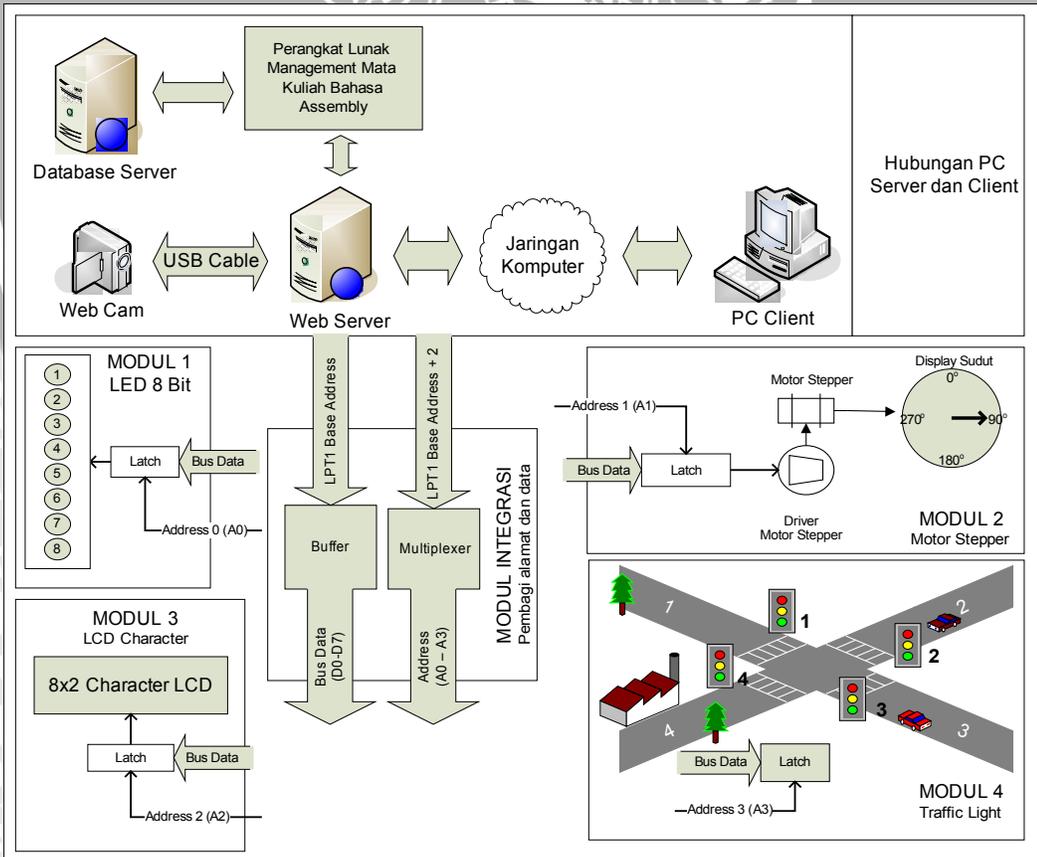


## BAB IV PERANCANGAN DAN PEMBUATAN ALAT

Perancangan dan realisasi alat bantu pembelajaran mata kuliah bahasa *Assembly* berbasis web meliputi perancangan dan pembuatan perangkat keras serta perancangan dan pembuatan perangkat lunak.

### 4.1. Perancangan Sistem Secara Keseluruhan

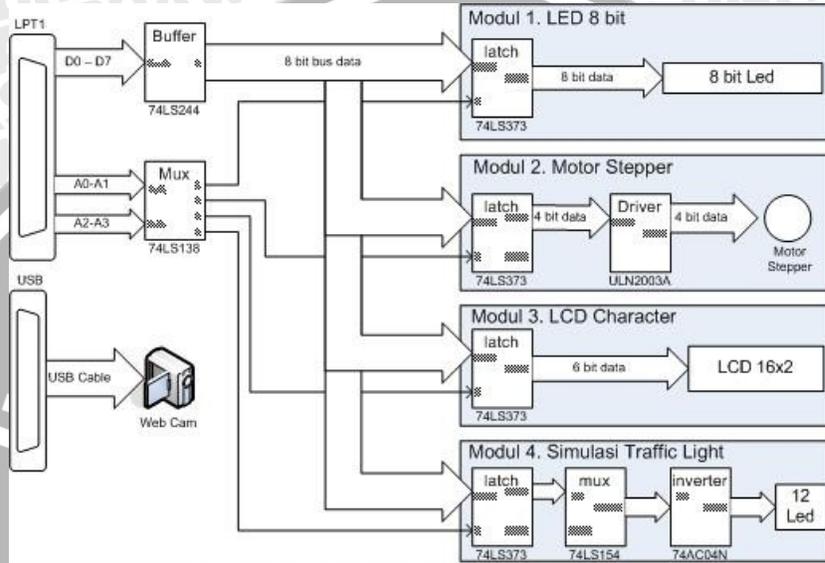
Sistem secara keseluruhan terdiri dari 4 modul standard pembelajaran pengaksesan perangkat keras yang terkoneksi dengan *port* paralel *server*. Penggabungan dan pengawasan ke 4 sistem modul tersebut dilakukan lewat *webcam* yang akan memberikan umpan balik kondisi modul-modul tersebut dari hasil kompilasi program *assembly user* lewat antarmuka *web*. Gambar 4.1 menunjukkan diagram sistem keseluruhan.



Gambar 4.1 Diagram Sistem Keseluruhan

4.2. Perancangan Perangkat Keras

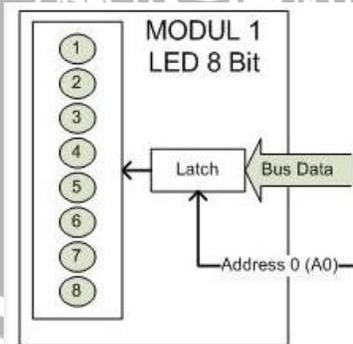
Dari diagram sistem keseluruhan dalam Gambar 4.1, terdapat 4 modul utama atas sistem keseluruhan. Hal ini dapat digambarkan dalam Diagram Sistem Perangkat Keras dalam Gambar 4.2.



Gambar 4.2 Diagram Sistem Perangkat Keras

4.2.1 Perancangan Modul 1 LED 8 Bit

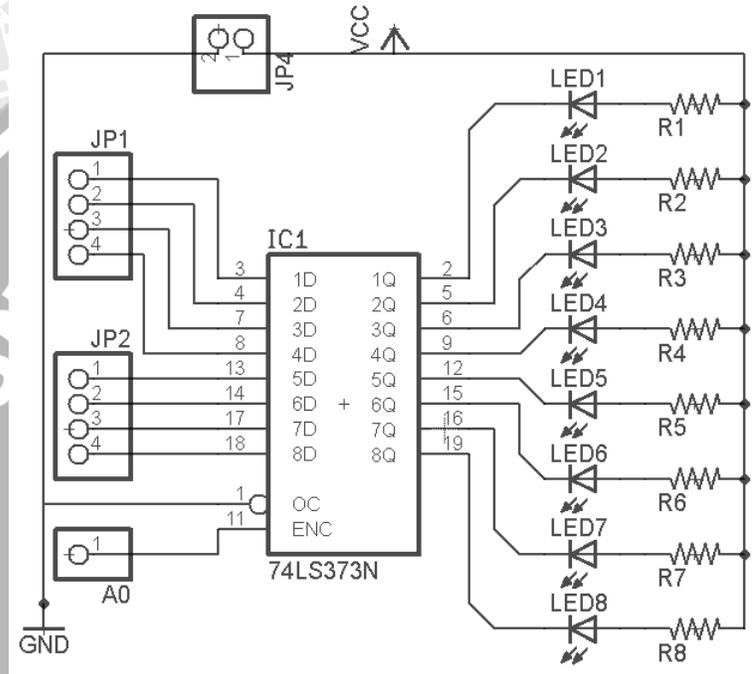
Pada bagian Modul 1, terdapat rangkaian LED 8 Bit. Terdiri dari 8 buah LED yang menunjukkan data binary dari kondisi bus data LPT1 (D0 – D7) sesuai Gambar 4.3.



Gambar 4.3 Modul 1. LED 8 bit

Data dari modul penggabung disalurkan lewat *bus data* dan ditahan oleh *latch*. Dengan trigger dari Address 0 (A0) maka data dari latch akan dikirimkan ke LED sesuai dengan bit-bit data masukan dari Bus Data.

Dari perancangan diatas, maka dapat digambarkan rangkaian LED 8 Bit dalam Gambar 4.4.



**Gambar 4.4** Rangkaian. LED 8 bit

R1 sampai dengan R8 adalah resistor pembatas arus yang berfungsi untuk membatasi arus yang melewati LED. Besar resistor pembatas arus dapat dihitung dengan persamaan sebagai berikut :

$$\begin{aligned}
 V_R &= V_{CC} - V_{OL} - V_{LED} \\
 V_R &= R \cdot I_R = R \cdot I_{LED} \\
 R &= \frac{V_R}{I_{LED}} \\
 R &= \frac{V_{CC} - V_{OL} - V_{LED}}{I_{LED}}
 \end{aligned}
 \tag{4.1}$$

Sesuai datasheet, latch tersebut memiliki karakteristik  $V_{OH(min)}$  sebesar 2.4 volt dan  $V_{OL(max)}$  sebesar 0.4 volt untuk  $V_{CC}$  sebesar 5 volt. Dengan kebutuhan tegangan maju di dalam LED (biru) sebesar 2.48 volt dan dengan  $I_{OL(max)} = 24$  mA maka digunakan  $I_{R1}$  yang lebih kecil dari  $I_{OL(max)}$  sehingga didapatkan  $I_{R1}=0.02$  A. Dengan menggunakan persamaan 4.1 dan data-data diatas, dapat ditentukan nilai  $R_1$  sampai dengan  $R_8$  sebagai berikut :

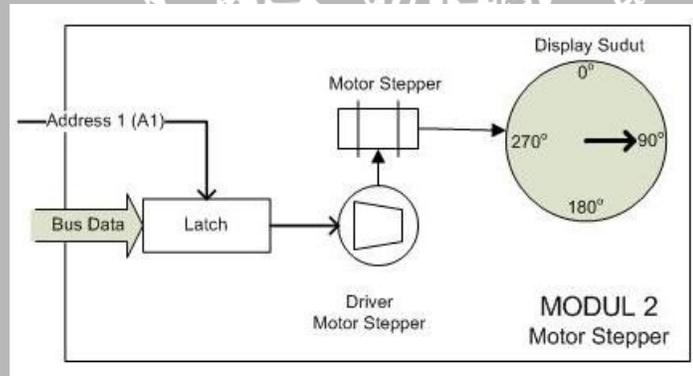
$$R_1 = \frac{V_{CC} - V_{OL} - V_{LED}}{I_{R1}}$$

$$R_1 = \frac{5 - 0.4 - 2.48}{0.02}$$

$$R_1 = 106\Omega \cong 100\Omega$$

#### 4.2.2 Perancangan Modul 2. *Motor Stepper*

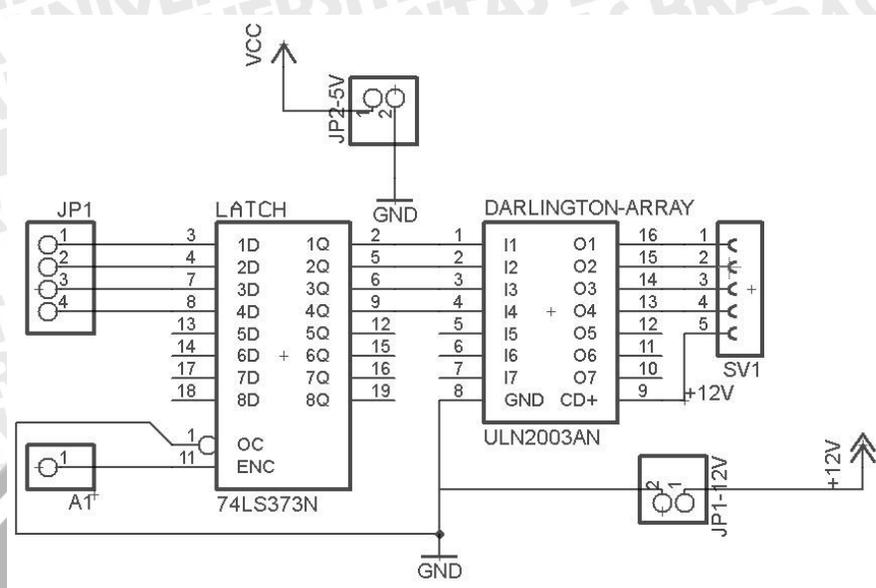
Pada bagian Modul 2. terdapat rangkaian *motor stepper* sederhana. Modul ini mengajarkan metodologis pemanfaatan algoritma urutan dalam bahasa assembly agar sebuah panah bergerak sekian derajat searah atau berlawanan arah dengan jarum jam. Hal ini dapat digambarkan dalam Gambar 4.5.



Gambar 4.5. Modul 2 Motor Stepper

Data dari modul penggabung disalurkan lewat Bus Data dan ditahan oleh Latch. Dengan trigger dari Address 0 (A0) maka data dari latch akan dikirimkan ke Driver motor stepper. Driver motor stepper menggerakkan motor stepper sesuai dengan sekuensial data dari latch sehingga didapatkan perputaran sudut yang diinginkan.

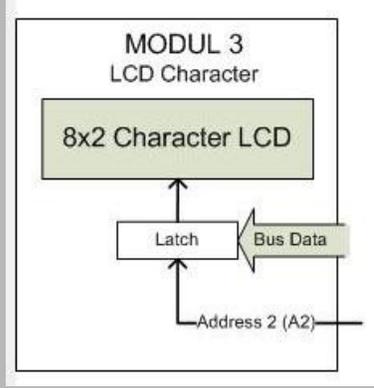
Dari perancangan diatas, maka dapat digambarkan rangkaian *Driver Motor Stepper* dalam Gambar 4.6.



Gambar 4.6 Rangkaian. Driver Motor Stepper

### 4.2.3 Perancangan Modul 3. LCD Character

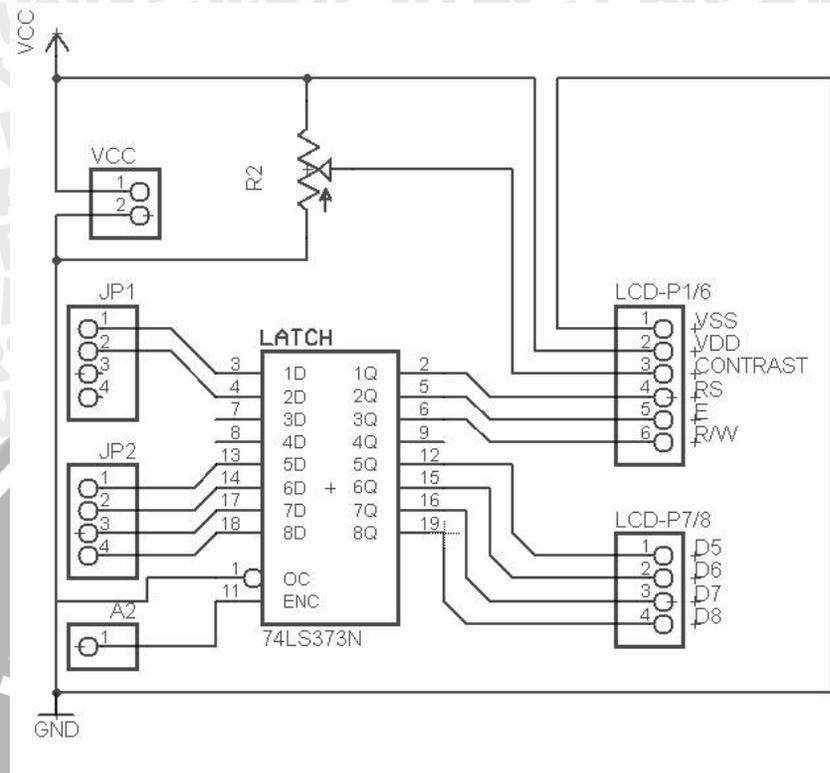
Sistem Modul 3. Rangkaian LCD *Character*, merupakan modul yang berisi penampil karakter. Kemampuan bahasa assembly yang diasah dalam hal ini adalah pengaksesan LCD sebagai bagian yang tak terpisahkan dari sistem microcontroller. Diagram modul 3 LCD *Character* dapat digambarkan dalam Gambar 4.7.



Gambar 4.7. Modul 3. LCD Character

Data dari modul penggabung lewat bus data dikirimkan dan ditahan oleh latch. Data dikirimkan dari latch ke LCD Character 8x2 ketika address 2 (A2) bernilai 1.

Dari perancangan diatas, maka dapat digambarkan Rangkaian Modul 3 LCD *Character* dalam Gambar 4.8.

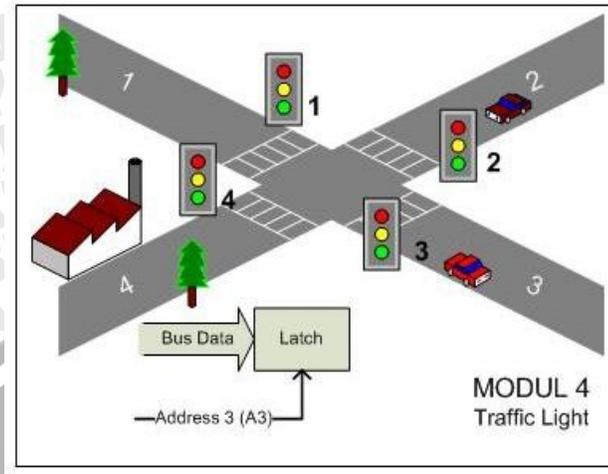


**Gambar 4.8** Rangkaian Driver LCD Character 8 Bit

Dalam pemrograman LCD, dikenal 2 mode pemrograman, yaitu mode 8 bit dan mode 4 bit. Pada mode 8 bit pemrograman LCD, diperlukan 8 bit data (D0-D7) serta 2 bit operasional (RS dan R/W). Sehingga modul 3 LCD Character ini dibatasi hanya menggunakan mode 4 bit operasional saja. Dimana hanya diperlukan 4 bit data (D0 – D3) dan 2 bit operasional (RS dan R/W).

#### 4.2.4 Perancangan Modul 4. Simulasi Traffic Light

Sistem Modul 4. Rangkaian Simulasi *Traffic Light*, merupakan modul yang dapat menggambarkan kondisi *traffic light* jalan dengan mengakses warna lampu merah, kuning dan hijau pada perempatan tertentu dengan pewaktuan dan urutan tertentu. Hal ini dapat digambarkan dalam Gambar 4.9.



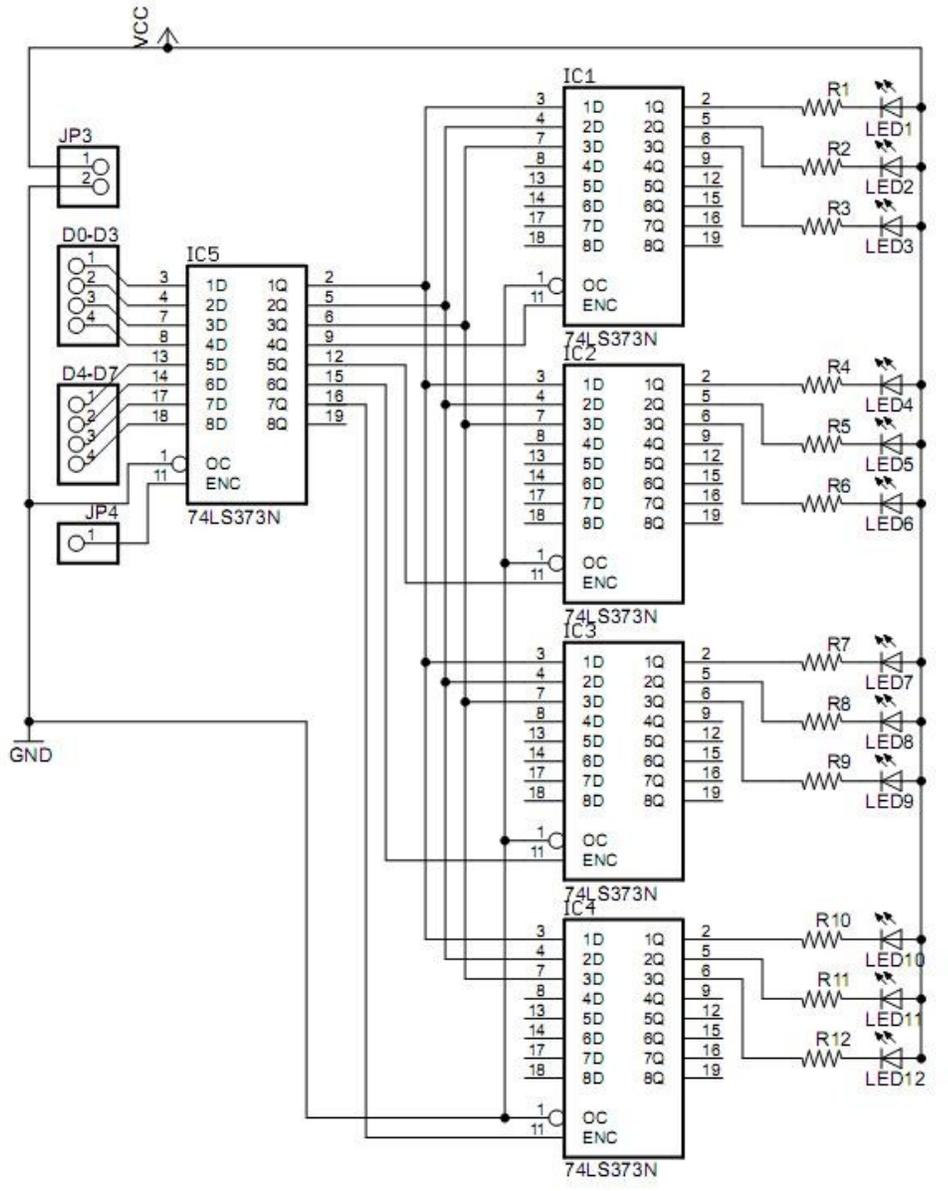
Gambar 4.9. Modul 4. Simulasi *Traffic Light*

Dari gambar Modul 4 Simulasi *Traffic Light*, dibutuhkan 4 perempatan jalan dimana disetiap perempatan jalan terdapat 3 lampu, yaitu merah, kuning dan hijau. Sehingga dibutuhkan 12 LED untuk merepresentasikan hal ini sebagai keluaran simulasi *traffic light*. Kombinasi logika sinyal masukan *traffic light* dapat digambarkan dalam tabel 4.1

Tabel 4.1 Kombinasi Logika Sinyal Masukan *Traffic Light*

Perempatan	D6	D5	D4	D3	Warna	D2	D1	D0
Perempatan 1	0	0	0	1	Merah	0	0	1
Perempatan 2	0	0	1	0	Kuning	0	1	0
Perempatan 3	0	1	0	0	Hijau	1	0	0
Perempatan 4	1	0	0	0	--		X	X

Dengan kebutuhan keluaran 12 bit serta kondisi logika kombinasi masukan hanya 4 bit, maka digunakan 4 buah latch untuk setiap perempatan dan 1 buah latch yang berfungsi sebagai kontroller utama perempatan, sehingga diperlukan 5 buah latch untuk modul *Traffic Light*. Dari perancangan diatas, maka dapat digambarkan rangkaian Modul 4. simulasi *Traffic Light* dalam Gambar 4.10.



Gambar 4.10 Rangkaian. Simulasi Traffic Light

Dengan menggunakan persamaan 4.1 dan data dari *datasheet*, maka dapat ditentukan nilai  $R_1$  sampai dengan  $R_{12}$  sebagai berikut:

$$R_{merah} = \frac{V_{CC} - V_{OL} - V_{LED}}{I_{LED}}$$

$$R_{merah} = \frac{5 - 0.4 - 1.63}{0.02}$$

$$R_{merah} = 148\Omega \approx 150\Omega$$

$$R_{kuning} = \frac{V_{CC} - V_{OL} - V_{LED}}{I_{LED}}$$

$$R_{kuning} = \frac{5 - 0.4 - 2.1}{0.02}$$

$$R_{kuning} = 125\Omega \approx 120\Omega$$

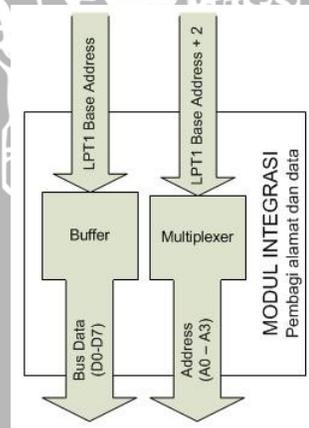
$$R_{hijau} = \frac{V_{CC} - V_{OL} - V_{LED}}{I_{R1}}$$

$$R_{hijau} = \frac{5 - 0.4 - 2.18}{0.02}$$

$$R_{hijau} = 121\Omega \approx 120\Omega$$

#### 4.2.5 Perancangan Modul Penggabung

Pada bagian Modul Penggabung, terdapat pembagi alamat dan data dari *Port LPT1 (Port paralel)* server. Dengan fasilitas alamat sebuah *port* paralel lebih dari 1, maka dapat digunakan alamat lain dari *port* paralel untuk menentukan alamat modul yang akan diberi data (D0-D7) dari sebuah *base address port paralel*. Prinsip inilah yang digunakan dalam mekanisme di dalam modul penggabung yang dapat digambarkan dalam Gambar 4.11.



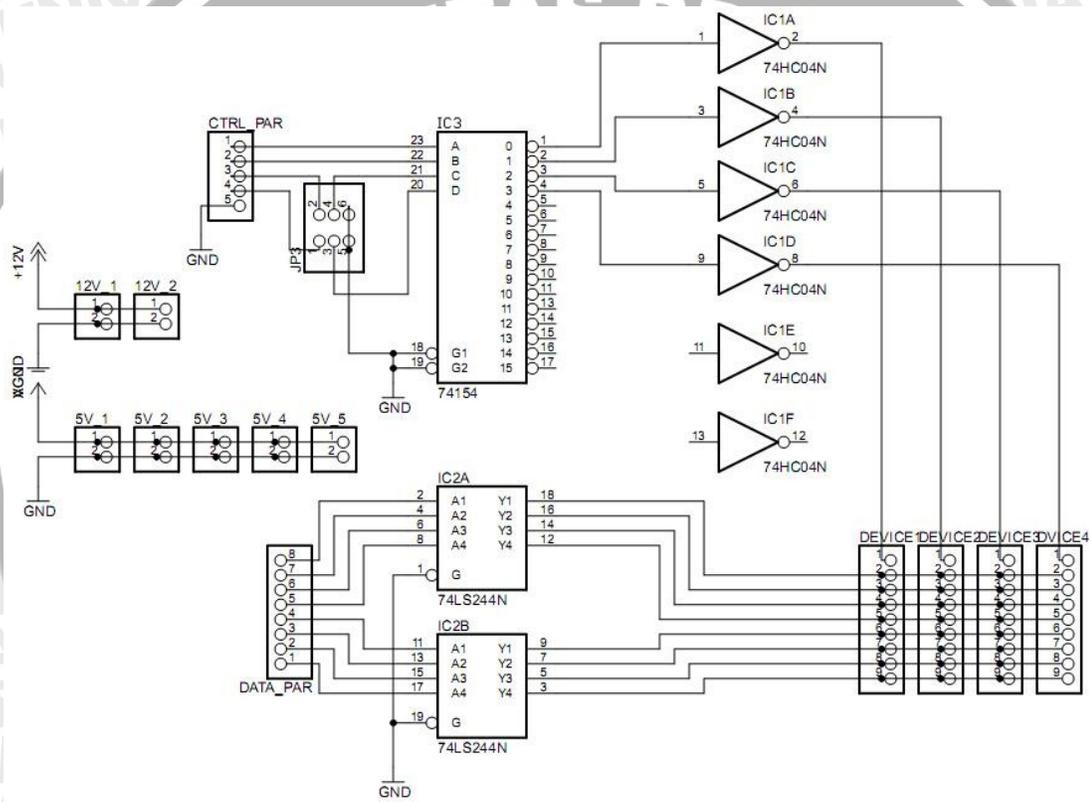
Gambar 4.11. Modul Penggabung

Pin yang digunakan untuk pengaturan alamat melalui multiplexer adalah pin port kontrol (BASE+2) Pin 1 (bit ke-0) dan Pin 14 (bit ke-1). Hal ini bisa digambarkan dalam Tabel 4.2 :

Tabel 4.2 Tabel Pengalamatan Modul

Data Pin		Data Port Kontrol			Modul
P.14	P.1	D.1	D.0	HEX	
0	0	1	1	0x03	Modul 1, 8-Bit LED
0	1	1	0	0x02	Modul 2, Motor Stepper
1	0	0	1	0x01	Modul 3, LCD Character
1	1	0	0	0x00	Modul 4, Simulasi Traffic light

Dari perancangan diatas, maka dapat digambarkan rangkaian modul penggabung alat dalam Gambar 4.12.



Gambar 4.12 Rangkaian Modul Penggabung