

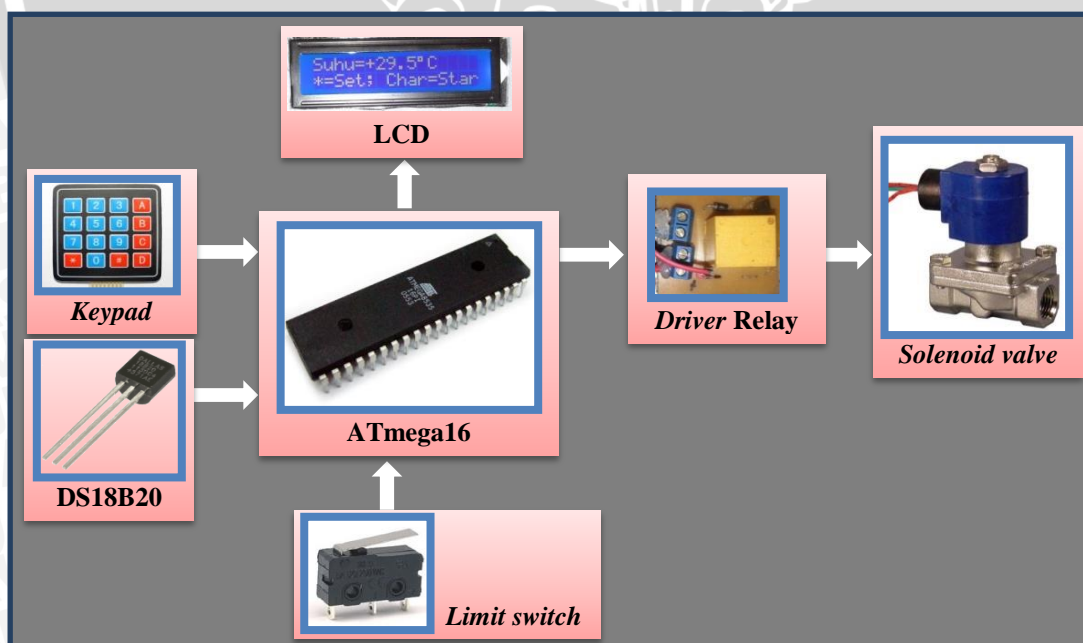
BAB IV PERANCANGAN DAN PEMBUATAN ALAT

Perancangan dan pembuatan alat dispenser otomatis ini perlu memahami karakteristik komponen penyusunnya serta konteks kerjanya pada rangkaian sehingga alat yang dirancang nantinya dapat bekerja maksimal sesuai spesifikasi yang telah ditentukan.

Pembuatan alat ini meliputi pembuatan perangkat keras sesuai diagram blok sistem dan pembuatan perangkat lunak sesuai dengan algoritma yang diinginkan kemudian mengintegrasikan pada keseluruhan sistem. Oleh karena itu direncanakan terlebih dahulu perangkat – perangkat yang mendukung sistem kerja alat ini. Untuk pengujian perangkat keras meliputi pengujian terhadap *solenoid valve*, pengujian driver relay pengujian sensor suhu, pengujian keypad dan LCD. Sedangkan untuk pengujian perangkat lunak berupa program yang menggunakan pemrograman bahasa C dengan memanfaatkan perangkat lunak *Code Vision AVR 2.0.5*.

4.1 Blok Diagram Sistem

Secara garis besar, diagram blok perancangan perangkat keras secara keseluruhan ditunjukkan dalam Gambar 4.1.



Gambar 4.1 Diagram blok sistem

Sesuai dengan judul skripsi tentang “Dispenser Pengisi Gelas Otomatis Menggunakan Mikrokontroler ATmega16 dan *Interface Keypad* “ maka dibuat sebuah

sistem yang dijabarkan pada blok diagram tersebut. Penjelasan dari diagram blok sistem sebagai berikut:

- 1) DS18B20 berfungsi sebagai sensor suhu untuk mengukur suhu pada dispenser
- 2) *Keypad* berfungsi sebagai *interface* untuk mengatur input berupa volume air yang diinginkan
- 3) *Limit switch* berfungsi untuk mendeteksi keberadaan gelas tepat dibawah keran yang dikopel dengan *solenoid valve* sehingga nantinya akan membuka keran secara otomatis
- 4) Mikrokontroler yang digunakan adalah ATmega16 yang berfungsi sebagai pengendali sistem utama
- 5) LCD berfungsi untuk menampilkan suhu pada dispenser dan masukan dari *keypad* pada saat sistem dijalankan
- 6) Rangkaian *driver* terdiri atas *relay* yang digunakan sebagai saklar elektronik yang menghubungkan keluaran mikrokontroler dengan *solenoid valve* yang dikopel dengan *limit switch*
- 7) *Solenoid valve* berfungsi untuk membuka dan menutup keran secara otomatis yang dikontrol menggunakan mikrokontroler ATmega16.

Dispenser memiliki bagian masukan berupa *key pad*, sensor suhu DS18B20, dan *limit switch*. *Keypad* tersebut terdiri dari tombol (A,B,C,D) untuk mengisi air di gelas sesuai dengan volume yang kita atur, contohnya jika kita ingin mengatur volume A maka cara menyettingnya dengan menekan tombol (*1) kemudian masukkan volume yang diinginkan dan simpan dengan menekan tombol (#), selain itu jika ingin memasukkan volume untuk tombol yang lainnya maka cara menyettingnya dengan menekan (*2) untuk tombol B, (*3) untuk tombol C, (*4) untuk tombol D kemudian tekan tombol (#) untuk menyimpannya. Proses selanjutnya jika ingin mengeluarkan air sesuai volume yang telah *disetting*, kita hanya menekan salah satu tombol *keypad* (A,B,C,D). Setelah itu jika gelas belum diletakkan dibawah keran (gelas tidak menekan *limit switch*) maka program belum berjalan dan ditampilkan pada LCD bahwa “ tidak ada gelas”, tetapi jika gelas sudah berada tepat dibawah keran (gelas menekan *limit switch*) maka program akan mengaktifkan *driver solenoid valve* .

Sensor yang digunakan dalam perancangan dispenser otomatis ini adalah sensor suhu DS18B20. Sensor suhu DS18B20 digunakan untuk menampilkan suhu pada dispenser pada air dingin dan air panas yang nantinya ditampilkan pada LCD. *Limit*

switch digunakan untuk mendeteksi ada tidaknya gelas di bawah keran sehingga nantinya akan mengaktifkan *solenoid valve*.

Sebagai pengendali utama sistem digunakan mikrokontroler ATmega 16. Mikrokontroler berfungsi untuk memproses masukan dari sensor, *limit switch* dan *keypad* menjadi sinyal kontrol pembuka dan penutup *solenoid valve*.

Bagian keluaran dari sistem ini berupa *solenoid valve* membuka dan menutup keran serta LCD sebagai tampilan sistem ini. Agar keluaran *valve* dapat dikontrol oleh mikrokontroler digunakan *driver solenoid valve*.

4.2 Perancangan Perangkat Keras

Perancangan perangkat keras meliputi perancangan minimum sistem mikrokontroler, *keypad* dan *driver solenoid valve*.

4.2.1 Perancangan Rangkaian Mikrokontroler ATmega16

Minimum sistem adalah suatu standar minimal rangkaian yang digunakan supaya suatu perangkat khususnya mikrokontroler dapat bekerja secara normal dan optimal. Komponen yang terdapat di dalam minimum sistem dapat disesuaikan dengan perangkat apa saja yang terhubung dan bekerja dengan mikrokontroler tersebut. Mikrokontroler yang digunakan pada sistem ini adalah ATmega 16 yang termasuk dalam keluarga AVR. ATmega16 secara *default* bekerja pada frekuensi *clock internal* 1 MHz akan tetapi dapat dimaksimalkan mencapai frekuensi *clock* 16 MHz dengan bantuan rangkaian *oscillator*. Rangkaian *oscillator* terdiri dari sebuah Xtal (Kristal) 16 MHz dan 2 buah kapasitor 22pF *nonpolar*.

Mikrokontroler ATmega16 dilengkapi dengan *port reset* di mana ketika *port* ini diaktifkan akan *me-reset* seluruh proses yang dikerjakan ketika mikrokontroler sedang aktif. *Port* ini perlu tambahan beberapa komponen pendukung agar dapat berfungsi. Komponen tersebut antara lain *push button*, kapasitor dan resistor. *Port reset* ini aktif pada keadaan logika rendah. *Datasheet* mikrokontroler menjelaskan bahwa tegangan yang dianggap logika rendah untuk *reset* berkisar pada jangkauan 0.1 - 0.9 Vcc di mana waktu minimal yang dibutuhkan untuk *reset* 1.5 us. Perhitungan minimal nilai kapasitor yang dibutuhkan supaya mikrokontroler dapat melakukan reset jika resistor yang digunakan bernilai 1 kΩ dan Vcc 5V dengan acuan waktu *reset* dan tegangan minimal kapasitor sesuai dalam *datasheet* ATmega 16 sebagai berikut:

$$\begin{aligned} V_{C(\min)} &= 0,1 \times V_{cc} \\ &= 0,1 \times 5 \text{ V} \\ &= 0,5 \text{ V} \end{aligned}$$

$$Vc = Vs \cdot e^{-\frac{t}{R.C(t)}}$$

$$0,5 = 5 \cdot e^{-\frac{1,5 \times 10^{-6}}{10^3 \cdot C}}$$

$$\frac{0,5}{5} = e^{-\frac{1,5 \times 10^{-6}}{10^3 \cdot C}}$$

$$\ln 0,1 = -1,5 \cdot 10^{-6} / 10^3 C$$

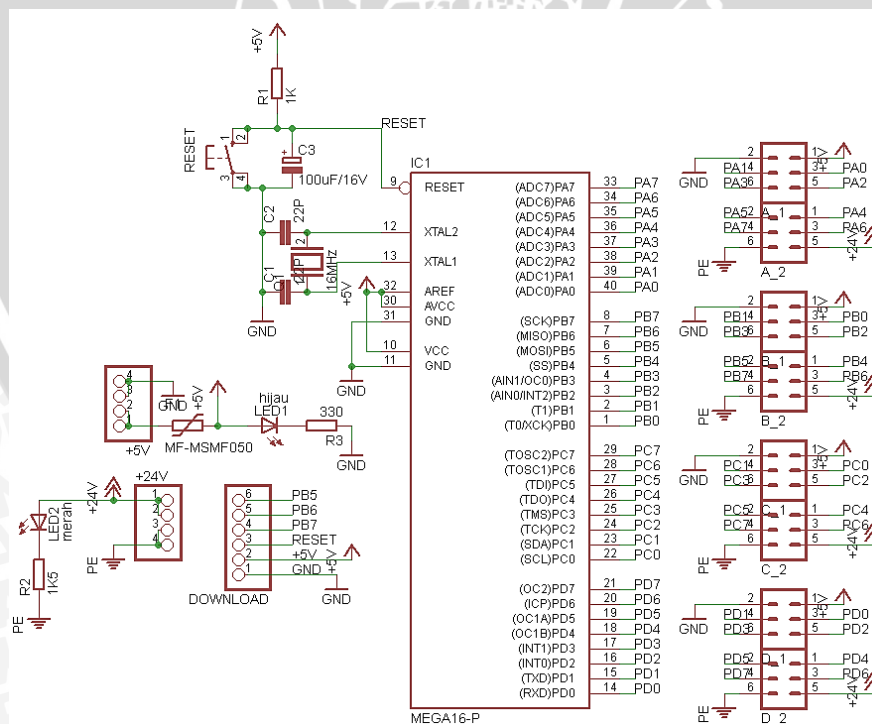
$$-2,3 = -1,5 \cdot 10^{-6} / 10^3 C$$

$$C = \frac{-1,5 \cdot 10^{-6}}{-2,3 \cdot 10^3}$$

$$C = 0,65 \cdot 10^{-9} \text{ F}$$

$$C = 0,65 \text{ nF}$$

Berdasarkan perhitungan di atas diperoleh nilai kapasitor minimal yang dibutuhkan supaya *reset* dapat aktif sebesar 0,65 nF, sehingga dalam perancangan harus menggunakan kapasitor dengan kapasitansi yang lebih besar. Gambar 4.3 menunjukkan skematik rangkaian minimum sistem mikrokontroler ATmega 16 yang digunakan dalam penelitian ini. Mikrokontroler ATmega 16 mempunyai 32 *port* I/O yang dapat dimanfaatkan. Semua *port* tersebut dikelompokkan ke dalam 4 bagian, yaitu *port* A, B, C, dan D di mana masing-masing memiliki 8 buah *port*. Pada perancangan ini *port* yang digunakan adalah:



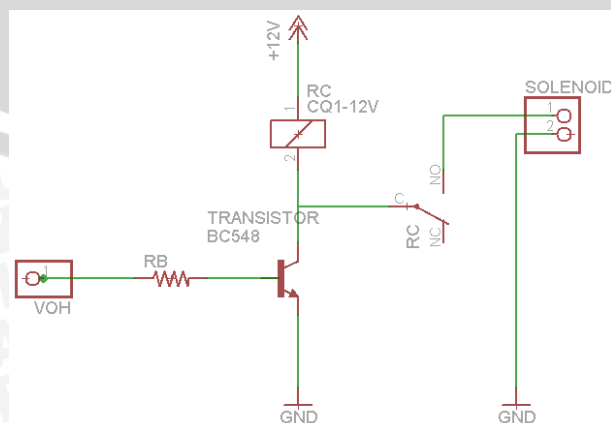
Gambar 4.2 Minimum sistem ATmega16

PORTA.0 = difungsikan untuk menerima sinyal masukan dari sensor suhu DS18B20

- PORTB.0 = digunakan sebagai antarmuka dengan LCD sebagai pin *register select*
- PORTB.1 = digunakan sebagai antarmuka dengan LCD sebagai pin *Read / Write*
- PORTB.2 = digunakan sebagai antarmuka dengan LCD sebagai pin *enable*
- PORTB.4 = digunakan sebagai antarmuka dengan LCD sebagai pin *data bus 4*
- PORTB.5 = digunakan sebagai antarmuka dengan LCD sebagai pin *data bus 5*
- PORTB.6 = digunakan sebagai antarmuka dengan LCD sebagai pin *data bus 6*
- PORTB.7 = digunakan sebagai antarmuka dengan LCD sebagai pin *data bus 7*
- PORTB.2= difungsikan untuk menerima sinyal *echo* yang dikeluarkan oleh sensor *ultrasonic HC-SR04*
- PORTC.0 = digunakan sebagai antarmuka dengan *keypad* sebagai pin *register select*
- PORTC.1 = digunakan sebagai antarmuka dengan *keypad* sebagai pin *Read / Write*
- PORTC.2 = digunakan sebagai antarmuka dengan *keypad* sebagai pin *enable*
- PORTC.3 = digunakan sebagai antarmuka dengan *keypad* sebagai pin *data bus 4*
- PORTC.4 = digunakan sebagai antarmuka dengan *keypad* sebagai pin *data bus 4*
- PORTC.5 = digunakan sebagai antarmuka dengan *keypad* sebagai pin *data bus 5*
- PORTC.6 = digunakan sebagai antarmuka dengan *keypad* sebagai pin *data bus 6*
- PORTC.7 = digunakan sebagai antarmuka dengan *keypad* sebagai pin *data bus 7*
- PORTD1 = difungsikan untuk memberi perintah pengaktifan *solenoid valve* yang ditujukan kepada *Driver solenoid valve*
- PORTD.6 = difungsikan untuk memberi perintah pengaktifan *limit switch* yang ditujukan kepada *Driver solenoid valve*.

4.2.2 Perancangan Driver Solenoid Valve

Solenoid valve dalam perancangan ini merupakan blok *output* yang dikontrol oleh sinyal dari mikrokontroller. *Solenoid valve* yang digunakan dalam perancangan ini adalah SLCI tipe *direct acting* yang dapat mengalirkan air bertekanan rendah.



Gambar 4.3 Rangkaian *driver solenoid valve*

Prinsip kerja dari *driver* ini adalah mengkonversi parameter keluaran dari mikrokontroler menjadi parameter masukan yang sesuai dengan kebutuhan *solenoid valve* yang digunakan. Berdasarkan *datasheet* parameter keluaran yang dikeluarkan oleh mikrokontroler ATmega 16 sebagai berikut:

- $V_{OH} = 4,2 \text{ V}$
- $V_{OL} = 0,7 \text{ V}$
- $I_{OH} = -20 \text{ mA}$
- $I_{OL} = 20 \text{ mA}$

Driver *valve* dapat berfungsi dengan baik jika transistor dapat bekerja pada kondisi saturasi saat logika mikrokontroler tinggi dan pada kondisi *cut off* saat logika mikrokontroler rendah. Merancang transistor bekerja pada kondisi *cut off* saat logika mikrokontroler tidak sulit karena transistor akan bekerja pada *cut off* jika $V_{BE} < 0,7 \text{ V}$ sementara keluaran logika rendah mikrokontroler yang masuk ke terminal basis – emitor transistor maksimal sebesar $0,7 \text{ V}$. Perancangan kondisi saturasi transistor dapat dilakukan dengan memperhitungkan nilai R_B , untuk memperoleh nilai R_B diperlukan parameter – parameter yang nantinya digunakan dalam perhitungan nilai R_B , berdasarkan *datasheet* transistor BC548(Motorolla,1996) sebagai berikut:

$$\beta_{\min} = 110 \qquad V_{BE(\text{sat})} = 0,7 \text{ V}$$

$$V_{CE(\text{sat})} = 0,6 \text{ V}$$

dan *datasheet* untuk relay 12 volt(Omron, 2009) sebagai berikut:

$$R_{\text{KOIL}} = R_C = 360 \ \Omega \qquad V_{CC} = 12 \text{ V}$$

Kemudian dapat diketahui nilai $I_{C(\text{sat})}$ dari persamaan:

$$I_{C(\text{sat})} = \frac{V_{CC} - V_{CE(\text{sat})}}{R_C}$$

$$I_{C(\text{sat})} = \frac{12 \text{ V} - 0,6 \text{ V}}{360 \ \Omega}$$

$$I_{C(\text{sat})} = 33 \text{ mA}$$

Setelah diketahui nilai $I_{C(\text{sat})}$, dapat dihitung nilai $I_{B(\text{sat})}$ dari persamaan:

$$I_{C(\text{sat})} = \beta_{\min} \times I_B$$

$$I_B = \frac{I_{C(\text{sat})}}{\beta_{\min}}$$

$$I_B = \frac{33 \text{ mA}}{110}$$

$$I_B = 0,3 \text{ mA}$$

Nilai R_B yang menyebabkan transistor saturasi saat output mikrokontroler logika tinggi dapat diketahui dari persamaan:

$$I_B = \frac{V_{OH} - V_{BE}}{R_B}$$

$$R_B = \frac{V_{OH} - V_{BE}}{I_B}$$

$$R_B = \frac{4,2 \text{ V} - 0,7 \text{ V}}{0,3 \times 10^{-3} \Omega}$$

$$R_B = 11,67 \text{ k}\Omega$$

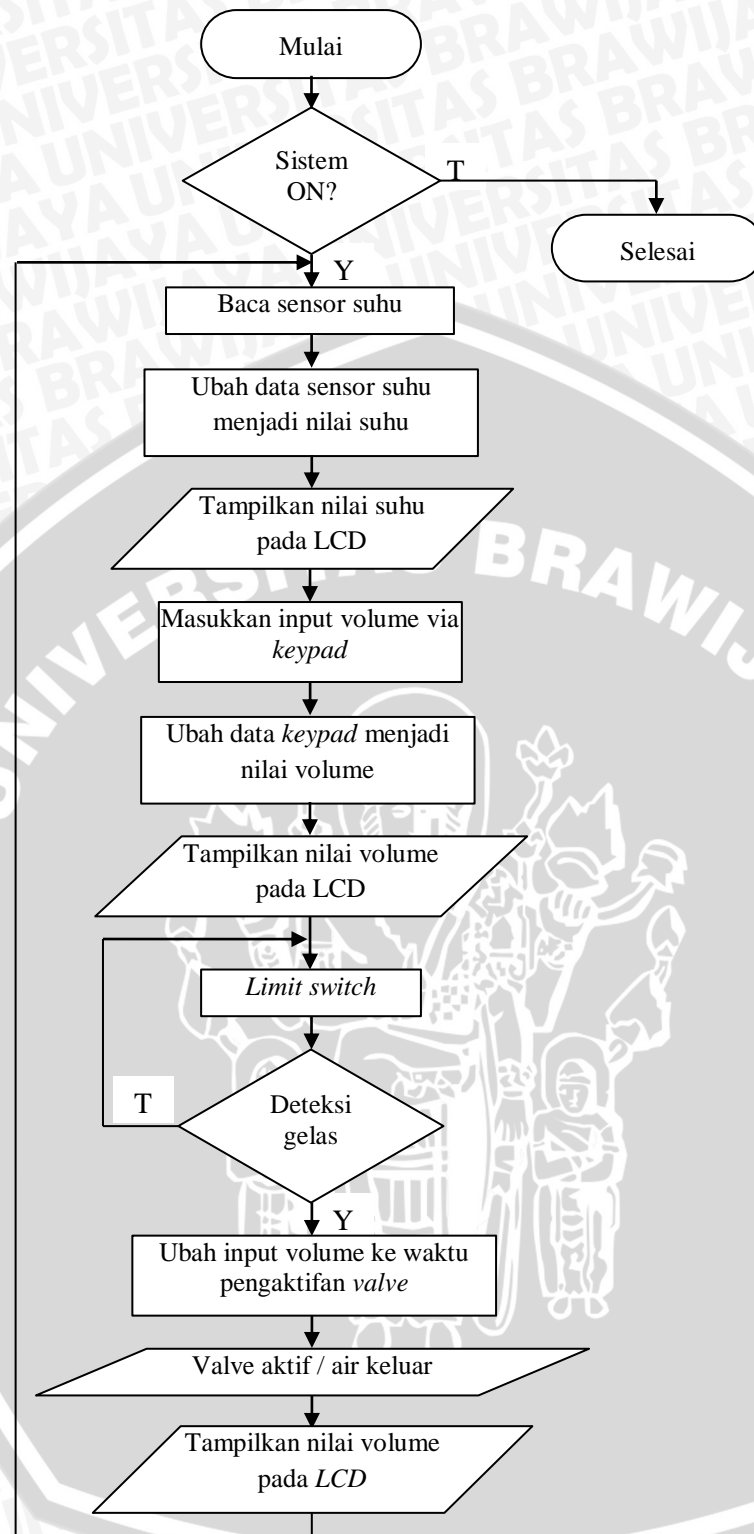
Nilai resistor yang digunakan dalam perancangan ini sebesar 10 k Ω .

Driver relay dapat berfungsi dengan baik jika transistor dapat bekerja pada kondisi saturasi saat logika mikrokontroler tinggi dan pada kondisi *cut off* saat logika mikrokontroler rendah. Merancang transistor bekerja pada kondisi *cut off* saat logika mikrokontroler tidak sulit karena transistor akan bekerja pada *cut off* jika $V_{BE} < 0,7 \text{ V}$ sementara keluaran logika rendah mikrokontroler yang masuk ke terminal basis – emitor transistor maksimal sebesar 0,7 V.

Nilai R_B hasil perhitungan merupakan nilai maksimal yang dapat digunakan agar transistor saturasi ketika port mikrokontroler logika tinggi, jika nilai resistor yang digunakan melebihi nilai R_B maka transistor tidak aktif karena arus I_B yang mengalir semakin kecil atau di bawah batas saturasi, oleh karena itu digunakan resistor sebesar 10 k Ω .

4.3 Perancangan Perangkat Lunak

Perancangan perangkat lunak pada mikrokontroler diawali dengan merancang diagram alir (*flowchart*) terlebih dahulu. Diagram alir ini berfungsi sebagai alur kerja masing-masing perangkat keras yang dikendalikan oleh mikrokontroler maupun proses kalkulasi yang terjadi di dalam mikrokontroler itu sendiri. Bahasa pemrograman yang digunakan dalam membuat program adalah bahasa C dan *compiler* yang digunakan adalah Code Vision AVR 2.5. Diagram alir perancangan perangkat lunak pada sistem mikrokontroler ditunjukkan dalam Gambar 4.4.



Gambar 4.4 Flowcart sistem keseluruhan

Gambar 4.4 di atas menjelaskan proses sistem secara keseluruhan, setelah sistem ON pertama yang dilakukan adalah membaca suhu pada dispenser melalui sensor suhu DS18B20, kemudian akan diolah pada mikrokontroler dan ditampilkan pada LCD dalam satuan ($^{\circ}\text{C}$). Proses selanjutnya yaitu masukkan volume yang diinginkan dengan menggunakan *interface keypad* setelah proses tersebut dilakukan, jika ingin

mengeluarkan air sesuai volume yang telah disetting, kita hanya menekan salah satu tombol keypad (A,B,C,D). Setelah itu jika gelas belum diletakkan dibawah keran (gelas tidak menekan *limit switch*) maka program belum berjalan dan ditampilkan pada LCD bahwa “ tidak ada gelas”, tetapi jika gelas sudah berada tepat dibawah keran (gelas menekan *limit switch*) maka program akan mengaktifkan *driver solenoid valve* sehingga keran terbuka dengan volume yang keluar sesuai dengan volume yang telah diatur pada proses sebelumnya serta ditampilkan pada LCD.

Perhitungan untuk volume yang diatur melalui keypad menjadi lama waktu pengaktifan valve pada sistem ini yaitu sesuai dengan rumus :

$$\text{waktu} = \frac{\text{volume via keypad}}{\text{debit air}}$$

untuk menentukan nilai debit sendiri, pada perancangan ini yaitu dengan mengukur volume air yang keluar dari *solenoid valve* kemudian nantinya didapatkan nilai debit yang digunakan sebagai parameter pada perancangan ini, untuk menentukan nilai debit sendiri dapat dilihat pada tabel 4.1.

Tabel 4.1 hasil pengukuran volume untuk mencari nilai debit

Waktu pengaktifan	Volume hasil pengukuran ke-			Rata – rata volume pengujian
	Pengujian 1	Pengujian 2	Pengujian 3	
10 detik	250 ml	250 ml	250 ml	250 ml
5 detik	125 ml	125 ml	125 ml	125 ml

Nilai debit dapat dicari setelah diketahui nilai dari volume pengukuran dan waktu pengaktifan, sesuai dengan persamaan maka:

$$\text{waktu} = \frac{\text{volume via keypad}}{\text{debit air}}$$

$$\text{debit air} = \frac{\text{volume via keypad}}{\text{waktu}}$$

untuk waktu pengaktifan valve selama 10 detik didapatkan nilai volume rata – rata sebesar 250 ml, sedangkan untuk waktu pengaktifan valve selama 5 detik didapatkan volume rata – rata sebesar 125 ml atau setengah dari volume dengan waktu pengaktifan selama 10 detik, dari hal tersebut dapat dicari nilai debit air sesuai persamaan:

$$\text{debit air} = \frac{250 \text{ ml}}{10 \text{ s}}$$

$$\text{debit air} = 25 \text{ ml/s}$$

nilai dari hasil pengukuran tersebut sebesar 25 ml/s nantinya digunakan sebagai parameter pada perancangan ini.