

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Sebelumnya telah ada alat *Centrifuge Balance* dengan merek CENTRON model CB220 yang diproduksi oleh Centron Corporation, alat tersebut memiliki fungsi untuk menimbang kantong darah sebelum diputar secara presisi menggunakan *centrifuge*, dengan spesifikasi 2 *load cell* 2,5 kg dan menggunakan LED untuk indikator *balance*.

Berdasarkan jurnal yang pertama yaitu mengenai “Pengembangan timbangan buah berbasis ATMega16” yang dibuat oleh Arif tri rahmawanto, alat ini memiliki spesifikasi *load cell* 5 kg, menggunakan ATMega16 sebagai mikrokontroler, dan menggunakan penguat instrumentasi seri AMP04, penguat jenis ini masih membutuhkan ADC *external* sebelum masuk ke mikrokontroler [3].

Kemudian jurnal yang kedua yaitu “Rancang bangun Timbangan buah dengan keluaran berat dan harga” yang dibuat oleh Try utami hidayani, alat ini memiliki spesifikasi *load cell* 5 kg, dan menggunakan ATMega32 sebagai mikrokontroler [4].

Dari beberapa referensi diatas, penulis ingin merancang *Centrifuge Balance* namun berbeda dengan alat yang sudah ada, penulis akan membuat *Centrifuge Balance* yang bisa berfungsi sebagai penimbang sampel yang ada di kuvet sebelum diputar di alat *Centrifuge*. Dengan menggunakan 2 *load cell* 1kg sebagai sensor, ATMega8 sebagai mikroprosesor, dan juga menggunakan *buzzer* sebagai

indikator *balance*. Alat ini juga penulis lengkapi dengan LCD sebagai alat yang menampilkan hasil penimbangan sampel.

2.1 Dasar Teori

2.2.1 Mikrokontroler ATmega8

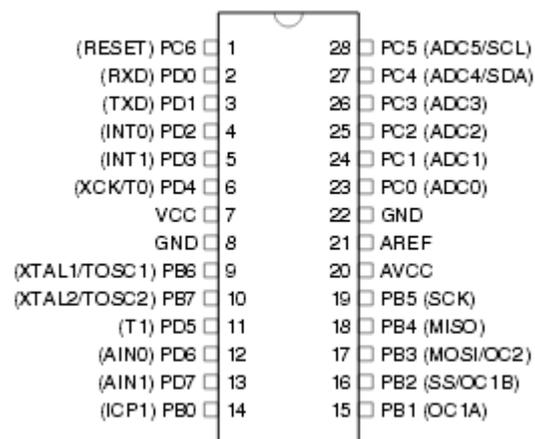
Mikrokontroler Sebagai teknologi baru, yaitu teknologi semikonduktor dengan kandungan transistor yang lebih banyak namun hanya membutuhkan ruang kecil serta dapat diproduksi secara masal (dalam jumlah banyak) membuat harganya menjadi lebih murah (dibanding mikroprosesor).

AVR merupakan salah satu jenis mikrokontroler yang di dalamnya terdapat berbagai macam fungsi. Perbedaannya pada mikro yang pada umumnya digunakan seperti MCS51 adalah pada AVR tidak perlu menggunakan *oscillator* eksternal karena di dalamnya sudah terdapat internal *oscillator*. Selain itu kelebihan dari AVR adalah memiliki *Power-On Reset*, yaitu tidak perlu ada tombol *reset* dari luar karena cukup hanya dengan mematikan *supply*, maka secara otomatis AVR akan melakukan *reset*. Untuk beberapa jenis AVR terdapat beberapa fungsi khusus seperti ADC, EEPROM sekitar 128 *byte* sampai dengan 512 *byte*.

AVR ATmega8 adalah mikrokontroler CMOS 8-Bit berarsitektur AVR RISC yang memiliki 8K*byte* in *System Programmable Flash*. Mikrokontroler dengan konsumsi daya rendah ini mampu mengeksekusi instruksi dengan kecepatan maksimum 16MIPS pada frekuensi 16MHz. Jika dibandingkan dengan ATmega8L perbedaannya hanya terletak pada besarnya tegangan yang diperlukan untuk bekerja. Untuk ATmega8 tipe L, mikrokontroler ini dapat bekerja dengan

tegangan antara 2,7 - 5,5 V sedangkan untuk ATmega8 hanya dapat bekerja pada tegangan antara 4,5 – 5,5 V.

ATmega8 memiliki 28 Pin, yang masing-masing pin nya memiliki fungsi yang berbeda - beda baik sebagai *port* maupun fungsi yang lainnya. Berikut akan dijelaskan fungsi dari masing-masing kaki ATmega8. Gambar konfigurasi pin ATmega8 ditunjukkan pada gambar 2.1.



Gambar 2.1 Konfigurasi Pin Atmega 8

a. VCC

Merupakan *supply* tegangan digital.

b. GND

Merupakan *ground* untuk semua komponen yang membutuhkan *grounding*.

c. Port B (PB7...PB0)

Didalam Port B terdapat XTAL1, XTAL2, TOSC1, TOSC2. Jumlah Port B adalah 8 buah pin, mulai dari pin B.0 sampai dengan B.7. Tiap pin dapat digunakan sebagai *input* maupun *output*. Port B merupakan sebuah 8- Bit

bidirectional I/O dengan *internal pull-up* resistor. Sebagai *input*, pin-pin 7 yang terdapat pada *port B* yang secara *eksternal* diturunkan, maka akan mengeluarkan arus jika *pull-up* resistor diaktifkan. Khusus PB6 dapat digunakan sebagai *input* kristal (*inverting oscillator amplifier*) dan *input* ke rangkaian *clock internal*, bergantung pada pengaturan *Fuse* bit yang digunakan untuk memilih sumber *clock*. Sedangkan untuk PB7 dapat digunakan sebagai *output* kristal (*output oscillator amplifier*) bergantung pada pengaturan *Fuse* bit yang digunakan untuk memilih sumber *clock*. Jika sumber *clock* yang dipilih dari *oscillator internal*, PB7 dan PB6 dapat digunakan sebagai I/O atau jika menggunakan *Asynchronous Timer/Counter 2* maka PB6 dan PB7 (TOSC2 dan TOSC1) digunakan untuk saluran *input timer*.

d. *Port C (PC5...PC0)*

Port C merupakan sebuah 7- Bit *bi-directional I/O port* yang di dalam masing-masing pin terdapat *pull-up* resistor. Jumlah pin nya hanya 7 buah mulai dari pin C.0 sampai dengan pin C.6. Sebagai keluaran/*output port C* memiliki karakteristik yang sama dalam hal menyerap arus (*sink*) ataupun mengeluarkan arus (*source*).

e. *RESET/PC6*

Jika *RSTDISBL Fuse* diprogram, maka PC6 akan berfungsi sebagai pin I/O. Pin ini memiliki karakteristik yang berbeda dengan pin-pin yang terdapat pada *port C* lainnya. Namun jika *RSTDISBL Fuse* tidak diprogram, maka pin ini akan berfungsi sebagai *input reset*. Jika level

tegangan yang masuk ke pin ini rendah dan pulsa yang ada lebih pendek dari pulsa 8 minimum, maka akan menghasilkan suatu kondisi *reset* meskipun *clock*-nya tidak bekerja.

f. *Port D (PD7...PD0)*

Port D merupakan 8-Bit *bi-directional I/O* dengan *internal pull-up* resistor. Fungsi dari *port* ini sama dengan *port-port* yang lain. Hanya saja pada *port* ini tidak terdapat kegunaan-kegunaan yang lain. Pada *port* ini hanya berfungsi sebagai masukan dan keluaran saja atau biasa disebut dengan *I/O*.

g. *AVcc*

Pin ini berfungsi sebagai *supply* tegangan untuk ADC. Untuk pin ini harus dihubungkan secara terpisah dengan *VCC* karena pin ini digunakan untuk analog saja. Bahkan jika ADC pada AVR tidak digunakan tetap saja disarankan untuk menghubungkannya secara terpisah dengan *VCC*. Jika ADC digunakan, maka *AVcc* harus dihubungkan ke *VCC* melalui *low pass filter*.

h. *AREF*

Merupakan pin referensi jika menggunakan ADC.

Pada AVR status *register* mengandung beberapa informasi mengenai hasil dari kebanyakan hasil eksekusi instruksi aritmatik. Informasi ini digunakan untuk *altering* arus program sebagai kegunaan untuk meningkatkan performa pengoperasian. *Register* ini di-*update* setelah operasi ALU (*Arithmetic Logic*

Unit) hal tersebut seperti yang tertulis dalam *datasheet* khususnya pada bagian *Instruction Set Reference*. Dalam hal ini untuk beberapa kasus dapat membuang penggunaan kebutuhan instruksi perbandingan yang telah didedikasikan serta dapat menghasilkan peningkatan dalam hal kecepatan dan kode yang lebih sederhana dan singkat. *Register* ini tidak secara otomatis tersimpan ketika memasuki sebuah rutin interupsi dan juga ketika menjalankan sebuah perintah setelah kembali dari interupsi. Namun hal tersebut harus dilakukan melalui *software*. Gambar status *register* ditunjukkan pada gambar 2.2.

Bit	7	6	5	4	3	2	1	0	
	I	T	H	S	V	N	Z	C	SREG
Read/write	R/W								
Initial Value	0	0	0	0	0	0	0	0	

Gambar 2.2 Status Register ATmega8

a. Bit 7 (I)

Merupakan Bit *Global Interrupt Enable*. Bit ini harus di-*set* agar semua perintah interupsi dapat dijalankan. Untuk perintah interupsi individual akan di jelaskan pada bagian yang lain. Jika Bit ini di-*reset*, maka semua perintah interupsi baik yang individual maupun yang secara umum akan di abaikan. Bit ini akan dibersihkan atau *cleared* oleh *hardware* setelah sebuah interupsi di jalankan dan akan di-*set* kembali oleh perintah RETI. Bit ini juga dapat diset dan di-*reset* melalui aplikasi dan intruksi SEI dan CLL.

b. Bit 6 (T)

Merupakan *Bit Copy Storage*. Instruksi *Bit Copy Instructions* BLD (*Bit Load*) and BST (*Bit Store*) menggunakan Bit ini sebagai asal atau tujuan untuk Bit yang telah dioperasikan. Sebuah Bit dari sebuah *register* dalam *Register File* dapat disalin ke dalam Bit ini dengan menggunakan instruksi BST, dan sebuah Bit di dalam Bit ini dapat disalin ke dalam Bit di dalam *register* pada *Register File* dengan menggunakan perintah BLD.

c. Bit 5 (H)

Merupakan Bit *Half Carry Flag*. Bit ini menandakan sebuah *Half Carry* dalam beberapa operasi aritmatika. Bit ini berfungsi dalam aritmatika BCD. Bit 4 (S) Merupakan *Sign Bit*. Bit ini selalu merupakan sebuah eksklusif di antara *Negative Flag* (N) dan *two's Complement Overflow Flag* (V).

d. Bit 3 (V)

Merupakan Bit *Two's Complement Overflow Flag*. Bit ini menyediakan fungsi aritmatika dua komplemen.

e. Bit 2 (N)

Merupakan Bit *Negative Flag*. Bit ini mengindikasikan sebuah hasil *negative* di dalam sebuah fungsi logika atau aritmatika.

f. Bit 1 (Z)

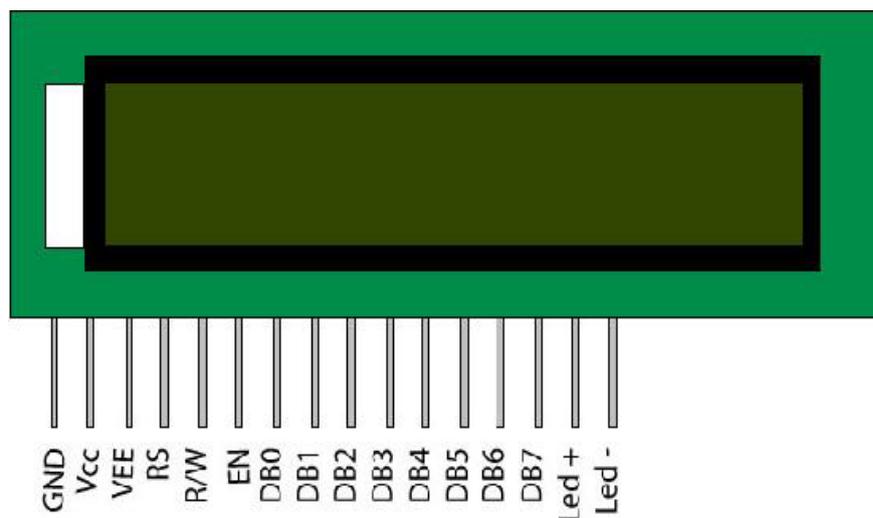
Merupakan Bit *Zero Flag*. Bit ini mengindikasikan sebuah hasil nol "0" dalam sebuah fungsi aritmatika atau logika.

g. Bit 0 (C)

Merupakan Bit *Carry Flag*. Bit ini mengindikasikan sebuah *carry* atau sisa dalam sebuah aritmatika atau logika [5].

2.2.2 *Liquid Crystal Display (LCD)*

LCD (Liquid Crystal Display) adalah sebuah *display* yang menggunakan kristal cair sebagai penampil utama. *LCD* dibagi menjadi dua bagian yaitu bagian depan panel *LCD* yang terdiri dari banyak *dot* atau titik *LCD* dan *microcontroller* yang menempel di bagian belakang panel *LCD* yang fungsinya untuk mengatur titik-titik *LCD* sehingga dapat menampilkan huruf, angka, dan simbol khusus yang dapat terbaca. Gambar *LCD* beserta konfigurasi pin ditunjukkan pada gambar 2.3.



Gambar 2.3 *LCD* karakter

LCD 16 x 2 memiliki beberapa pin. Diantara pin *LCD* 16 x 2 tersebut adalah EN, RS, RW, yaitu untuk jalur EN dinamakan *enable*. Jalur ini difungsikan untuk memberitahu *LCD* bahwa anda sedang mengirimkan sebuah data. Untuk mengirimkan data ke *LCD*, maka melalui program EN harus dibuat logika *low* “0” dan *set* pada dua jalur kontrol yang lain RS dan RW. Ketika dua jalur yang lain

telah siap, *set* EN dengan logika “1” dan tunggu untuk sejumlah waktu tertentu (sesuai dengan *datasheet* dari *LCD* tersebut) dan berikutnya *set* EN ke logika *low* “0” lagi.

Kemudian untuk jalur RS adalah jalur *register select*. Ketika RS berlogika *low* “0”, data akan dianggap sebagai sebuah perintah atau instruksi khusus (seperti *clear screen*, posisi kursor, dll). Ketika RS berlogika *high* “1”, data yang dikirim adalah *data text* yang akan ditampilkan pada *display LCD*.

Selanjutnya yang terakhir jalur RW adalah jalur kontrol *Read/Write*. Ketika RW berlogika *low* (0), maka informasi pada *bus data* akan dituliskan pada layar *LCD*. Ketika RW berlogika *high* ”1”, maka program akan melakukan pembacaan memori dari *LCD*. Sedangkan pada aplikasi umum pin RW selalu diberi logika *low* ”0” [6].

2.2.3 Sensor Berat (*Load Cell*)

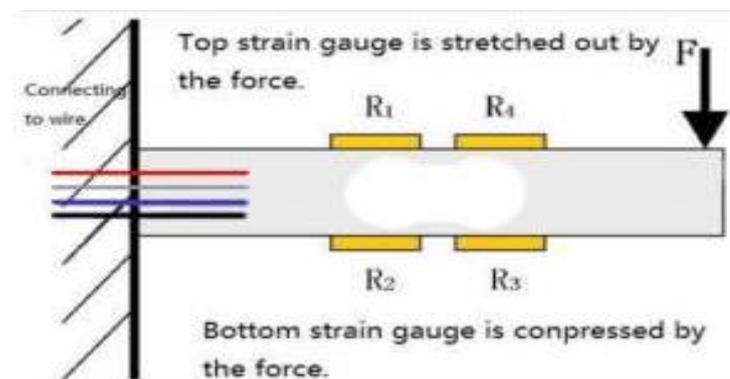
Load cell adalah sensor yang dapat mendeteksi adanya perubahan massa yang ditimbulkan oleh gaya dan gravitasi suatu benda. Perubahan yang ditimbulkan oleh gaya dan gravitasi benda nantinya akan dijadikan sebuah sinyal analog dan akan diteruskan ke transduser. Transduser berfungsi mengubah sinyal analog yang ditimbulkan oleh *load cell* ke besaran listrik.

Menurut Piskorowski et al. dalam Sugriawan *load cell* adalah sebuah sensor gaya yang dihasilkan dari suatu tekanan tertentu. Sensor *load cell* banyak digunakan dalam industri yang memerlukan peralatan untuk mengukur berat.

Menurut Mauselein et al. dalam Sugriawan *Load cell* berisi sebuah pegas (*spring*) logam mekanik dengan mengaplikasikan beberapa *foil metal strain*

gauges (SG). *Strain* dari pegas mekanik muncul sebagai pengaruh dari pemberian beban yang kemudian ditransmisikan pada *strain gauge*. Pengukuran sinyal yang dihasilkan dari *load cell* adalah dari perubahan resistansi *strain gauge* yang linier dengan gaya yang diaplikasikan.

Prinsip kerja *load cell* dihitung dari perubahan resistansi yang terjadi akibat timbulnya sebuah regangan pada *foil metal strain gauge*. Perubahan resistansi diakibatkan oleh pemberian sebuah beban pada sisi yang elastis sehingga mengalami perubahan tekanan sesuai dengan yang dihasilkan oleh *strain gauge*. Dari hasil perubahan tekanan pada beban akan dirubah menjadi tegangan oleh komponen pendukung yang ada. Secara sederhana prinsip kerja *load cell* ditunjukkan pada gambar 2.4.



Gambar 2.4 Prinsip kerja sensor *load cell*

Berdasarkan *datasheet* ketika bagian lain dari sensor yang lebih elastis mendapat tekanan, maka pada sisi lain akan mengalami perubahan regangan yang sesuai dengan yang dihasilkan oleh *strain gauge*, hal ini terjadi karena ada gaya yang seakan melawan pada sisi lainnya. Perubahan nilai resistansi yang diakibatkan oleh perubahan gaya yang diubah menjadi nilai tegangan oleh

rangkaian pengukuran yang ada. Berat dari sebuah objek yang diukur dapat diketahui dengan mengukur besarnya nilai tegangan yang timbul.

2.2.4 IC HX711

HX711 adalah modul timbangan yang memiliki prinsip kerja menguatkan perubahan tegangan yang terukur pada sensor *load cell* dan mengkonversinya ke dalam besaran listrik melalui rangkaian yang ada. HX711 presisi 24-Bit *analog-todigital converter* (ADC) yang didesain untuk sensor timbangan digital (*weight scales*) dan industrial kontrol aplikasi yang terkoneksi dengan sensor jembatan (*bridge sensor*). Modul melakukan komunikasi dengan *computer*/mikrokontroler melalui TTL

Kelebihan dari IC HX711 adalah pada struktur yang sederhana, mudah dalam penggunaan, hasil yang stabil serta memiliki sensitivitas tinggi, dan mampu mengukur perubahan dengan cepat. Dalam aplikasinya IC HX711 digunakan pada bidang *aerospace*, mekanik, elektrik, kimia, konstruksi, farmasi dan lainnya, digunakan untuk mengukur gaya, gaya tekanan, perpindahan, gaya tarikan, torsi, dan percepatan [7].

2.2.5 Buzzer

Buzzer adalah sebuah komponen elektronika yang cara kerjanya dengan mengubah listrik menjadi getaran suara. Pada dasarnya prinsip kerja *buzzer* hampir sama dengan *loud speaker*, jadi *buzzer* juga terdiri dari kumparan yang terpasang pada diafragma dan kemudian kumparan tersebut dialiri arus sehingga menjadi elektromagnet, kumparan tadi akan tertarik ke dalam atau keluar, tergantung dari arah arus dan polaritas magnetnya, karena kumparan dipasang

pada diafragma maka setiap gerakan kumparan akan menggerakkan diafragma secara bolak balik sehingga membuat udara bergetar yang akan menghasilkan suara. *Buzzer* penulis gunakan sebagai tanda kalau sampel sudah *balance* [4].

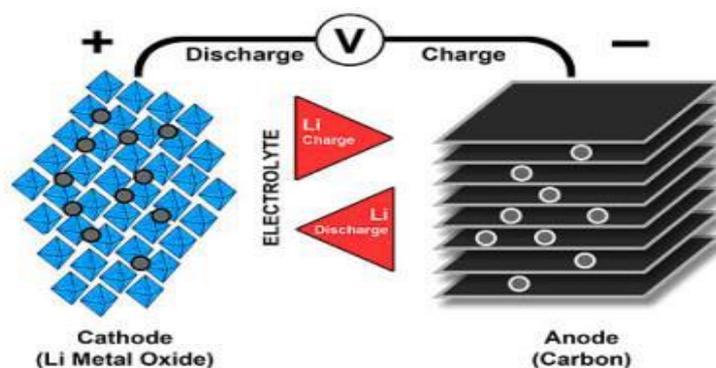
2.2.6 Baterai *Li-Ion Ultrafire*

Baterai adalah alat listrik-kimiawi yang menyimpan energi dan mengeluarkan tenaganya dalam bentuk listrik. Banyak jenis baterai yang bisa kita temui di pasaran, banyak model dan variasi yang dapat kita pilih sesuai dengan kebutuhan kita. Baterai memiliki dua sifat yaitu baterai primer dan baterai sekunder (*rechargeable battery*), disebut baterai primer berarti baterai ini hanya bisa satu kali pakai saja sedangkan baterai sekunder berarti baterai yang dapat dipakai berkali-kali dengan cara isi ulang bila dayanya sudah mulai habis. Penulis menggunakan modul *charger* untuk mengisi ulang baterai sehingga bisa menggunakan *charger smartphone* biasa untuk isi ulang baterai, modul *charger* juga memiliki lampu indikator yang sehingga memudahkan pengguna dalam mengetahui apakah baterai sudah penuh atau belum. Baik baterai primer dan sekunder, kedua-duanya bersifat mengubah energi kimia menjadi energi listrik dan masing-masing memiliki beberapa bentuk dan spesifikasi yang berbeda-beda. Baterai terbagi dalam beberapa jenis, yang salah satunya adalah baterai jenis *Li-Ion (lithium-ion)* yang memiliki sifat sebagai baterai sekunder (*rechargeable battery*). 2 Baterai *Li-Ion* merupakan baterai generasi selanjutnya yang diciptakan setelah baterai tipe NiCd, baterai yang digunakan pada alat ini adalah baterai *Li-On Ultrafire* yang memiliki spesifikasi tegangan sebesar 3,7 V. Gambar ditunjukkan pada gambar 2.5.



Gambar 2.5 Baterai *Li-Ion Ultrafire*

Baterai *Li-Ion* menggunakan katoda (positif), anoda (negatif) dan elektrolit sebagai konduktor, katoda terdiri dari oksida metal dan anoda terdiri dari karbon. Selama baterai bekerja, *ion* mengalir dari anoda ke katoda melalui elektrolit dan pemisah, arah balik pengisian dan *ion* mengalir dari katoda ke anoda ditunjukkan pada gambar 2.6



2.6 Proses Mengalirnya *Ion* Pada *Li-Ion*

Pada saat baterai digunakan atau tidak digunakan, *ion* bergerak diantara katoda (positif elektroda) dan anoda (negatif elektroda). Pada saat baterai tidak

digunakan anoda mengalami oksidasi atau kehilangan elektron dan katoda memperoleh aliran elektron tersebut, pada saat baterai digunakan maka proses yang terjadi adalah kebalikan dari proses ini [6].

2.3 Statistik Data

a. Rata – rata

Penghitungan rata-rata dilakukan dengan menjumlahkan seluruh nilai data suatu kelompok sampel, kemudian dibagi dengan jumlah sampel tersebut. Rata-rata dari sampel tersebut dirumuskan sebagai berikut :

$$\bar{X} = \frac{\sum Xi}{n} \dots\dots\dots(2-1)$$

Keterangan :

\bar{X} = Rata – rata

$\sum Xi$ = Jumlah nilai data

n = Banyaknya data

b. Simpangan

Simpangan adalah perbedaan antara nilai yang sebenarnya dengan nilai yang terukur.

$$Simpangan = Y - X \dots\dots\dots(2-2)$$

Keterangan :

Y = Nilai yang sebenarnya

X = Nilai yang terukur