

PROTOTYPE PENCAMPUR WARNA OTOMATIS BERBASIS MIKROKONTROLER AT89S51

Umi Fadlilah¹, Fatah Yasin Al Irsyadi², Aryanto Hari Pratikto³
^{1,2,3}Jurusan Teknik Elektro, Fakultas Teknik, Universitas Muhammadiyah Surakarta
Jl. A. Yani Tromol Pos 1 Pabelan Kartasura 57102 Telp 0271 717417
Email: ifadtofa@gmail.com

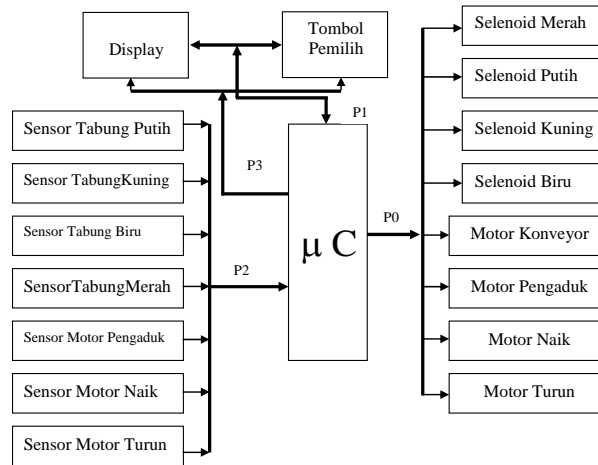
Abstrak

Alat pencampur warna otomatis sesungguhnya merupakan peralatan elektronika yang sangat dibutuhkan terutama oleh pengusaha di bidang pengoplosan (pencampuran) warna, seperti dalam mencampur warna cat atau membuat warna baru dari berbagai variasi warna. Biasanya para pemilik toko cat masih menggunakan alat manual dalam mencampur warna. Prinsip kerja dari prototype pencampur warna otomatis ini didasarkan dari proses pencampuran warna pada empat buah tabung warna pokok, yaitu putih, kuning, biru, dan merah. Keempat warna tersebut memungkinkan terjadinya warna turunan yang variatif. Pada alat ini disediakan 10 tombol untuk contoh memilih campuran warna. Gelas pencampur dapat dijalankan dengan dukungan konveyor dan dilengkapi dengan pengaduk warna serta solenoid untuk mengatur pembukaan maupun penutupan kran cat. Penggerak dari alat pencampur warna cat otomatis ini adalah mikrokontroler AT89S51 yang merupakan keluarga MCS-51 dan mempunyai kemampuan stabil dalam menjalankan proses sesuai target yang diharapkan. Pencampuran warna cat menggunakan perbandingan menurut prosentase yang ditetapkan dari hasil pengujian, sehingga setelah menekan salah satu tombol, maka akan mewujudkan suatu warna yang diinginkan, dalam hal ini ada 10 warna turunan dari kombinasi warna cat utama.

Kata kunci : Campuran; Mikrokontroler; Prosentasi; Tombol Pemilih; Warna

Pendahuluan

Pencampur warna ini menggunakan mikrokontroler sebagai komponen inti, karena dengan mikrokontroler hanya dibutuhkan sedikit *hardware*, walaupun menggunakan *software* atau bahasa pemrograman yang cukup rumit. Alat ini menggunakan beberapa komponen pendukung yang digabungkan dengan mikrokontroler, sehingga dapat berfungsi sebagai pencampur warna, diantaranya saklar untuk memilih warna yang akan dicampur, *solenoid* sebagai kran atau katub untuk cairan warna yang akan dicampur, motor DC untuk konveyor dan turun naik pengaduk serta sebagai pengaduk, seven segment untuk menampilkan kode angka pilihan dan mikro *switch* untuk saklar otomatisnya. Hubungan dari komponen tersebut dapat dilihat pada Gambar 1. Blok diagram pencampur warna tersebut terdiri atas mikrokontroler, *display*, *solenoid*, motor, tombol pemilih dan sensor berupa *micro switch*. Mikrokontroler sebagai pengontrol utama semua komponen yang terhubung, *seven segment display* sebagai tampilan angka yang mewakili warna dasar, *solenoid* sebagai kran atau katub cairan warna, motor berfungsi sebagai penggerak konveyor, pengaduk dan pengontrol naik turun pengaduk, tombol pemilih untuk memasukkan angka yang mewakili warna dasar, serta sensor *micro switch* untuk pengontrol otomatis motor dan *solenoid*.



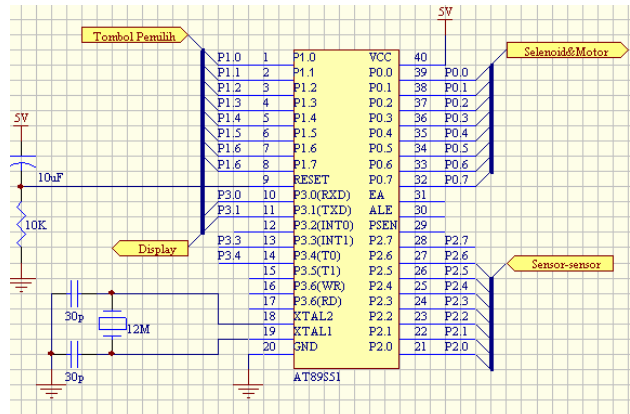
Gambar 1. Blok Diagram Pencampur Warna

Port 1 difungsikan ganda, yaitu sebagai *input* untuk tombol pemilih dan *output* untuk *display*, karena menghemat komponen yang dipakai (jadi, tidak perlu menambah PPI8255). Ide penggunaan ganda Port 1 muncul karena ada metode *multiplexer*, sehingga komponen tersebut diakses secara bergantian selama waktu tertentu. Jika Port 1 sebagai *input* tombol pemilih, maka kondisi Port 1 harus dibuat *high* dengan memberi data “1” ke Port 1, kemudian *digit select* untuk *display* dimatikan dulu baru kemudian tombol dibaca. Jika sebagai *output*, maka data *display* langsung dikeluarkan.

1. Perancangan Hardware

a. Mikrokontroler AT89S51

Mikrokontroler AT89S51 berfungsi sebagai komponen utama yang bertugas menerima masukan dari tombol pemilih, menampilkan masukan tombol, mengolahnya, mengeluarkan hasil olahan tombol, serta membaca sensor pengontrol *solenoid* dan motor.



Gambar 2. Rangkaian AT89S51 pada Prototype Pencampur Warna

AT89S51 membutuhkan tambahan rangkaian eksternal agar dapat bekerja dengan baik, yaitu rangkaian osilator dan rangkaian *reset*. Rangkaian osilator sebagai pembangkit gelombang dan rangkaian *reset* untuk menghidupkan kembali setelah mati pada pin-pin AT89S51 yang berada pada kondisi awal (semua *port* pada kondisi *high*). Rangkaian osilator dibangun dari kombinasi kristal 12 MHz dan dua buah kapasitor keramik 30 pF yang dihubungkan dengan pin 18 dan 19, sedangkan untuk rangkaian *reset* dibangun oleh kapasitor 10µF dan resistor 10 KΩ yang dihubungkan dengan pin 9.

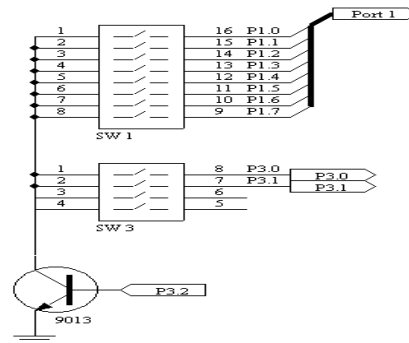
Pada Gambar 2 juga dapat dilihat hubungan *port* dengan komponen pendukung, *port* 0 untuk kontrol *solenoid* dan motor, *port* 1 untuk mengambil data masukan dari tombol pemilih dan untuk mengeluarkan data ke *display seven segment*, *port* 2 untuk membaca sensor pengontrol *solenoid* dan motor, *port* 3 untuk kontrol *display* dan tombol pemilih serta sebagian digunakan sebagai tombol pemilih.

b. Tombol Pemilih

Tombol pemilih sebagai pemilih jenis warna yang akan dicampur sehingga menghasilkan warna tertentu yang terdiri atas sepuluh macam warna pilihan yang sudah ditentukan. Pada perancangan ini digunakan tombol jenis *push button*, yaitu tombol yang hanya bisa ditekan tapi tidak mengunci karena hanya

dibutuhkan picu sesaat sebagai *input* mikrokontroler untuk memilih kombinasi warna yang sudah ditentukan sebelumnya. Gambar 3 adalah rangkaian tombol pemilih,

- Kaki 1 sw1 =tombol no 1
- Kaki 2 sw1 =tombol no 2
- Kaki 3 sw1 =tombol no 3
- Kaki 4 sw1 =tombol no 4
- Kaki 5 sw1 =tombol no 5
- Kaki 6 sw1 =tombol no 6
- Kaki 7 sw1 =tombol no 7
- Kaki 8 sw1 =tombol no 8
- Kaki1 sw3 =tombol no 9
- Kaki2 sw3 =tombol no10

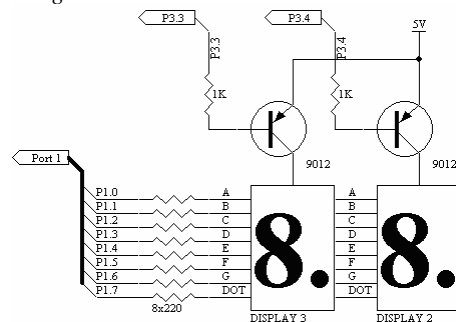


Gambar 3. Rangkaian Tombol Pemilih

Pada Gambar 3, dapat dilihat bahwa *input* saklar dihubungkan pada *port 1* dan sebagian pada *port 3*. Jumlah kombinasi warna yang akan dipilih berjumlah sepuluh buah, sehingga membutuhkan sepuluh buah *port* masukan (delapan *port* pada *port 1* dan dua *port* pada *port 3*). *Common* dari masing-masing saklar dihubungkan menjadi satu dan dikontrol sebuah transistor yang dihubungkan dengan *ground* (aktif *low*) dan dikontrol oleh *port 3.2*. Transistor ini berfungsi untuk mematikan tombol pemilih saat *port 1* mengeluarkan data untuk *display*, karena *port 1* difungsikan ganda sebagai *input* tombol pemilih dan sebagai *output display*.

c. Display Seven Segment

Pencampur warna ini menggunakan tampilan jenis *seven segment* karena dianggap cukup untuk menampilkan angka *input* masukan dan cukup jelas dilihat. Tampilan berjumlah dua digit. *Seven segment* yang digunakan adalah jenis *common anoda*, datanya *low* (0) dan *common*-nya *high* (1). *Seven segment* dihubungkan dengan *port 1* paralel dengan tombol pemilih agar menghemat *port*. Hubungan tersebut menggunakan metode *multiplexer* dan pengaksesannya bergantian dengan jeda waktu tertentu. Gambar 4 adalah rangkaian untuk *seven segment*.

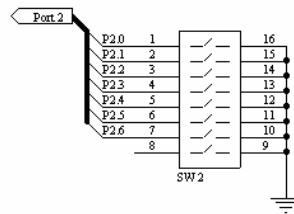


Gambar 4. Rangkaian Seven Segment

Jalur data kedua *seven segment* dihubungkan secara paralel ke *port 1*. Masing-masing *common* atau saklarnya dihubungkan dengan P3.3 untuk digit satu dan P3.4 untuk digit dua. Saklar dibuat dari transistor tipe PNP dengan seri C9012, karena *common* membutuhkan catu +5V. Penyalaan *seven segment* menggunakan mode *scanning* yaitu *seven segment* aktif bergantian, tapi karena cepatnya waktu *scanning*, maka seolah-olah *seven segment* menyala semua. *Seven segment* nyala bila datanya *low* (0), sedangkan untuk mengaktifkan saklar maka basis dari transistor harus di-*low*, sehingga arus akan mengalir dari emitor menuju kolektor (transistor dalam kondisi saturasi). Mematikan *seven segment* dengan cara membuat basis transistor di level *high*, jadi aliran arus akan terputus (transistor dalam kondisi *cut off*).

d. Sensor otomatis

Sensor otomatis berfungsi sebagai kontrol untuk *solenoid* dan motor sesuai dengan kode angka yang dipilih. *Solenoid* berisi cairan warna yang akan dicampur. Motor digunakan untuk konveyor, naik turunnya pengaduk dan untuk motor pengaduk itu sendiri. Sensor yang digunakan berupa saklar yang biasa disebut dengan *limit switch*. Digunakannya *limit switch* karena biayanya lebih murah dari pada *proximity switch* dan pada aplikasi ini masih bisa digunakan saklar yang langsung bersentuhan dengan alat, yaitu gelas tempat pencampuran cairan warna.

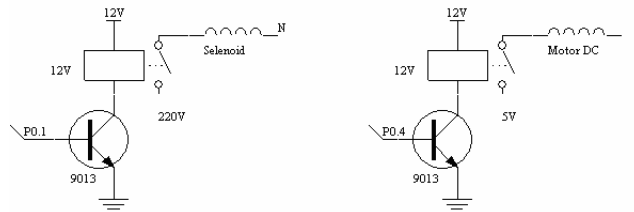


Gambar 5. Rangkaian Sensor Otomatis

Gambar 5 memperlihatkan bahwa sensor dihubungkan dengan *port 2* pada kondisi aktif *low*, jadi, jika sensor tersentuh gelas, maka kondisi pada *port 2* akan *low* (0), tetapi jika tidak tersentuh gelas, kondisi pada *port 2* akan *high* (1).

e. Driver solenoid dan Motor

Driver solenoid dan motor ini terdiri atas sebuah transistor dan *relay*, karena *solenoid* ini mempunyai koil 220 V yang tidak mungkin dikontrol langsung oleh mikrokontroler. Konveyor menggunakan motor DC 12V, sedangkan pengaduk menggunakan motor DC 5V serta untuk naik turunnya pengaduk digunakan motor DC 5V. Perancangan ini menggunakan empat buah *solenoid* untuk katup empat macam warna pokok (merah, putih, kuning, biru) dan empat buah motor DC, jadi dibutuhkan delapan *driver* untuk menggerakkan *solenoid* dan motor tersebut.



Gambar 6. Driver *Solenoid* dan Motor

Gambar 6 adalah driver untuk *solenoid* dan motor yang terdiri atas sebuah transistor dan *relay*. Jika basis transistor mendapat logika 1, maka transistor akan saturasi, sehingga kolektor dan emitor akan terhubung. Hal ini dilanjutkan dengan hubungan salah satu koil *relay* ke *ground* yang mengakibatkan *relay* akan aktif dan *solenoid* juga akan aktif. Jika basis transistor mendapat logika 0, maka transistor akan *cut off*, sehingga memutus aliran *ground* ke koil *relay* sehingga *relay* dan *solenoid* akan mati.

2. Perancangan Software

a. Tombol pemilih

Tombol pemilih berfungsi sebagai *input* untuk memilih kombinasi warna yang akan dicampur. Tombol pemilih terdiri atas sepuluh macam pilihan warna yang sudah ditentukan sebelumnya, agar bisa diolah oleh mikrokontroler, sehingga diperlukan program yang harus diisikan ke mikrokontroler. Perancangan ini menggunakan bahasa pemrograman *Assembly*. Bahasa ditulis dan disimpan dalam ekstensi *.ASM*, kemudian di-*compile* menggunakan ASM51 yang berfungsi untuk membuat file *.HEX*, lalu diisikan ke mikrokontroler. *Listing program* untuk membaca tombol pemilih adalah :

Bagian Pertama

```
scankey: setb    ensw
              mov    r6,#0
              jb     p3.0,p31
              mov    r6,#9
              mov    dsp,r6
              clr    ensw
              ret
p31:         jb     p3.1,port1
              mov    r6,#10
              mov    dsp,r6
              clr    ensw
              ret
```

Lanjut ke Bagian Kedua

Bagian Kedua

```
port1: mov    p1,#0ffh
       nop
       mov    a,p1
       clr    ensw
       mov    r6,#8
lsk:   rlc    a
       jnc    xit
       djnz  r6,lsk
xit:   cjne  r6,#0,ada
       ret
ada:   mov    dsp,r6
       ret
```

Program tersebut diberi nama atau label *scankey*, atau dalam bahasa *assembly* disebut *address relative*, yaitu program untuk membaca tombol mana yang dipilih. Label ini berfungsi sebagai tanda untuk pemanggilan pada program utama. Pembacaan tombol terbagi dalam tiga pembacaan, yaitu tombol sembilan, tombol sepuluh dan tombol satu sampai delapan. Tombol sembilan dan sepuluh terhubung ke P3.0 dan P3.1, tombol satu sampai delapan terhubung ke *port* 1, sehingga tidak mungkin dibaca bersamaan. Prinsip program ini adalah membaca P3.0, P3.1 dan *port* 1 secara terus menerus bergantian. Langkah pertama adalah mengaktifkan *common* tombol menggunakan sebuah transistor untuk mengalirkan *ground* (tegangan negatif) ke masing-masing *port*. Jika ada tombol yang ditekan (*aktif low*), lalu membaca P3.0 yang jika ditekan, maka akan tampil angka sembilan dan jika P3.1 ditekan, maka akan tampil angka sepuluh. Bila salah satu *port* 1 ditekan, maka akan tampil sesuai dengan tombol mana yang ditekan, Jika ditekan P1.0, maka akan tampil angka satu, ditekan P1.1 akan tampil angka dua, ditekan P1.2 akan tampil angka tiga, dan seterusnya (mewakili jenis warna yang akan dicampur dan telah ditentukan sebelumnya).

b. Display seven segment

Display berfungsi untuk menampilkan angka dari hasil penekanan tombol yang mewakili jenis warna yang akan dicampur dan berjumlah dua digit. *Display* menggunakan *seven segment common anoda*, sehingga untuk menyalakannya dibutuhkan data *low* dan saklarnya *high*. Listing program untuk *display* :

Bagian Pertama

```
display: push    dph
          push dpl
          push    b
          mov     dptr,#tbl
          mov     r7,#50
ndisplay: mov     a,dsp
          mov     b,#10
          div    ab
          movc   a,@a+dptr
          mov     seg,a
          clr    dg0
          call   delay
```

Lanjut ke Bagian Kedua

Bagian Kedua

```
setb    dg0
mov     a,b
movc   a,@a+dptr
mov     seg,a
clr    dg1
call   delay
setb    dg1
djnz   r7,ndisplay
pop     b
pop     dpl
pop     dph
ret
```

Prinsip kerja dari program *display* ini adalah mencocokkan data yang akan ditampilkan dengan tabel, lalu mengeluarkannya ke *display* dengan mode *scanning* dan *seven segment* yang akan aktif bergantian, walaupun kenyataannya seperti semua menyala.

c. Pemilihan Warna

Perancangan alat ini terdiri atas empat macam warna pokok yang akan dicampurkan sehingga diperoleh warna baru, yaitu putih, merah, biru dan kuning. Tiap pencampuran, tidak semua warna pokok dipakai, tergantung dari tombol yang ditekan dan mewakili jenis warna yang akan dicampur. Warna-warna pokok tersebut akan dicampur dengan durasi waktu tertentu sehingga menghasilkan warna baru.

Listing program untuk pemilihan warna yang akan dicampur :

Bagian Pertama

```

;Putih *****
                    mov     a,#0
                    movc   a,@a+dptr
                    mov     b,a
                    setb   rkonv
lputih: call      display
                    jb     sputih,lputih
                    clr    rkonv
                    mov     a,b
                    cjne   a,#0,putihon
                    jmp    kuning
putihon: setb     rputih          ;isi putih

isiputih: call     display
                    djnz   b,isiputih
                    clr    rputih
                    call   tunggu

;kuning *****
kuning: mov     a,#1
                    movc   a,@a+dptr
                    mov     b,a
                    setb   rkonv
lkuning: call   display
                    jb     skuning,lkuning
                    clr    rkonv
                    mov     a,b
                    cjne   a,#0,kuningon
                    jmp    biru
kuningon: setb     rkuning        ;isi kuning
isikuning: call   display
                    djnz   b,isikuning
                    clr    rkuning
                    call   tunggu
    
```

Lanjut ke Bagian Kedua

Bagian Kedua

```

;biru*****
biru:   mov     a,#2
                    movc   a,@a+dptr
                    mov     b,a
                    setb   rkonv
lbiru:  call      display
                    jb     sbiru,lbiru
                    clr    rkonv
                    mov     a,b
                    cjne   a,#0,biruon
                    jmp    merah
biruon: setb     rbiru          ;isi biru

isibiru: call     display
                    djnz   b,isibiru
                    clr    rbiru
                    call   tunggu

;merah*****
merah:  mov     a,#3
                    movc   a,@a+dptr
                    mov     b,a
                    setb   rkonv
lmerah: call   display
                    jb     smerah,lmerah
                    clr    rkonv
                    mov     a,b
                    cjne   a,#0,merahon
                    jmp    aduk
merahon: setb     rmerah        ;isi merah
isimerah: call   display
                    djnz   b,isimerah
                    clr    rmerah
                    call   tunggu
    
```

Prinsip kerja dari program tersebut adalah mencocokkan tombol yang ditekan dan mewakili warna yang akan dicampur dengan data yang sudah dibuat pada tabel.

d. Pengaduk

Program ini berfungsi untuk mengontrol motor pengaduk setelah menentukan warna yang akan dicampur . Listing program untuk pengaduk adalah :

Bagian Pertama

```

;aduk*****
aduk:  setb rkonv          ;konveyor on
laduk: call display
    
```

Bagian Ketiga

```

laduk1: call display
                    jb     sdn,laduk1
                    clr    rdn
                    setb   raduk        ;pengaduk on
                    mov     b,#100
laduk2: call display
                    djnz   b,laduk2
                    clr    raduk
    
```

Bagian Kedua

```

jb     saduk,laduk
                    clr    rkonv
                    setb   rdn          ;pengaduk turun
    
```

Bagian Keempat

```

setb   rup          ;pengaduk naik
laduk3: call display
                    jb     sup,laduk3
                    clr    rup
                    mov     dsp,#0
                    ret
    
```

Motor pengaduk tersebut bekerja setelah semua warna dihasilkan. Pengaduk ini akan bekerja secara otomatis berdasarkan hasil pembacaan sensor yang dibaca mikrokontroler mulai dari turun, mengaduk dan naik kembali.

Pengujian Alat

T O M B O L	Kombinasi Perbandingan Waktu, Prosentase dan Volume Prototype Pencampuran Warna							Hasil warna
	Waktu Pengisian Warna Cat (detik)				Prosentase P:K:B:M	Volume (Mililiter) P:K:B:M	Total Volume (Mililiter)	
	Putih (P)	Kuning (K)	Biru (B)	Merah (M)				

Pengujian dilakukan pada sepuluh macam contoh pencampuran warna, walaupun masih banyak warna lain yang bisa dihasilkan. Pencampur warna ini merupakan miniatur sistem dasar pencampuran warna otomatis yang sudah mewakili kinerja dari alat pencampur warna yang sesungguhnya, tetapi belum mewakili hasil atau produk cat yang sesungguhnya dan produk yang dihasilkan lebih encer bila dibandingkan dengan produk cat yang berada di pasaran. Hal ini disebabkan karena *solenoid* yang dipakai tidak memungkinkan untuk campuran cat yang lebih kental, karena lubang *solenoid* yang sangat kecil.

Tabel 2.1 diketahui bahwa dengan empat warna pokok dapat dihasilkan setidaknya sepuluh warna turunan, walaupun bisa dihasilkan warna-warna lain dengan menggunakan empat warna pokok tersebut, tapi dengan durasi waktu yang berbeda. Durasi waktu mewakili jumlah cat yang dikeluarkan dari tabung yang akan menentukan hasil dari pencampuran warna tersebut. Contohnya untuk menghasilkan warna hijau dibutuhkan warna kuning dan biru dengan komposisi yang hampir sama, dalam pencampur warna ini diwakili dengan perbandingan 50% : 50% dengan durasi waktu 2 detik untuk warna kuning dan 7,5 detik untuk warna biru.

Jika durasi waktunya diubah, maka warna yang dihasilkan tentunya akan lain juga, tetapi dalam pengujian ini tidak disebutkan kemungkinan warna yang terjadi jika durasi waktunya diubah untuk membatasi perluasan masalah yang dibahas. Pengujian ini menunjukkan volume cat yang dihasilkan sebanyak satu gelas. Durasi waktu masing-masing pencampuran tidak sama, hal ini disebabkan karena masing-masing cat membutuhkan waktu yang berbeda untuk mengisi satu gelas penuh yaitu untuk putih 10 detik, kuning 4 detik, biru 15 detik, merah 12 detik. Perbedaan waktu ini disebabkan karena perbedaan lubang dan kumparan *solenoid* yang digunakan dan faktor keenceran cat yang tidak mungkin sama karena diaduk secara manual dengan takaran tertentu sebelum dimasukkan ke dalam tabung.

Tabel 1. Data Hasil Pengujian Pencampur Warna

T O M B O L	Kombinasi Perbandingan Waktu, Prosentase dan Volume Prototype Pencampuran Warna							Hasil warna
	Waktu Pengisian Warna Cat (detik)				Prosentase P:K:B:M	Volume (Mililiter) P:K:B:M	Total Volume (Mililiter)	
	Putih (P)	Kuning (K)	Biru (B)	Merah (M)				
1	–	2	7,5	–	50 % : 50%	75:75	150	Spring green
2	8,8	0,3	0,9	–	87,5% : 6,25% : 6,25%	131,25 : 9,38 : 9,38	150	Light green
3	8,8	–	1,9	–	87,5% : 12,5%	131,25 : 18,75	150	Light turquoise
4	8,8	–	–	1,5	87,5% : 12,5%	131,25 : 18,75	150	Rose
5	–	–	7,5	6	50 % : 50%	75 : 75	150	Purple
6	7,5	–	1,9	1,5	75% : 12,5% : 12,5%	112,5 : 18,75 : 18,75	150	Pink

Lanjutan Tabel 1.

T O M B O L	Kombinasi Perbandingan Waktu, Prosentase dan Volume Prototype Pencampuran Warna							Hasil warna
	Waktu Pengisian Warna Cat (detik)				Prosentase P:K:B:M	Volume (Mililiter) P:K:B:M	Total Volume (Mililiter)	
	Putih (P)	Kuning (K)	Biru (B)	Merah (M)				
7	–	3	–	3	75% : 25%	112,5 : 37,5	150	Light orange
8	8,8	0,5	–	–	87,5% : 12,5%	131,25 : 18,75	150	Light yellow
9	–	0,3	0,9	10,5	6,25% : 6,25% : 87,5%	9,38 : 9,38 : 131,25	150	Fire brick
10	2,5	1	3,8	3	25% : 25% : 25% : 25%	37,5 : 37,5 : 37,5 : 37,5	150	Dark olive green

Kesimpulan

Berdasarkan data hasil pengujian, alat pencampur warna ini dapat mencampurkan warna secara otomatis berdasarkan durasi waktu untuk menentukan jumlah cat yang dicampur. Alat ini merupakan salah satu miniatur dari sistem pencampur warna otomatis yang hanya dapat mencampur maksimal empat warna pokok untuk menjadi warna baru dan belum dapat mencampur lebih dari empat warna pokok. Alat ini dirancang untuk mencampur cat encer dan tidak dirancang untuk cat kental karena *solenoid* yang dipakai adalah *solenoid* dengan lubang kecil. *Prototype* alat pencampur warna ini semoga dapat direalisasikan menjadi alat sesungguhnya sehingga dapat membantu pengusaha cat/pencampur warna dalam sistem pengerjaan mencampur warna paten untuk pilihan pembeli secara otomatis.

Daftar Pustaka

Eko Putra, Agfianto, (2000), “*Belajar Mikrokontroler AT89C51/52/55 Teori dan Aplikasi*”, Gaya Media, Yogyakarta

Suhata, ST, “*Aplikasi Mikrokontroler Sebagai Pengendali Peralatan Elektronik*”, Gramedia, Jakarta

Sutanto, Budhy, (2003), “Mengenali Seri S MCS51”, *Komputek*, Minggu ketiga Juli 2003, hal.27

Wasito S. Vademekum, (2001), “*Elektronika*”, Gramedia, Jakarta

<http://pdf1.alldatasheet.com/datasheet-pdf/view/77367/ATMEL/AT89S51.html>

<http://pdf1.alldatasheet.com/datasheet-pdf/view/83230/PHILIPS/ADC0804.html>

<http://pdf1.alldatasheet.com/datasheet-pdf/view/5775/MOTOROLA/TL494.html>

<http://pdf1.alldatasheet.com/datasheet-pdf/view/8296/NSC/DAC0808.html>