

Perancangan Pengatur Laju Air Pada Kran Secara Otomatis Menggunakan Mikrokontroler

Yusmartato¹⁾, Zulfadli Pelawi²⁾, Luthfi Parinduri³⁾

^{1,2)}Prodi Teknik Elektro, ³⁾Prodi Teknik Industri, Fakultas Teknik
Universitas Islam Sumatera Utara, Medan
yusmartato@ft.uisu.ac.id; luthfip@yahoo.co.id

Abstrak

Perancangan Mikrokontroler sebagai sistem pengatur Laju air pada washtapel ini menggunakan sebuah photo dioda yang mana Anodanya dihubungkan ke Vcc sedangkan Katodanya dihubungkan ke basis dari transistor BC547 melalui sebuah resistor 1K. Apabila LED infra merah menyala maka sinarnya akan terpancar, apabila pancaran dari LED infra merah tersebut mengenai tangan maka pancaran LED tersebut akan memantul. Pantulan LED tersebut akan balik mengenai dari sensor photo dioda. Hal ini tentunya bergantung dari jarak antara sensor dengan tangan, jika terlalu jauh maka pantulannya tidak akan sampai menuju sensor photodiode. Apabila masih dalam jangkauannya maka photo dioda akan aktif sehingga Vcc (logika 1) akan menuju basis transistor BC547. Dengan kata lain tegangan aktif/on juga. Dengan demikian maka pada kolektor dari transistor BC547 tersebut akan berlogika 0. Logika 0 tersebut akan masuk menjadi in-put dari port. P2.0 dari mikrokontroler.

Kata Kunci : Mikrokontroler, Laju Air, Sensor, Transistor

I. PENDAHULUAN

Kemajuan bidang elektronika saat ini dalam bidang industri dan rumah tangga, diantaranya adalah sistem pengontrolan peralatan elektronik yang ada berupa komputer, mesin cuci, pengatur motor listrik dan lain-lain sebagainya, Dimana sistem pengontrolan berawal dari sistem analog ke digital.

Pada lingkungan masyarakat banyak mendapatkan kendala dalam suatu bidang pengaturan laju air pada kran, karena dalam pengaturan laju air pada kran ini tujuannya adalah untuk menghemat pemakaian air, untuk mempermudah pengaturan laju air pada kran ini maka penulis menggunakan suatu sistem mikrokontroler sebagai pengatur laju air pada kran secara otomatis, mikrokontroler bukan hanya mampu mengatur segala sistem tetapi juga mampu melindungi sistem serta dapat memperbaiki kinerja sistem menjadi lebih baik. Hal itu terjadi karena multifungsi dari mikrokontroler sebagai pusat kerja suatu sistem.

Perancangan ini menggunakan Ic mikrokontroler AT89S51 untuk mengoperasikan suatu sistem laju air pada kran. Karena Ic mikrokontroler AT89S51 mudah diperoleh dipasaran dengan harga yang relatif murah, Memiliki Flash EPROM untuk menyimpan program, Tidak banyak memerlukan tambahan komponen lain, Banyak referensi mengenai hardware maupun software, terutama di situs internet, Mudah dibuat perangkat pengembangannya yang berupa software maupun programernya.

Dasar dari mikrokontroler adalah mikroprosesor, oleh karena itu dalam perancangan

ini mula-mula membahas mengenai mikroprosesor serta komponen pendukungnya secara umum terlebih dahulu, tapi dalam perancangan ini lebih menjurus ke dunia mikrokontroler, khususnya mikrokontroler AT89S51.

II. TINJAUAN PUSTAKA

2.1 Mikrokontroler AT89S51

Mikrokontroler adalah suatu alat pengontrol atau pengendali yang berukuran kecil. Sebelum mikrokontroler ada, telah terlebih dahulu yang disebut mikroprosesor, bila dibandingkan mikrokontroler dan mikroprosesor jauh lebih unggul mikrokontroler dimana dalam mikrokontroler telah tersedia CPU (Central Processor Unit), RAM, ROM, dan port I/O dalam satu chip.

Mikrokontroler AT89S51 merupakan produk ATMEL yang memiliki fasilitas yaitu : Kompatibel dengan MCS-51, 4 Kbyte memori program yang dapat ditulis hingga 1000 kali, 0 Kecepatan clock - 33MHz, 128 byte memori RAM internal, 32 jalur input - output (4 buah port paralel I/O), 2 Timer/Counter 16 bit, 2 Data pointer, 6 Interrupt (2 timer, 2 counter, 1 serial, 1 reset), ISP (In System Programmabel) Flash Memory, Port serial full-duplex.

Sebuah komputer mikro memiliki tiga komponen utama : unit pengolah pusat (CPU = *central processor unit*), memori dan sistem input/output (I/O) untuk dihubungkan dengan perangkat luar. CPU, yang mengatur sistem kerja komputer mikro, dibangun oleh sebuah mikroprosesor. Memori terdiri atas Flash Memory untuk menyimpan program dan RAM untuk menyimpan data. Sistem I/O bias dihubungkan

dengan perangkat luar misalnya sebuah keyboard dan sebuah monitor, bergantung pada aplikasinya. Apabila CPU, memori dan sistem I/O dibuat dalam sebuah chip semikonduktor, maka inilah yang dinamakan mikrokontroler.

Generasi pertama mikrokontroler dibuat oleh Intel Corporation pada akhir decade 70-an. Mikrokontroler yang dinamakan sebagai keluarga 8051, kemudian menjadi sebuah standar industri. Banyak perusahaan lain yang membuat dan mengembangkan 8051 seperti Philips Semiconductor, Atmel Corporation, Dallas Semiconductor, dan lain – lain. Selain itu berkembang juga mikrokontroler keluarga lain, seperti Motorola dengan seri 68HC-nya, Atmel dengan AVR-nya, Microchip dengan PIC-nya dan masih banyak lagi.

Mikrokontroler tidak hanya terdiri atas memori internal ataupun system I/O, tetapi juga dilengkapi dengan fasilitas lain (*on-chip facilities*) seperti timer, port serial, pencacah (*counter*), kendali interupsi, pengubah analog ke digital (*ADC=analog-to-digital converter*) dan digital ke analog (*DAC= digital-to-analog converter*) bergantung pada untuk aplikasi apa mikrokontroler tersebut dirancang. Mikrokontroler generasi terbaru bisa dilengkapi dengan port USB, koneksi LAN, penggerak (*driver*) LCD, antar muka untuk koneksi ke hard disk ataupun decoder mp3.

Perangkat lunak (*software*) pun berkembang pesat. Pada awalnya bahasa pemrograman mikrokontroler memakai bahasa assembler (bahasa tingkat rendah), kini para perancang sistem berbasis mikrokontroler telah banyak menggunakan bahasa tingkat tinggi seperti bahasa C, BASIC ataupun pascal. Dengan menggunakan bahasa pemrograman tingkat tinggi ini para perancang lebih cepat mengembangkan aplikasi karena beberapa baris perintah dalam bahasa assembler bias digantikan dengan hanya satu baris perintah bahasa tingkat tinggi. Kelemahannya adalah, memori program bisa lebih besar bila dibandingkan dengan bahasa assembler. Selain itu, tidak semua bahasa assembler bisa dengan mudah diterjemahkan ke bahasa tingkat tinggi, sehingga masih tetap dibutuhkan baris program yang ditulis dengan bahasa assembler dalam program tingkat tinggi (*in-line assembler*). Mikrokontroler generasi terbaru telah dirancang agar lebih sesuai dengan bahasa tingkat tinggi karena bagaimanapun bahasa tingkat tinggi lebih “bisa dibaca”.

2.2 Sistem Minimum Rangkaian

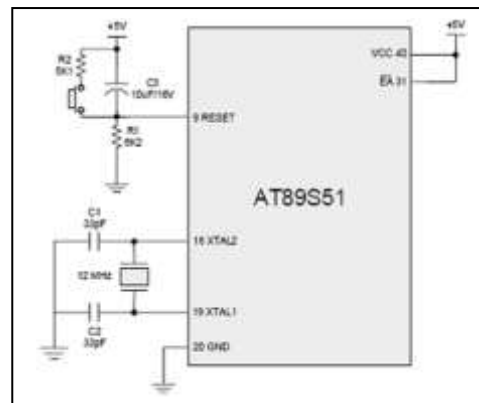
Agar mikrokontroler dapat bekerja maka diperlukan komponen tambahan berupa rangkaian reset dan clock.

Mengapa perlu reset pada mikrokontroler, karena pada saat power dinyalakan instruksi yang pertamakali dieksekusi oleh mikrokontroler adalah instruksi yang tersimpan pada address 0000h. Agar program counter (PC) dapat menunjukkan address 0000h pada saat awal maka mikrokontroler perlu di-reset. Caranya adalah dengan memberikan pulsa

high pada pin reset selama minimal 2 machine cycle (jika $f_{crystal} = 12 \text{ MHz}$ maka $2MC = 2 \text{ uS}$). Setelah itu baru diberikan pulsa low. Kondisi ini dapat dipenuhi dengan memasang rangkaian RC yang akan mensuplay tegangan VCC ke pin 9 selama kapasitor mengisi muatan / *charging*. Konstanta waktu pengisian dapat dihitung dalam mengalikan nilai R dan C. Pada rangkaian dibawah ini terdapat $T=R.C = (8K2) \cdot (10\mu F) = 82mS$. Setelah kapasitor terisi, maka pin 9 akan low. Tombol push button dipasang agar pada saat *running* Mikrokontroler dapat juga di-reset. Pin EA / Eksternal Access harus dihubungkan ke +5V agar mikrokontroler dapat mengambil byte instruksi dari ROM internal mikrokontroler.

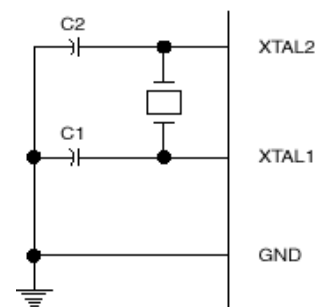
2.3 Osilator

Fungsi dari osilator adalah untuk menyediakan sinyal clock dan pewaktuan bagi semua perangkat internal. Untuk menyediakan sinyal clock, bisa digunakan rangkaian osilator internal ataupun pembangkit sinyal clock eksternal, pin ini berfungsi sebagai masukan.



Gambar 1. Rangkaian minimum AT89S51

Apabila dipakai osilator internal, maka sebuah Kristal atau resonator keramik digunakan sebagai penentu frekuensi sinyal clock dengan 2 buah kapasitor (C1 dan C2), seperti ditunjukkan pada gambar C1 dan C2 biasanya digunakan 30 pF bila Kristal dipakai atau 47 pF bila resonator keramik yang dipakai.



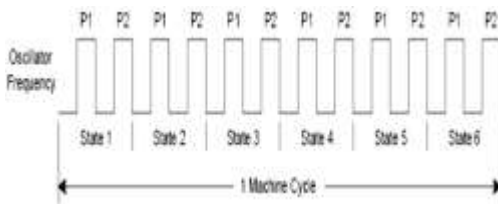
Gambar 2. Rangkaian clock

Osilator akan menyediakan pewaktuan internal untuk perangkat – perangkat internal,

seperti timer dan port serial. Dalam memakai port serial, pemilihan frekuensi Kristal osilator sangat menentukan Karena frekuensi Kristal menentukan baud rate yang bisa dihasilkan nilai 11.059 MHz paling banyak dipakai karena frekuensi ini mampu menghasilkan baud rate standart yang sering digunakan.

Hal yang paling mendasar dari osilator sebenarnya adalah untuk menentukan siklus mesin (*mechine cycle*). Secara sederhana satu siklus mesin diartikan sebagai waktu minimum yang diperlukan oleh mikrokontroler untuk menjalankan satu perintah. Siklus mesin ini akan menentukan kecepatan mikrokontroler (artinya seberapa cepat mikrokontroler menjalankan suatu perintah). Satu siklus mesin terdiri dari 12 x priode frekuensi osilator (dengan frekuensi 12 MHz, satu siklus mesin adalah 1 mikro detik). Satu siklus mesin ini dibagi menjadi 6 state. Setiap state terdiri dari 2 fase dimana setiap fase terdiri atas 1 priode osilator.

Dalam mikrokontroler dikenal istilah Machine cycle (MC) / Siklus Mesin, dimana 1 MC = 6 state = 12 priode clock, Jika frekuensi cristal yang digunakan adalah 12 MHz maka 1 MC = 12/frekuensi crystal = 12/12 = 1uS



Gambar 3. Siklus mesin AT89S51

2.4 Waktu Eksekusi Intruksi

Waktu eksekusi sebuah instruksi oleh mikrokontroler tergantung dari jenis instruksi dan frekuensi clock yang digunakan. Setiap instruksi memiliki panjang byte dan jumlah siklus yang berbeda. Byte instruksi (**Byte**) menandakan jumlah lokasi memori yang dipakai. Siklus instruksi (**Cycle**) menandakan jumlah machine cycle yang dibutuhkan. Waktu eksekusi dapat dihitung dengan rumus :

$$T_{inst} = \frac{C \times 12}{\text{frekuensi crystal}}$$

Dimana :

Tinst : waktu yang dibutuhkan untuk mengeksekusi 1second jumlah macine cycle

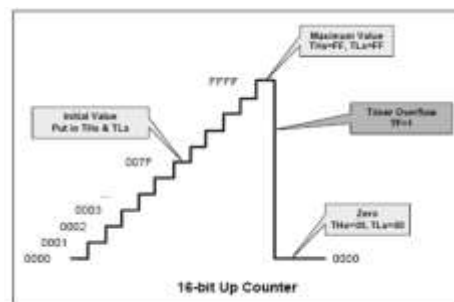
Pada sebuah mikrokontroler dengan frekuensi crystal 12 MHz. Waktu yang diperlukan untuk mengeksekusi perintah : `Mov A,#30h`, Kita dapat melihat lembar data 8051 Operational Code Mnemonic diketahui bahwa intruksi dengan format

`Mov A,#n` adalah instruksi dengan byte = 1 dan Cycle = 1. Maka : $T_{inst} = (1 \times 12) / 12 \text{ MHz} = 1 \text{ uS}$

Timer / Counter Interrupt

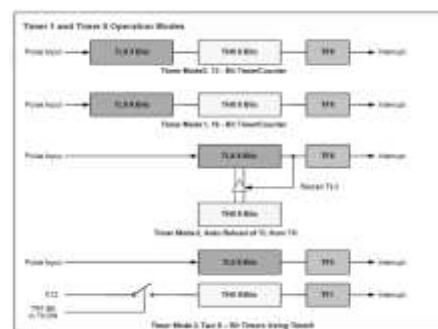
Timer/counter pada AT89S51 adalah sebuah Up Counter, nilai counternya akan naik (Increment) dari nilai awalnya sampai nilai maksimumnya dan kembali ke nilai nol. Saat bergulir menjadi nol (*overflow*), maka sebuah timer flag akan bernilai 1. Flag ini dapat diuji oleh program untuk menandakan bahwa counter telah selesai menghitung, atau flag tersebut bisa digunakan untuk meng-intrupsi program.

Nilai awal timer/counter harus dimasukkan dulu ke dalam timer register Timer High (TH) dan Timer Low(TL) sebelum timer/counter dijalankan.



Gambar 4. Timer / Counter Interrupt

Mode Operasi Timer



Gambar 5. Mode operasi timer

Pemilihan mode operasi timer ditentukan pada bit M1 dan M0 dalam register TMOD. Ada 4 mode yaitu :

1. Mode 0 : 13-bit Timer/counter
2. Mode 1 : 16-bit Timer/counter
3. Mode 2 : 8-bit Autoreload Timer/Counter
4. Mode 3 : Two 8 bit Timer/counter

Timer mode 0. 13-bit Timer/counter.

Dengan mensetting M1 dan M0 = 00 dengan TMOD menyebabkan register THx berfungsi sebagai counter 8 bit dan register TLx berfungsi sebagai counter 5 bit. Ketika overflow. TF1x akan 1. Nilai maksimumnya adalah 8191d atau 1FFFh.

Timer mode 1. 16-bit Timer/Counter.

Register THx dan TLx masing – masing berfungsi sebagai counter 8 bit. Ketika overflow, TFlx akan 1. Nilai maksimumnya adalah 65535d atau FFFFh.

Timer mode 2. 8-bit Autoreload Timer/Counter.

Register TLx berfungsi sebagai counter 8-bit. Register THx berfungsi mengisi ulang / autoreload register TLx ketika terjadi overflow (TFx=1).

Timer mode 3. Two 8 bit Timer/Counter.

Pada mode 3 Timer berfungsi sebagai counter 8 bit yang benar – benar terpisah satu sama lain. Timer 0 berfungsi sebagai timer sekaligus sebagai counter secara terpisah. TLO digunakan sebagai counter 8 bit yang menghitung pulsa eksternal, dengan timer flag TF0. TH0 digunakan sebagai timer 8 bit yang menghitung pulsa clock internal, dengan timer flag TF1. Pada mode 3. Timer 1 tidak dapat digunakan sebab timer flag TF1 digunakan sebagai timer flag TH0.

Timer/Counter dapat dihidup – matikan secara proram dengan mengatur TRx maupun secara hardware dengan memberikan logika 0 pada pin INTx. Berikut adalah tabel nilai TMOD sesuai dengan mode dan control timer/counter.

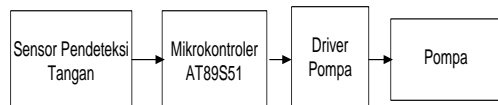
Perancangan Hardware

Perancangan sistem alat ini diawali dengan pembuatan blok diagram dari sistem tersebut. Dimana tiap-tiap blok saling berhubungan satu dengan yang lain. Perancangan sistem ini dibagi dalam dua bagian yaitu: perancangan secara hardware dan perancangan secara software. Dan masing-masing blok akan dibahas dalam bab ini.

Perancangan Diagram Blok Sistem

Untuk memudahkan pembuatan sistem ini maka digunakanlah diagram blok sebagai langkah awal pembuatan sistem. Diagram blok menggambarkan secara umum bagaimana cara kerja rangkaian secara keseluruhan.

Diagram blok ini memiliki peran yang sangat penting. Adapun diagram blok sistem ini adalah sebagai berikut:



Gambar 6. Blok Diagram

Adapun fungsi dari masing-masing blok tersebut adalah:

Blok Sensor Pendeteksi Tangan :

Dalam perancangan ini menggunakan sensor yang dapat mendeteksi adanya benda/tangan. Sensor ini berfungsi untuk mendeteksi adanya benda yang ada di depan sensor tersebut.

Blok Mikrokontroler AT89S51 :

Berfungsi sebagai pengontrol atau pengendali semua cara kerja rangkaian sehingga sistem ini dapat bekerja sesuai fungsi masing-masing.

Blok Driver Pompa :

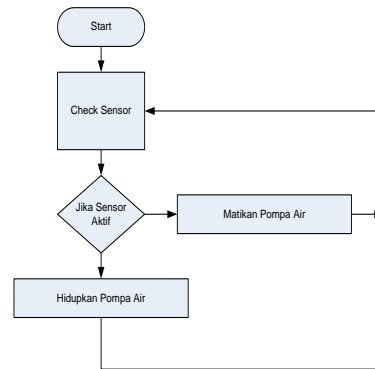
Bagian ini berfungsi untuk menjadi driver atau pengendali dari keaktifan Pompa.

Blok Pompa :

Blok ini berfungsi untuk megisap air dan memasukkannya ke dalam bak.

Perancangan Software

Di dalam perancangan software dibagi menjadi dua bagian yaitu perancangan flowchart dan perancangan program yang digunakan. Adapun flowchart yang digunakan adalah sebagai berikut:



Gambar 7 . Flowchart

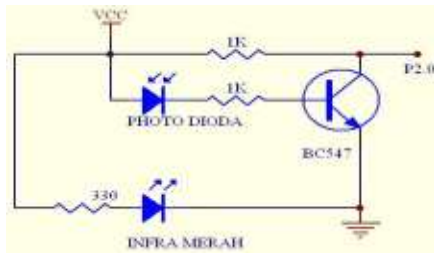
Sedangkan program yang digunakan adalah:

```

p0      equ    080h
p1      equ    090h
p2      equ    0a0h
p3      equ    0b0h
org     0000h
start:  jnb    p2.0,on_pump
        jmp    off_pump
on_pump: setb   p0.0
        call  delay
        clr   p0.0
        jmp  start
off_pump: clr   p0.0
        jmp  start
delay:   mov   r3,#0ch
del2:    mov   r4,#0ffh
del1:    mov   r5,#0ffh
        djnz  r5,$
        djnz  r4,del1
        djnz  r3,del2
        ret
        end
    
```

Rangkaian Sensor Pendeteksi Tangan

Adapun rangkaian sensor pendeteksi tangan yang digunakan adalah sebagai berikut:



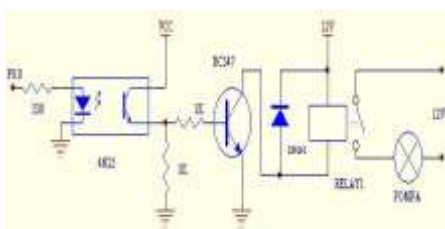
Gambar 8. Rangkaian sensor pendeteksi tangan

Dalam perancangan sensor pendeteksi tangan ini menggunakan sebuah photodiode yang mana Anodanya dihubungkan ke Vcc sedangkan Katodanya dihubungkan ke basis dari transistor BC547 melalui sebuah resistor 1K. Apabila LED infra merah menyala maka sinarnya akan terpancar, apabila pancaran dari LED infra merah tersebut mengenai tangan maka pancaran LED tersebut akan memantul. Pantulan LED tersebut akan balik mengenai dari sensor photodiode. Hal ini tentunya bergantung dari jarak antara sensor dengan tangan, jika terlalu jauh maka pantulannya tidak akan sampai menuju sensor photodiode. Apabila masih dalam jangkauannya maka photodiode akan aktif sehingga Vcc (logika 1) akan menuju basis transistor BC547. Dengan kata lain tegangan Vbe (0,7V) dari transistor tersebut akan terpenuhi, sehingga transistor BC547 akan aktif/on juga. Dengan demikian maka pada kolektor dari transistor BC547 tersebut akan berlogika 0. Logika 0 tersebut akan masuk menjadi input dari port P2.0 dari mikrokontroler.

Sedangkan apabila pantulan dari infra merah tidak sampai menuju sensor photodiode maka photodiode tidak akan aktif/on. Dengan demikian Vcc tidak akan mengalir menuju basis transistor dan akan mengakibatkan transistor akan off. Dengan off nya transistor maka pada port P2.0 dari mikrokontroler akan berlogika 1.

Rangkaian Driver Pompa

Komponen yang digunakan pada perancangan rangkaian driver Valve ini adalah : Opto-Coupler (4N25), BC 547, Dioda IN 4002, Resistor 1KΩ, serta Relay 12V. Rangkaian driver pompa ini merupakan rangkaian yang menjadi penggerak atau pengendali proses aktifnya pompa yang digunakan.



Gambar 9. Rangkaian Driver Pompa

Pada perancangan driver Pompa inii digunakan 4N25. 4N25 merupakan sebuah Optocoupler yang dapat memperkuat data keluaran dari mikrokontroler. Driver dengan menggunakan 4N25 ini juga sangat cocok untuk mengamankan data yang masuk ke mikrokontroler akibat adanya loncatan bunga api yang terjadi saat pompa sedang aktif/on. Jika mikrokontroler mengirimkan perintah untuk mengaktifkan 4N25, maka optocoupler akan aktif atau ON karena photo transistor menerima sinyal yang dikirimkan oleh infra red pada optocoupler. Kemudian data tersebut dikirimkan oleh 4N25 ke transistor BC 547 dengan demikian BC 547 akan aktif sebagai saklar posisi. Lalu data tersebut akan mengaktifkan relay sehingga relay menjadi ON, dimana pada awalnya relay dalam kondisi Normally Close akan berubah aktif menjadi Normally Open. Pada saat relay aktif pada kondisi ON, maka relay bertindak sebagai saklar dan mengalirkan tegangan 12V terhadap beban Pompa yang digunakan, sehingga Pompa tersebut akan hidup.

III. HASIL DAN PEMBAHASAN

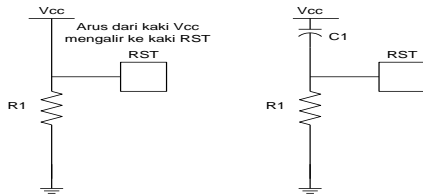
Pengujian dan pengukuran dilakukan untuk membuktikan apakah rangkaian yang sudah dibuat bekerja sesuai dengan yang direncanakan. Pertama sekali pengujian dilakukan pada setiap bloknya dan pengujian beberapa blok yang saling berkaitan.

Setelah semua komponen dipasang dan semua instalasi selesai, lalu dilakukan pemeriksaan ulang terhadap jalur PCB, solderan dan pengawatan agar pengujian dan pengukuran dapat dilaksanakan dengan cepat dan baik.

3.1 Analisa Hardware

Bagian ini merupakan pemroses keseluruhan dari sistem ini. Rutin yang dikerjakan ditulis dalam bahasa assembling yang selanjutnya didownload pada memori internal yang tersedia. Mikrokontroler ini buatan ATMEL yang kompatibel dengan keluarga MCS-51, di dalamnya terdapat 4 Kbyte of In-System Reprogrammable Flash Memory, dengan 32 jalur I/O. 128 Bytes RAM. Pada rangkaian ini tidak semua port dipakai (P0, P1, P2, P3) sebagai input dan output. Rangkaian eksternal sebagai pembangkit frekuensi yang dipakai sesuai karakteristiknya yaitu pada C₂, C₃ dan XTAL sedangkan untuk rangkaian reset dipergunakan komponen C₁ dan R₁. Dalam pengujian didapat hasil pengukuran seperti Tabel 1.

| Tabel 1. titik pengukuran kondisi power on reset | |
|--|----------|
| Titik pengukuran | Power On |
| TP1 | H → L |
| TP2 | H ↔ L |
| TP3 | L |

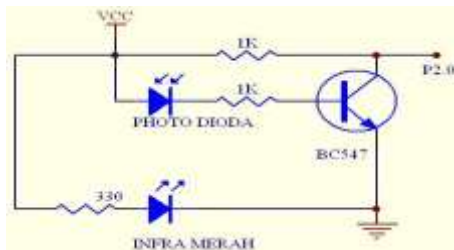


Gambar 10. Aliran arus dan perubahan tegangan pada reset otomatis

Pada saat sumber daya diaktifkan, maka kapasitor C_1 sesuai dengan sifat kapasitor akan terhubung singkat pada saat itu sehingga rangkaian ekuivalennya tampak pada gambar Arus mengalir dari VCC langsung ke kaki RST sehingga kaki tersebut berlogika 1. Kemudian kapasitor terisi hingga tegangan pada kapasitor (V_c) yaitu tegangan antara Vcc dan titik antara kapasitor C_1 dan resistor R_1 mencapai Vcc, otomatis tegangan pada R_1 atau tegangan RST akan berlogika 0 (gambar 4.2b) dan proses reset selesai.

3.2 Pengujian Rangkaian Sensor Pendeteksi Tangan

Saat sensor mendeteksi adanya benda/tangan yang berada pada jarak deteksinya maka Vcc akan mengalir melalui photodiode dan Resistor 1K menuju base dari transistor BC547 (Transistor type NPN), hingga mengakibatkan Transistor tersebut akan ON. Dengan demikian maka pada kaki kolektor dari Transistor Q1 akan berlogika 0 (bertegangan sebesar 0,3 mV). Namun Jika sensor tidak mendeteksi adanya benda/tangan dihadapannya maka akan mengakibatkan transistor BC547 akan off, sehingga pada kaki kolektor Q1 akan berlogika rendah atau 1 (bertegangan 4,89V).



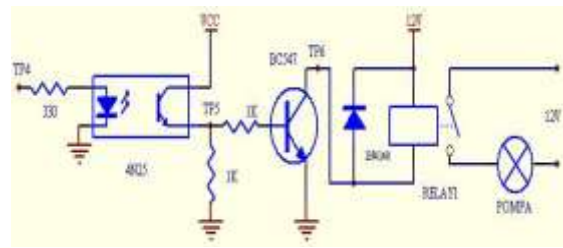
Gambar 12. Gambar rangkaian sensor level air

Adapun tabel hasil pengukurannya dapat dilihat pada Tabel 2.

Tabel 2. Titik pengukuran pada driver beban valve dan pompa

| Kondisi | P 2.0 |
|------------------------|--------------|
| Mendeteksi Benda | 0 (0,3mV) |
| Tidak Mendeteksi Benda | 1 (4,89V) |

Pengujian Rangkaian Driver Pompa



Gambar 11. Gambar titik pengukuran rangkaian driver

Jika mikrokontroler mengirimkan perintah berupa logika 1 untuk mengaktifkan 4N25, maka optocoupler akan aktif dikarenakan infra merah akan ON (akan menyinari photo transistor) sehingga kolektor dan emitor dari photo transistor akan bersambung dan akan mengakibatkan arus dari Vcc akan mengalir melalui kolektor menuju emitor dari photo transistor hingga akan masuk ke kaki basis dari Transistor BC547 dan mengakibatkan Transistor BC547 akan ON atau aktif sebagai saklar. Sebelumnya data dari Ground akan mengalir ke basis dari transistor BC547 yang mengakibatkan transistor Cut Off.

Saat Transistor BC547 ON maka arus dari Ground akan mengalir dari emitor menuju kolektor dari Transistor BC547, dengan demikian arus akan mengalir dari Vcc menuju coil dari relay menuju ke Ground sehingga akan mengaktifkan relay dan mengakibatkan posisi relay menjadi ke Normaly Open (NO). Jika Transistor dalam keadaan CUT Off (data pada base berlogika 0) maka relay pada posisi normally close. Pada saat relay aktif pada kondisi normally open, maka relay bertindak sebagai saklar dan mengalirkan tegangan 12V terhadap beban Pompa, sehingga Pompa akan hidup.

Setiap relay yang digunakan pada perancangan alat ini mampu mengendalikan beban hingga 3A. Adapun hasil pengukuran yang di dapat pada titik pengukuran di atas adalah sebagai berikut:

Pada pengukuran yang dilakukan dengan menggunakan logic probe, saat beban Pompa Hidup, maka pada TP4 berlogika 1, pada TP5 akan berlogika 1 juga sedangkan pada TP6 akan berlogika 0 atau tegangan mendekati 0,3 mV. Pada saat Pompa dalam kondisi Mati, maka pada TP4 akan berlogika 0, pada TP5 akan berlogika 0, sedangkan pada TP6 akan berlogika 1.

Tabel 3. Titik pengukuran pada driver beban valve dan pompa

| Kondisi | TP4 | TP5 | TP6 |
|-------------|-----|-----|-----|
| Pompa Hidup | 1 | 1 | 0 |
| Pompa Mati | 0 | 0 | 1 |

3.3 Analisa Program

Untuk memudahkan pengenalan masing-masing port dalam pemrograman, dibutuhkan sebuah inisialisasi masing-masing port tersebut, adapun port-port yang digunakan haruslah diberi alamat dari masing-masing port tersebut. Untuk port p1 memiliki alamat 090h sedangkan port p3 alamatnya 0b0h.

```
p0      equ    080h
p1      equ    090h
p2      equ    0a0h
p3      equ    0b0h
```

Pertama-tama program diawali dengan pengalamatan awal program (diawali dengan alamat 0000h).

```
org     0000h
```

Selanjutnya mikrokontroler akan memeriksa port p2.0 apakah berlogika 0 atau 1, jika berlogika 0 maka mikrokontroler akan lompat ke rutin on_pump, namun apabila berlogika 1 maka program akan lompat ke rutin off_pump.

```
start: jnb    p2.0,on_pump
        jmp    off_pump
```

Pada rutin on_pump port p0.0 akan diberi logika 1 lalu program akan lompat ke rutin delay kemudian diikuti dengan memberi logika 0 pada port p0.0 yang selanjutnya diikuti dengan lompat ke rutin start.

```
on_pump: setb   p0.0
          call  delay
          clr   p0.0
          jmp  start
```

Pada rutin off_pump, isi dari port p0.0 diberi logika 0, lalu program akan lompat ke rutin start.

```
off_pump: clr    p0.0
          jmp    start
```

Subroutine delay berisikan proses penundaan waktu yang dilakukan pada saat setiap beep akan dihidupkan. Lamanya waktu tunda yang diberikan bergantung dari nilai r1, r2 dan r3, apabila nilai tersebut dinaikkan maka akan menyebabkan penundaan waktu yang lebih lama. Demikian sebaliknya jika nilai r1, r2, dan r3 dikurangi/dikecilkan maka akan mengakibatkan penundaan waktu yang lebih cepat.

```
delay:  mov    r3,#0ch
del2:   mov    r4,#0ffh
del1:   mov    r5,#0ffh
        djnz  r5,$
        djnz  r4,del1
        djnz  r3,del2
        ret
```

Pada akhir dari program akan selalu diakhiri dengan perintah end.

End

IV. KESIMPULAN

1. Dalam perancangan sensor pendeteksi tangan menggunakan sebuah photodiode.
2. Untuk memudahkan pengenalan masing-masing port dalam pemrograman, dibutuhkan sebuah inisialisasi masing-masing port.
3. Perancangan alat sesuai dengan yang diharapkan, dan sudah berjalan semestinya.
4. Setiap relay yang digunakan pada perancangan alat ini mampu mengendalikan beban hingga 3A

DAFTAR PUSTAKA

- [1]. Sriwijaya, 2008, *Intisari Elektronika*, Jakarta. George Loveday.
- [2]. Usman, 1991, *Teknik Antar muka + Pemrograman Mikrokontroler AT89S51*, Jakarta. Penerbit Andi.
- [3]. Suhata, 2005, *VB Sebagai Pusat Kendali Peralatan Elektronika*, Jakarta. Penerbit PT. Elex Media Komputindo.
- [4]. Website : www.BelajarMickrokontroler.com
- [5]. Rachmad Setiawan, 2006, *Mikrokontroler MCS-51*, Graha Ilmu. Yogyakarta
- [6]. Scanlon, LeoJ., 1986, *Assembly Language Programming with The IBM PC AT*, Simon & Scuster Inc., New York, USA.