanik Propeller Display



Gambar 3. Empat Jalur Sliding Contact

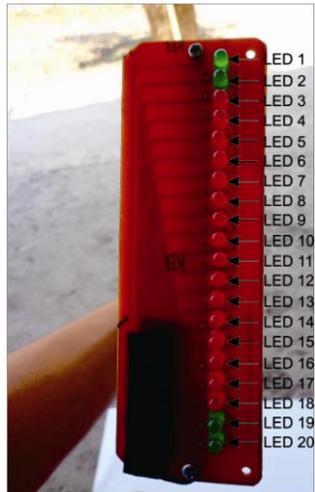
Penyuplaian sumber tegangan ke mikrokontroler ATmega16 (1) dan rangkaian lainnya yang ikut berputar dilakukan dengan cara menggesekkan sumber tegangan dengan benda yang bersifat konduktor yang ditempatkan pada poros atau as *propeller* yang ikut berputar. Akan menjadi masalah ketika penyuplaian menggunakan kabel, karena apabila penyuplaian menggunakan kabel yang terjadi kabel akan terlilit pada *propeller* yang berputar dan kabel bisa saja terputus. Terdapat empat jalur yang dihantarkan dengan cara ini, yaitu tegangan 12 V, GND, *clock keyboard* dan data dari *keyboard*.

Besi tralis dengan tebal 2mm dibentuk kotak dengan panjang 50cm, lebar 50cm dan tinggi 14cm, yang digunakan sebagaiudukan motor universal dan tempat diletakkanya rangkaian mikrokontroler (2), *driver motor*, dan *power supply*.

Pada perancangan *Propeller display* ini menggunakan dua buah sistem yang keduanya menggunakan mikrokontroler ATmega16 sebagai basisnya.

LCD 2x16 berfungsi untuk memberikan informasi kepada pengguna tentang proses program yang sedang berjalan pada mikrokontroler, seperti menu tampilan, setingan tampilan dan lain-lain. Port yang digunakan adalah PORTB pada Mikrokontroler ATmega16 (2).

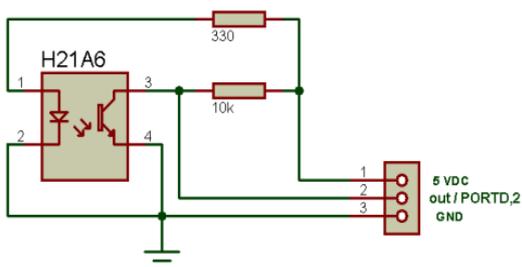
Penampil dari *propeller display* adalah dari 20 LED berukuran 5mm yang disusun vertikal dan dirangkai secara *common katoda*, dengan 16 buah LED berwarna merah dan 4 LED berwarna hijau. LED merah digunakan untuk menampilkan karakter yang ingin di tampilkan dan LED hijau digunakan sebagai garis tepi tampilan.



Gambar 4. LED Penampil

Untuk membentuk sebuah karakter *propeller display* hanya membutuhkan tujuh buah LED saja. Dengan asumsi bahwa sebuah karakter tersusun dari 7x5 LED. Dibandingkan dengan dot matrik, *propeller display* tentu sangatlah hemat dalam penggunaan LED. Dot matrik membutuhkan 35 buah LED untuk membentuk sebuah karakter *propeller display* hanya butuh 7 buah LED, apabila hendak menampilkan 10 karakter pada dot matrik tanpa spasi dibutuhkan 350 buah LED, sedangkan *propeller display* cukup hanya dengan 7 buah LED saja.

Otocoupler pada *propeller display* digunakan sebagai acuan penentu posisi tampilan ketika *propeller display* berputar. Optocoupler juga berfungsi sebagai penghitung putaran motor pemutar *propeller display*. Output dari optocoupler dihubungkan ke PORTD.2 pada mikrokontroler ATmega16 (1) sebagai *interrupt external* (INT.0).



Gambar 5. Rangkaian Optocoupler

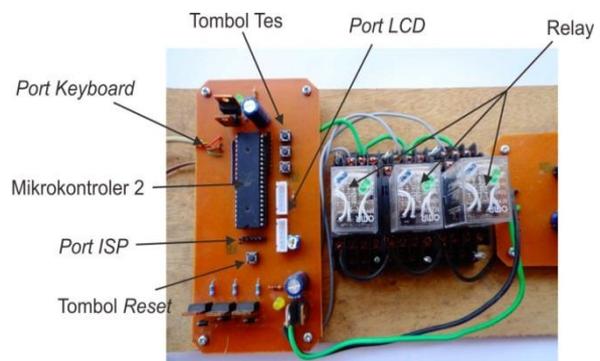
Keyboard PS/2 pada *propeller display* digunakan untuk menginputkan karakter berupa huruf, angka dan simbol-simbol ke mikrokontroler ATmega16 (1) dan mikrokontroler ATmega16 (2). *Keyboard PS/2* juga digunakan sebagai pengontrol sistem *propeller display* seperti mengontrol tampilan LCD 2x16 dan mengontrol kecepatan putar motor universal.

LCD 2x16 berfungsi untuk memberikan informasi kepada pengguna tentang proses program yang sedang berjalan pada mikrokontroler, seperti menu tampilan, setingan tampilan dan lain-lain. Port yang digunakan adalah PORTB pada Mikrokontroler ATmega16 (2).

Driver motor pada *propeller display* tersusun dari transistor, relay dan komponen dasar lainnya. Transistor yang digunakan adalah NPN TIP31 dan relay yang digunakan adalah jenis relay QPDT atau disebut juga dengan relay 4PDT.

Transistor TIP31 pada *driver motor* berfungsi seperti saklar yang dapat memutus dan menghubungkan arus dan tegangan pada relay berdasarkan perintah dari mikrokontroler yang masuk pada kaki basis transistor. PORT yang digunakan adalah PORTA.0, PORTA.1 dan PORTA.2.

Output dari relay digunakan untuk mengalirkan arus listrik ke motor universal. Dioda pada relay digunakan untuk membuang muatan yang tersisa sesaat ketika relay di "off" kan. Karena salah satu sifat lilitan yaitu dapat menyimpan arus sekalipun hanya sementara.

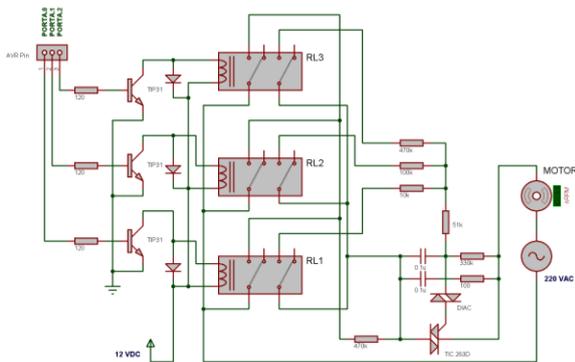


Gambar 6. *Driver Motor Propeller Display*

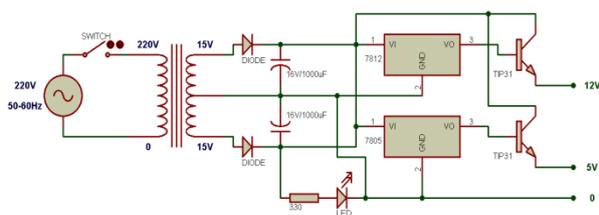
Output relay tidak langsung dihubungkan dengan motor. Karena motor yang dikontrol adalah motor universal yang memiliki RPM tinggi, maka pada perancangan ini pengontrolan dilakukan dengan cara *rheostat*, dengan menambahkan tahanan pada rotor dengan nilai yang berbeda serta penambahan rangkaian DIAC dan TRIAC sebagai pengasut tegangan masuk pada motor.

Output dari relay digunakan untuk mengalirkan arus listrik ke motor universal. Terdapat tiga buah relay yang digunakan sebagai pengontrol kecepatan putar motor. Dioda pada relay digunakan untuk membuang muatan yang tersisa sesaat ketika relay di “off” kan. Karena salah satu sifat lilitan yaitu dapat menyimpan arus sekalipun hanya sementara.

Power supply pada *propeller display* ini berkemampuan 1 Amper dan memiliki dua buah output tegangan, yaitu tegangan 5 V dan tegangan 12 V. Tegangan 5 V digunakan untuk menyuplai 2 buah mikrokontroler, 20 LED dan LCD 2x16. Sedangkan tegangan 12 V digunakan untuk menyuplai relay.



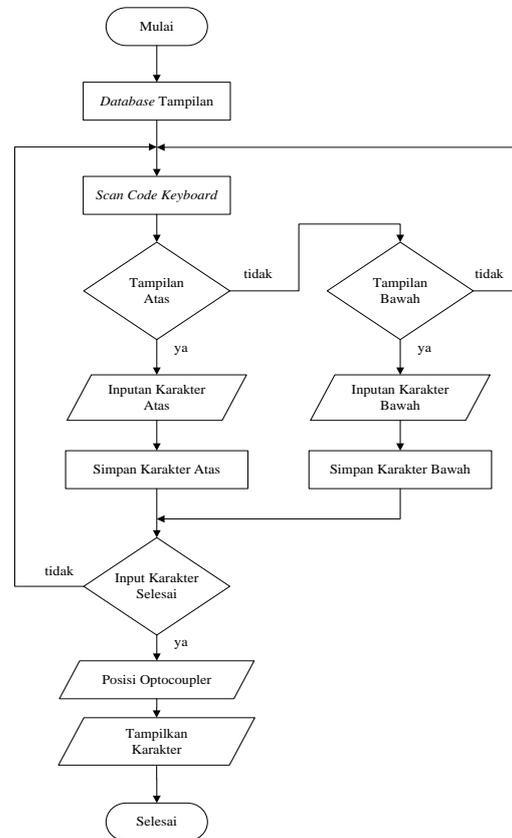
Gambar 3.13. Rangkaian *Driver Motor*



Gambar 3.14. Rangkaian *Power Supply*

2.2. Perancangan *Software*

Perancangan program *propeller display* diawali dengan membuat alur program (*flowchart*) kemudian penulis menjadikannya sebagai acuan dalam pembuatan program proyek tugas akhir ini.



Gambar 7. *Flowchart* Program Tampilan

3. HASIL PENELITIAN DAN ANALISA

Berikut hasil tampilan pada *propeller display* dengan tampilan “Dwi Nurul Saputro” pada baris atas tampilan dan “NIM: D400 080 036” pada baris bawah tampilan.



Gambar 9. Hasil Tampilan *Propeller Display*



Gambar 10. Tampilan LCD 2x16

Untuk menampilkan karakter pada *propeller display* dapat dilakukan dengan langkah-langkah sebagai berikut.

1. Sistem dihidupkan dengan menekan saklar “on” kemudian pada LCD 2x16 akan menampilkan menu “Start” dan “Set Tampilan”.
 - a) Apabila menu yang dipilih adalah “Start” maka yang akan ditampilkan pada LED penampil adalah tampilan *default* yaitu karakter “Dwi Nurul Saputro” pada baris atas tampilan dan “NIM: D400 080 036” pada baris bawah tampilan.
 - b) Apabila menu yang dipilih adalah “Set Tampilan” maka LCD akan menampilkan sub menu “Set Tampilan”.
2. Sub menu dari “Set Tampilan” adalah menu “Atas”, “Bawah” dan “Tampil”. ”Atas” digunakan untuk memasukkan karakter pada tampilan atas LED penampil, “Bawah” digunakan untuk memasukkan karakter tampilan bawah pada LED penampil, sedangkan “Tampil” digunakan untuk menyimpan karakter yang telah dimasukkan dan selanjutnya akan ditampilkan pada LED penampil.

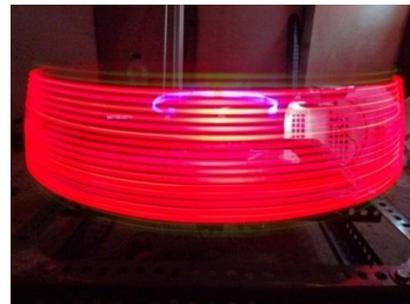
3. Memasukkan karakter tampilan pada *propeller display*. Misalnya saja karakter “Dwi Nurul Saputro” pada tampilan atas dan “NIM: D400 080 036” pada tampilan bawah. Untuk menyimpan karakter dapat dilakukan dengan menekan tombol “Ok” atau tombol “F7”.
4. Untuk mengaktifkan motor pemutar, dilakukan dengan menekan tombol “F1” untuk kecepatan rendah, “F2” untuk kecepatan sedang dan “F3” untuk kecepatan tinggi. Untuk menghentikan motor dilakukan dengan menekan tombol “F4”.

Terdapat beberapa tombol pada *keyboard PS/2* yang oleh penulis difungsikan sebagai kursor dan sebagai pengontrol kecepatan motor pemutar *propeller display*. Tombol-tombol yang dimaksud dapat dilihat pada Tabel 1.

Tabel 1. Tombol Pengontrol dan Kursor

Tombol	Fungsi
F1	Speed 1 Motor
F2	Speed 2 Motor
F3	Speed 3 Motor
F4	Motor Off
F5	Kursor Up
F6	Kursor Down
F7	Ok
F8	Cancel

3.1. Pengujian LED Display

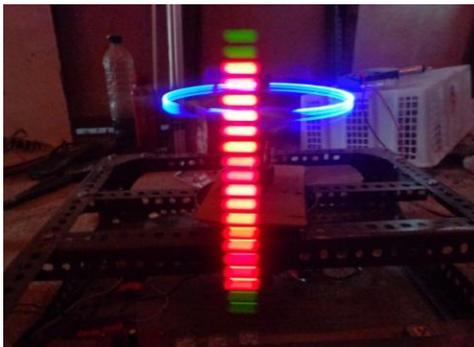


Gambar 11. Tampilan Pengujian 20 LED Display

Pengujian LED *display* dilakukan untuk mengetahui apakah ke20 LED penampil berfungsi dengan baik. Pengujian ini dilakukan dengan cara memberikan masukan tegangan pada sistem keseluruhan, kemudian pada mikrokontroler ATmega16 (1) diprogram untuk mengaktifkan ke20 LED dengan cara memberikan logika *high* pada PORTA, PORTB dan PORTC. Kemudian setelah diprogram, motor pemutar diaktifkan untuk berputar.

3.2. Pengujian Optocoupler

Pengujian dilakukan untuk mengetahui posisi pada *propeller* saat berputar. Pengujian dilakukan dengan cara megaktifkan INT0, yang digunakan untuk masukkan optocoupler, kemudian memberi logika *high* pada PORTA, PORTB dan PORTC setelah itu diberi waktu tunda atau *delay* selama 1 ms (*miley second*).



Gambar 12. Tampilan Pengujian Optocoupler

3.3. Pengujian Input Karakter *Keyboard PS/2*



Gambar 13. Pengujian Input Karakter Pada LCD 2x16

Pengujian ini dilakukan dengan memberikan masukan karakter dengan cara mengetikkan karakter melalui *keyboard PS/2*. Tampilan input karakter *keyboard PS/2* lalu ditampilkan pada LCD 2x16.

Pada Gambar 413. dapat dilihat inputan karakter pada baris atas tampilan “Teknik Elektro” dan pada baris tampilan bawah “UMS”. Inputan karakter yang masukkan akan tersimpan. Selanjutnya inputan karakter akan ditampilkan pada 20 LED penampil. Tampilan pada 20 LED penampil dapat dilihat pada Gambar 14.

Pada pengujian ini dapat diketahui bahwa karakter yang diinputkan *keyboard PS/2* ketika sebelum diputar sama dengan tampilan pada LED penampil saat *propleller* diputar.



Gambar 14. Tampilan Pengujian Karakter Pada LED *Display*

3.4. Pengujian Pengaruh Tampilan Terhadap Kecepatan Motor dan Jumlah Maksimal Karakter



Gambar 15. Tampilan LED Penampil Pada *Speed 1*



Gambar 16. Tampilan LED Penampil Pada *Speed 2*



Gambar 17. Tampilan LED Penampil Pada *Speed 3*

Pada *propeller display* ini tersedia 3 opsi kecepatan, kecepatan rendah, sedang dan tinggi. Pada pengujian ini digunakan peralatan pengukur RPM *tachometer digital* berjenis *photo*.

Dari Gambar 15. sampai Gambar 17. merupakan tampilan dari kecepatan yang berbeda. Dengan menggunakan *delay* antar kolom selama 100 us (*micro seconds*), terlihat pada Gambar 15. tampilan terlihat rapat, pada Gambar 16. tampilan terlihat sedikit melebar dan pada Gambar 17. Tampilan terlihat melebar.

Karena kecepatan motor dapat mempengaruhi tampilan, seperti yang terlihat pada Gambar 15. sampai Gambar 17, maka pada pengujian jumlah maksimal karakter dilakukan pada masing-masing kecepatan yang berbeda. Tampilan dari percobaan ini dapat dilihat pada Gambar 18, sampai Gambar 20.



Gambar 18. Tampilan Maksimal Karakter *Speed 1*



Gambar 19. Tampilan Maksimal Karakter *Speed 2*



Gambar 20. Tampilan Maksimal Karakter *Speed 3*

Pada setiap kecepatan dilakukan pengukuran dan didapatkan hasil sebagai berikut.

Tabel 3. Hasil Pengukuran Kecepatan dan Jumlah Maksimal Karakter

Kecepatan	(RPM)	Jumlah Maksimal Karakter
<i>Speed 1</i>	950	94
<i>Speed 2</i>	1.150	82
<i>Speed 3</i>	1.320	70

Ketika dilakukan pengukuran RPM, data yang ditampilkan pada *tachometer digital* sebenarnya tidak tetap dan selalu berubah-ubah. Hal ini dikarenakan kecepatan motor yang tidak konstan. Data yang tertera pada Tabel 4.2. merupakan data yang diambil karena data tersebut yang paling sering muncul pada masing – masing kecepatan.

Dari hasil pengukuran pada Tabel 3. diketahui bahwa semakin tinggi putaran motor, maka semakin sedikit pula jumlah karakter yang dapat ditampilkan. Walaupun jumlah karakter yang dapat ditampilkan lebih sedikit, tapi pada kecepatan inilah didapatkan kualitas tampilan yang terbaik. Karena semakin tinggi putaran maka semakin tinggi pula *framerate* yang dihasilkan. *Framerate* merupakan banyaknya bingkai tampilan dalam setiap detiknya. Berikut perhitungan untuk mengetahui *framerate* dalam msing-masing kecepatan.

$$f = RPM / 60$$

Keterangan:

$f = \text{framerate (frame per second)}$

RPM = *Revolution Per Minutes*

60 = detik dalam satu menit

Dari perhitungan didapatkan hasil *framerate* pada masing – masing kecepatan sebagai berikut.

Tabel 4. *Framerate* Tiap Kecepatan *PropellerDisplay*

Kecepatan	RPM	<i>Framerate</i> (fps)
<i>Speed 1</i>	950	15
<i>Speed 2</i>	1.150	19
<i>Speed 3</i>	1.320	22

Dari Tabel 4.3. dapat dilihat bahwa *framerate* maksimal yang didapatkan hanya 22 fps, bila dibandingkan dengan televisi yang mampu dengan 25 fps, *framerate propeller display* masih kurang. Karena untuk mencapai *framerate* yang tinggi memang harus dibutuhkan motor dengan

kecepatan tinggi dan keseimbangan baling – baling yang terjaga dari kemungkinan adanya getaran saat *propeller* berputar.

Dengan *framerate* yang didapat pada masing – masing kecepatan. Dapat pula dihitung seberapa cepat *propeller* berputar dalam perbandingan jarak dan waktu.

$$v = f . u . 3600$$

Keterangan :

$v =$ kecepatan dalam Km/h

$f = \text{framerate}$

$u =$ keliling dari *propeller display* ketika berputar

3600 = detik dalam setiap jam

Panjang lengan *propeller display* adalah 50 cm. Berarti untuk jari – jarinya $r = 25$ cm.

a) *Speed 1 (15 fps)*

$$\begin{aligned} v &= f . u . 3600 = 15 . 1.57m . 3600 \\ &= 84.780 \text{ m/h} \\ &= 84,78 \text{ km/h} \end{aligned}$$

b) *Speed 2 (19 fps)*

$$\begin{aligned} v &= f . u . 3600 = 19 . 1.57m . 3600 \\ &= 107.388 \text{ m/h} \\ &= 107,39 \text{ km/h} \end{aligned}$$

c) *Speed 3 (22 fps)*

$$\begin{aligned} v &= f . u . 3600 = 22 . 1.57m . 3600 \\ &= 124.344 \text{ m/h} \\ &= 124,33 \text{ km/h} \end{aligned}$$

Dari perhitungan pada *speed 1* sampai *speed 3* dapat dilihat, kecepatan motor dapat mencapai 124,33 km/h. Untuk itu bagian sistem 1 pada baling – baling yang berputar perlu diperiksa sebelum digunakan untuk mencegah lepasnya bagian karena putaran yang cepat.

4. KESIMPULAN

Kesimpulan yang dapat diambil dari Tugas Akhir ini adalah sebagai berikut :

1. *Propeller Display* 20 LED POV dengan diameter tampilan 50 cm, dapat menampilkan sebanyak 72 karakter 5x7 dalam sekali tampil. Penggunaan jumlah LED yang sangat hemat, bila dibandingkan dengan dot matrik yang menggunakan sebanyak 2.520 buah LED.
2. *Framerate* menentukan kualitas tampilan pada *propeller display*, karena semakin tinggi *framerate* yang mampu dihasilkan maka semakin baik pula tampilan yang dihasilkan.
3. Kecepatan Tertinggi yang dihasilkan motor pemutar adalah 1.320 RPM yang mampu menampilkan karakter dalam 22 fps (*frame per second*).
4. Optocoupler selain difungsikan sebagai *scan count* putaran, dapat juga difungsikan sebagai penentu posisi tampilan pada *propeller display*.

5. DAFTAR PUSTAKA

- Rafik Manihar, Sheikh. 2012. *The Power Saving Low Cost Rotating 8 Led Information Display*. *Ijser* 3:1-5
- Putra, Yudha A. 2011. *Tampilan Karakter Dot Matrik Melalui Short Message Service Berbasis Mikrokontroler AT89S52*. Surakarta : Tugas Akhir, Universitas Muhammadiyah Surakarta
- Gici, Nita. 2010. *Embedded system Interfacing ps_2 pc keyboard to AVR ATmega8535 microcontroller*. (<http://nitagici.wordpress.com/2010/06/15/interfacing-ps2-pc-keyboard-to-avr-amicrocontroller/>, diakses 25 Juni 2012 pukul 21.28 WIB)
- Wijatmiko, Titis. 2007. *Rancang Bangun Alat Pengatur Motor Universal Pada Sewing Machine Motor*. Semarang : Tugas Akhir, Universitas Negeri Semarang
- Bejo, Agus. 2008. *C & AVR Rahasia Kemudahan Bahasa C dalam Mikrokontroler ATmega 8535*. Yogyakarta : Graha Ilmu
- Budiharto, Widodo. 2011. *Aneka Proyek Mikrokontroler*. Yogyakarta : Graha Ilmu