

**ALAT PENGHITUNG DAN PEMILAH KERUPUK PUTIH  
PADA *HOME INDUSTRY* BERBASIS MIKROKONTROLER**



**ACHMAD MAULANA  
5215125336**

**Skripsi Ini Ditulis Untuk Memenuhi Sebagian Persyaratan  
Dalam Memperoleh Gelar Sarjana Pendidikan**

**PROGRAM STUDI PENDIDIKAN TEKNIK ELEKTRONIKA  
FAKULTAS TEKNIK UNIVERSITAS NEGERI JAKARTA**

**2017**

## HALAMAN PENGESAHAN

NAMA DOSEN	TANDA TANGAN	TANGGAL
<u>Dr. Muhammad Yusro, MT</u> (Pembimbing 1)		16/8/2017
<u>Drs. Pitoyo Yuliatmojo, MT</u> (Pembimbing 2)		18/8/2017

## PENGESAHAN PANITIA UJIAN SKRIPSI

NAMA DOSEN	TANDA TANGAN	TANGGAL
<u>Dr. Moch. Sukardjo, M.Pd</u> (Ketua)		16/8-2017
<u>Dr. Baso Maruddani, MT</u> (Sekretaris)		16/8-17
<u>Aodah Diamah, M.Eng</u> (Dosen Ahli)		14/8/2017

Tanggal Lulus : 8/8/2017

## HALAMAN PERNYATAAN

Dengan ini saya menyatakan bahwa:

1. Karya tulis skripsi saya ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik sarjana, baik di Universitas Negeri Jakarta maupun di perguruan tinggi negeri lain.
2. Karya tulis ini adalah murni gagasan, rumusan dan penelitian saya sendiri dengan arahan dosen pembimbing.
3. Dalam karya tulis ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini, serta sanksi lainnya sesuai dengan norma yang berlaku di Universitas Negeri Jakarta.

Jakarta, Agustus 2017

Yang Menyerahkan



1310245736592/336  
6000  
DAN BILANGAN  
Achmad Maulana

5215125336

## KATA PENGANTAR

Segala puji serta syukur saya panjatkan ke hadirat Allah SWT. Yang telah memberikan rahmat, karunia, hidayah, dan pertolongan-Nya sehingga saya dapat menyelesaikan skripsi dengan judul “Alat Penghitung Dan Pemilah Kerupuk Putih Pada *Home Industry* Berbasis Mikrokontroler” yang merupakan syarat dalam menyelesaikan studi untuk meraih gelar sarjana pendidikan di jurusan teknik elektro fakultas teknik universitas negeri Jakarta.

Selama melakukan penelitian ini penulis mengucapkan terima kasih kepada:

1. Bapak Drs. Pitoyo Yuliatmojo, MT. selaku koordinator program studi dan dosen pembimbing 2 yang memberikan solusi dan pendapat dalam menyelesaikan skripsi ini.
2. Bapak Dr. Muhammad Yusro, MT. selaku dosen pembimbing 1 yang telah memberikan arahan dalam menyelesaikan skripsi ini.
3. Keluarga tercinta, yang telah memberikan semangat dan dukungan baik moril maupun materi.
4. Teman-teman Jurusan Teknik Elektronika yang memberikan bantuan dan semangat kepada penulis.

Semoga Allah SWT. Membalas kebaikan semua pihak yang telah membantu dan meringankan dalam penyusunan skripsi ini. Penulis menyadari bahwa banyak kekurangan dalam penyusunan skripsi ini, namun peneliti berharap semoga penelitian ini dapat bermanfaat bagi pembacanya.

Jakarta, Agustus 2017

Penulis,

Achmad Maulana

## ABSTRAK

**Achmad Maulana**, Alat Penghitung Dan Pemilah Kerupuk Putih Pada *Home Industry* Berbasis Mikrokontroler. Skripsi. Jakarta, Program Studi Pendidikan Teknik Elektronika, Fakultas Teknik, Universitas Negeri Jakarta, 2017.

Penelitian ini bertujuan untuk merancang bangun sistem penghitung kerupuk putih berbasis mikrokontroler.

Metode yang digunakan dalam penelitian ini adalah metode rekayasa teknik yaitu membuat rancangan alat serta melakukan uji coba berupa program yang kemudian akan diterapkan pada alat penghitung dan pemilah kerupuk putih pada *home industry* berbasis mikrokontroler. Pada perancangan penelitian ini memiliki beberapa bentuk umum antara lain perancangan perangkat keras berupa sensor *infrared*, sensor mekanik, motor *gearbox*, LCD. Dan menggunakan perangkat lunak Arduino IDE untuk memprogram alat agar dapat berfungsi sesuai dengan rancangan.

Alat penghitung dan pemilah kerupuk putih pada *home industry* berbasis mikrokontroler ini sudah melalui tahap pengujian, yaitu tahap pengujian setiap sensor sebagai *input* dan aktuator yang menjalankan mikrokontroler. Sehingga dapat diketahui bahwa alat penghitung kerupuk putih ini dapat memisahkan kerupuk yang bentuknya tidak sempurna serta dapat meringankan pekerjaan pada *home industry* kerupuk putih.

Kata Kunci: Alat Penghitung dan Pemilah, Kerupuk Putih, *Home Industry*, Mikrokontroler

## **ABSTRACT**

**Achmad Maulana**, *White Cracker Counter And Sorter Machine At Home Industry Based Microcontroller. Essay. Jakarta : Electronics Education Studies Program, Faculty of Engineering, State University of Jakarta, 2017.*

*This study aims to design the build system of white crackers counter machine based microcontroller.*

*The method used in this research is the method that makes the design engineering and testing tools are programs that will be applied to the White Cracker Counter And Sorter Machine At Home Industry Based Microcontroller. In this study design has several common forms include hardware design in the form of infrared sensors, mechanical sensors, motors gearbox, LCD, and using Arduino IDE to program the machine to work in accordance with the draft.*

*This machine has been through the testing phase of sensor and actuator which executes the command Arduino. So that it can run counting and sorting the white crackers and can alleviate the work on home industry of white crackers.*

*Keywords: counter and sorter machine, white cracker, home industry microcontroller*

## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN PENGESAHAN.....	ii
HALAMAN PERNYATAAN.....	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
<i>ABSTRACT</i> .....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR.....	x
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Identifikasi Masalah .....	3
1.3 Pembatasan Masalah .....	4
1.4 Rumusan Masalah .....	4
1.5 Tujuan Penelitian .....	4
1.6 Manfaat Penelitian .....	4
BAB II TINJAUAN PUSTAKA.....	6
2.1 Kerangka Teori.....	6
2.1.1 Kerupuk Putih.....	6
2.1.2 Arduino.....	7
2.1.3 Bahasa Pemrograman <i>Visual Basic</i> .NET 2010 .....	12
2.1.4 <i>Sensor Photo Infrared</i> .....	14
2.1.5 <i>Relay</i> .....	<b>Error! Bookmark not defined.</b>
2.1.6 Motor DC <i>Gearbox</i> .....	15
2.1.7 <i>Buzzer</i> .....	16
2.1.8 LCD 2x16 .....	17
2.2 Kerangka Berpikir.....	19
2.2.1 Blok Diagram .....	20
2.2.2 Alur Kerja Sistem .....	21
BAB III METODOLOGI PENELITIAN.....	23
3.1 Tempat dan Waktu Penelitian .....	22

3.2 Alat dan Bahan Penelitian .....	22
3.3 Diagram Alir Penelitian .....	23
3.3.1 Perancangan Penelitian .....	28
3.4 Teknik dan Prosedur Pengumpulan Data .....	33
3.5 Teknik Analisis Data .....	34
3.5.1 Pengujian <i>Powersupply</i> .....	34
3.5.2 Pengujian <i>Sensor Photo Infrared</i> .....	35
3.5.3 Pengujian Rangkaian <i>Driver Relay</i> .....	35
3.5.4 Pengujian LCD .....	36
3.5.5 Pengujian Maket .....	36
BAB IV HASIL PENELITIAN .....	37
4.1 Deskripsi Hasil Penelitian .....	37
4.1.1 Prinsip Kerja Alat .....	37
4.1.2 Langkah Kerja Alat .....	37
4.2 Analisis Hasil Penelitian .....	38
4.2.1 Hasil pengujian <i>powersupply</i> .....	42
4.2.2 Hasil pengujian <i>sensor infrared</i> .....	42
4.2.3 Hasil Pengujian Rangkaian <i>Driver Relay</i> .....	44
4.2.4 Hasil Pengujian LCD .....	45
4.2.5 Hasil Pengujian Maket .....	46
4.3 Pembahasan .....	48
4.4 Aplikasi Hasil Penelitian .....	49
BAB V KESIMPULAN DAN SARAN .....	50
5.1 Kesimpulan .....	50
5.1 Saran .....	50
DAFTAR PUSTAKA .....	51
LAMPIRAN .....	52



## DAFTAR TABEL

Tabel 2.1 Keterangan <i>Pin</i> ICSP Pada Arduino UNO .....	9
Tabel 2.2 Versi <i>Visual basic</i> .....	13
Tabel 3.1 <i>Input</i> Arduino UNO .....	32
Tabel 3.2 <i>Output</i> Arduino UNO .....	33
Tabel 3.3 Pengujian <i>Powersupply</i> .....	35
Tabel 3.4 Pengujian <i>Sensor Infrared</i> .....	35
Tabel 3.5 Pengujian <i>Driver Relay</i> .....	35
Tabel 3.6 Pengujian LCD.....	36
Tabel 3.7 Pengujian Maket .....	36
Tabel 4.1 Hasil Pengujian <i>Powersupply</i> .....	42
Tabel 4.2 Hasil Pengujian <i>Sensor Infrared</i> .....	43
Tabel 4.3 Hasil Pengujian Rangkaian <i>Driver Relay</i> .....	44
Tabel 4.4 Hasil Pengujian LCD 16x2 .....	46
Tabel 4.5 Hasil Pengujian Maket .....	47
Tabel 4.6 Persentase <i>Error</i> Penghitungan Alat.....	47

## DAFTAR GAMBAR

Gambar 2.1 kerupuk putih yang telah dikeringkan dan yang sudah digoreng.....	6
Gambar 2.2 Arduino UNO.....	8
Gambar 2.3 Tampilan Awal Arduino IDE 1.6.12.....	11
Gambar 2.4 Software IDE Arduino.....	12
Gambar 2.5 Simbol (a) <i>Photodiode</i> , (b) <i>Infrared LED</i> .....	14
Gambar 2.6 <i>Relay</i> .....	15
Gambar 2.7 Motor DC <i>Gearbox</i> .....	16
Gambar 2.8 Bentuk Dan Simbol <i>Buzzer</i> .....	17
Gambar 2.9 LCD 2x16.....	17
Gambar 2.10 Blok diagram sistem alat.....	20
Gambar 3.1 <i>Flowchart</i> Perancangan Alat.....	23
Gambar 3.2 <i>Flowchart</i> Cara Kerja Alat.....	24
Gambar 3.3 <i>Flowchart</i> Program.....	27
Gambar 3.4 <i>Sensor Photodiode Infrared</i> .....	29
Gambar 3.5 <i>Powersupply Switching</i> .....	30
Gambar 3.6 <i>Driver Relay 2 Channel</i> .....	30
Gambar 3.7 Disain Alat Penghitung Kerupuk (a) Tampak Atas, (b) Tampak Depan, (c) Tampak Samping.....	31
Gambar 4.1 Diagram <i>Wiring</i> Alat.....	39
Gambar 4.2 GUI ( <i>Graphical User Interface</i> ) Alat.....	40
Gambar 4.3 Alat Penghitung Kerupuk Putih Berbasis Mikrokontroler.....	41
Gambar 4.4 <i>Console</i> Alat.....	41
Gambar 4.5 <i>Testing</i> Program <i>Sensor Photo infrared</i> .....	43
Gambar 4.6 <i>Testing</i> Program <i>Relay</i> .....	44
Gambar 4.7 <i>Testing</i> Program LCD.....	45
Gambar 4.8 <i>Testing</i> Program Maket.....	46

## DAFTAR LAMPIRAN

LAMPIRAN 1 <i>Data List Code</i> .....	52
LAMPIRAN 2 Panduan Penggunaan Alat Pemilah Dan Penghitung Kerupuk Putih Berbasis Mikrokontroler.....	59

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Perkembangan ilmu pengetahuan dan teknologi pada zaman modern ini telah mengalami peningkatan yang sangat pesat. Seiring berkembangnya teknologi, istilah sensor pun sering terdengar di telinga kita, sensor adalah alat untuk mendeteksi ataupun mengukur sesuatu, yang digunakan untuk mengubah variasi mekanis, magnetis, panas, sinar, dan kimia menjadi tegangan dan arus listrik. Dalam lingkungan sistem pengendali robotika, sensor memberikan kesamaan yang menyerupai mata, pendengaran, hidung, lidah yang kemudian akan diolah oleh kontroler sebagai otaknya (Petruzella, 2001). Keadaan seperti ini mengalami dampak yang besar pada semua bidang kehidupan manusia terutama pada bidang industri.

Di dalam dunia industri makanan, khususnya dalam proses produksi kerupuk putih, penghitungan jumlah produksi dan pemisahan kerupuk mentah (atau kerupuk yang belum digoreng) dengan yang rusak atau cacat produksi masih dilakukan secara *manual*, sehingga membutuhkan waktu yang lama dan kurang akurat dalam pemilihan kerupuk putih tersebut. Sebenarnya jika proses produksi tersebut dilakukan secara otomatis akan menguntungkan bagi perusahaan yang bersangkutan dan pekerja itu sendiri. Ini dikarenakan bahwa dengan otomatisasi dalam produksi akan menimbulkan proses yang membutuhkan waktu yang lebih singkat dan lebih akurat.

Dalam perhitungan jumlah produksi dan pemisahan kerupuk putih yang cacat produksi secara otomatis pada sebuah alat penghitung kerupuk putih yang menggunakan mikrokontroler. Menurut Winoto (2010, 3) mikrokontroler adalah sebuah sistem mikroprosesor yang di dalamnya sudah terdapat CPU, ROM, RAM, I/O, *Clock* dan peralatan internal lainnya yang sudah saling terhubung dan *terorganisasi* dengan baik oleh pabrik pembuatnya dan dikemas dalam satu *chip* yang siap dipakai. Karena kemampuannya yang tinggi, bentuknya yang kecil, konsumsi dayanya rendah dan harga yang terjangkau maka mikrokontroler begitu banyak digunakan.

Untuk membuat alat penghitung kerupuk yang sesuai dengan kegunaan yang diinginkan, mikrokontroler harus diprogram terlebih dahulu dengan baik menggunakan perangkat lunak mikrokontroler. Adapun ide pemilihan judul oleh penulis adalah alat penghitung dan pemilah kerupuk putih pada industri kerupuk putih berbasis mikrokontroler. Pemilihan judul dan permasalahan ini adalah berdasarkan pada kenyataan dilapangan sekarang ini, yaitu masih banyak pabrik kerupuk yang proses menghitung jumlah produksi secara *manual* oleh tangan manusia, sehingga muncul ide penulis untuk menjawab masalah.

Untuk konsep dasar pembuatan alat ini sendiri, penulis memperolehnya dari beberapa sumber dalam bentuk jurnal yang membahas seputar alat penyeleksi dan penghitung uang logam berbasis mikrokontroler. Contoh model pertama yang dilakukan oleh Mr. Richard D. Primdahl dan Mr. Donald E. Raterman (1990), mereka mendisain alat yang diberi nama *Coin Sorter With Counter And Brake Mechanism*. Alat tersebut memiliki tujuan untuk menghitung dan memisahkan uang logam secara otomatis. Lalu contoh model kedua yang dilakukan oleh

Karyanto (2011), dalam tugas akhirnya beliau merancang alat dengan nama Prototipe Mesin Penyeleksi dan Penghitung Uang Logam Berbasis Mikrokontroler ATMEL AVR AT90S8515. Alat tersebut memiliki tujuan yang sama dengan contoh pertama namun terdapat perbedaan pada mikrokontrolernya. Dan contoh tugas akhir yang ketiga dibuat oleh Nur Widiyanto (2014), beliau membuat Rancang Bangun Otomatis Alat Penghitung Benih Ikan Menggunakan Arduino. Alat tersebut bertujuan untuk menghitung jumlah benih ikan yang diinginkan menggunakan mikrokontroler arduino, sensor optocoupler, *keypad*, aktuator motor DC, dan *output* tampilan LCD.

Dari ketiga konsep tersebut, penulis berusaha untuk memodifikasi dengan mengganti objek yang akan dihitung yaitu kerupuk putih putih, lalu merubah bentuk maket yang disesuaikan dengan ukuran kerupuk, serta sistem kendalinya menggunakan perangkat Arduino.

## **1.2 Identifikasi Masalah**

Berdasarkan latar belakang masalah di atas, maka dapat diidentifikasi suatu masalah yang akan diteliti yaitu sebagai berikut:

1. Bagaimana merancang alat penghitung dan pemilah kerupuk putih?
2. Sensor apa yang digunakan pada alat penghitung kerupuk putih?
3. Bagaimana memanfaatkan Arduino sebagai pengendali alat penghitung kerupuk putih?
4. Apakah alat penghitung kerupuk dapat memisahkan kerupuk putih yang bentuknya tidak sempurna atau cacat produksi?

### 1.3 Pembatasan Masalah

Karena terbatasnya sarana dan prasarana dalam pembuatan alat, maka masalah yang akan dibahas akan dibatasi pada:

1. Kerupuk putih yang akan dihitung adalah yang adonannya sudah dikeringkan dan belum digoreng.
2. Sistem program penghitung kerupuk putih menggunakan perangkat lunak Arduino IDE dan *Visual basic*.
3. Alat penghitung kerupuk putih menggunakan perangkat keras sensor *photo infrared*, Arduino UNO, motor DC, dan LCD.
4. Alat penghitung kerupuk putih dapat memilah kerupuk yang bentuknya tidak sempurna.

### 1.4 Rumusan Masalah

Berdasarkan latar belakang masalah diatas, maka dirumuskan masalah sebagai berikut, “Bagaimana merancang alat penghitung dan pemilah kerupuk putih pada *home industry* berbasis mikrokontroler?”

### 1.5 Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk membuat alat penghitung dan pemilah kerupuk putih pada *home industry* berbasis mikrokontroler sebagai penghitung kerupuk putih yang sudah dikeringkan, agar dapat diimplementasikan dan meringankan pekerjaan menghitung kerupuk putih pada suatu *home industry* yang memproduksi kerupuk putih.

### 1.6 Manfaat Penelitian

Adapun manfaat yang akan diperoleh dari penelitian ini, antara lain:

1. Memudahkan *home industry* kerupuk putih dalam menghitung jumlah kerupuk yang akan dijual.
2. Dapat memisahkan bentuk kerupuk putih yang baik dengan yang tidak sempurna atau cacat produksi.
3. Dapat menambah pengetahuan tentang salah satu aplikasi sistem kontrol menggunakan Arduino dalam kehidupan sehari-hari.



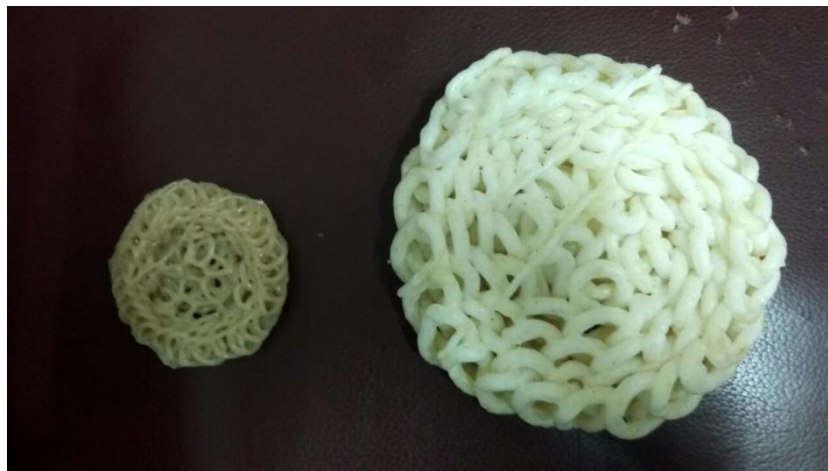
## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Kerangka Teori**

##### **2.1.1 Kerupuk Putih**

Menurut Harianto (2007: 16) kerupuk adalah salah satu makanan ringan yang cocok disandingkan dengan berbagai jenis makanan. Kerupuk terdiri atas berbagai jenis. Antara lain kerupuk ikan, kerupuk udang, kerupuk pati singkong atau sering disebut dengan kerupuk putih. Kerupuk putih merupakan jenis kerupuk yang banyak digemari dan dapat dijumpai mulai dari warung, rumah makan pinggir jalan, hingga restoran.



**Gambar 2.1 Kerupuk Putih Yang Telah Dikeringkan Dan Sudah Digoreng**

Kerupuk putih dapat dibuat dari campuran tepung tapioka, tepung terigu, ikan tenggiri, bawang putih, garam, dan penyedap rasa.

Kerupuk putih mula-mula dibuat dengan cara memasak tepung tapioka dan tepung terigu hingga menjadi adonan seperti bubur. Setelah itu adonan diangkat,

kemudian dimasukkan ke dalam mesin pengaduk bersama dengan garam, bawang putih, dan penyedap rasa. Aduk hingga rata.

Selanjutnya, adonan dimasukkan ke dalam mesin pencetak. Setelah dicetak kerupuk direbus kembali. Setelah itu kerupuk dijemur selama satu hingga dua hari, hingga benar-benar kering. Pada saat dijemur sebaiknya kerupuk sering dibolak-balik (diaduk-aduk) agar keringnya merata. Setelah itu kerupuk dapat digoreng.

### **2.1.2 Arduino**

Menurut Andrianto (2016: 15) Arduino adalah suatu perangkat prototipe elektronik berbasis mikrokontroler yang fleksibel dan *open-source*, perangkat keras dan perangkat lunaknya mudah digunakan. Perangkat ini ditujukan bagi siapapun yang tertarik/ memanfaatkan mikrokontroler secara praktis dan mudah. Bagi pemula dengan menggunakan board ini akan mudah mempelajari pengendalian dengan mikrokontroler, bagi disainer pengontrol menjadi lebih mudah dalam membuat prototipe ataupun implementasi; demikian juga bagi para hobi yang mengembangkan mikrokontroler. Arduino dapat digunakan ‘mendeteksi’ lingkungan dengan menerima masukan dari berbagai sensor (misal: cahaya, suhu, inframerah, ultrasonik, jarak, tekanan, kelembaban) dan dapat ‘mengendalikan’ peralatan sekitarnya (misal: lampu, berbagai jenis motor, dan aktuator lainnya).

Arduino merupakan rangkaian yang *open-source* dan bebas digunakan asalkan memenuhi persyaratan yang telah ditentukan sesuai yang tercantum pada <http://creativecommons.org/licenses/by-sa/2.5/>, sedangkan perangkat lunak dapat

digunakan pada berbagai *platform* sistem operasi (Windows, Mac OS, linux) dan dapat diunduh dari <http://arduino.cc/en/Main/Software>.

Informasi tentang sejarah perkembangan arduino, macam-macam *board* Arduino serta contoh aplikasi yang sangat luas dan lengkap dapat diunjungi di <http://www.arduino.cc>.

### 2.1.2.1 *Board* Arduino UNO

Menurut Darmawan (2016: 24) *board* Arduino UNO menggunakan mikrokontroler ATmega 328. Secara umum posisi/letak *pin-pin* terminal I/O pada berbagai *board* Arduino posisinya sama dengan posisi/letak *pin-pin* terminal I/O dari Arduino UNO yang mempunyai 14 *pin* digital yang dapat di set sebagai *input/output* (beberapa diantaranya mempunyai fungsi ganda), 6 *pin input* analog.

Pada bagian ini akan dijelaskan fungsi dari *pin* dan terminal pada *board* Arduino Uno. *Board* Arduino Uno dapat dilihat pada Gambar 2.2.



**Gambar 2.2** Arduino UNO

#### a. *USB to Computer*

Digunakan untuk koneksi ke komputer atau alat lain menggunakan komunikasi serial RD-232 *standard*. Bekerja ketika JP0 dalam posisi 2-3.

b. DC1, 2.1 mm *powerjack*

Digunakan sebagai sumber tegangan (catu daya) dari luar, sudah terdapat regulator tegangan yang dapat meregulasi masukan tegangan antara +7V sampai +18V (masukan tegangan yang disarankan antara +9V s/d +12V). *pin* 9V dan 5V dapat digunakan sebagai sumber ketika diberi sumber tegangan dari luar.

c. ICSP, 2x3 *pinheader*

Untuk memprogram *bootloader* ATmega atau memprogram Arduino dengan *software* lain, berikut ini keterangan fungsi tiap *pin*:

**Tabel 2.1 Keterangan *Pin* ICSP Pada Arduino UNO**

1	MISO	+5V	2
3	SCK	MOSI	4
5	RST	GND	6

(Sumber: Arduino Belajar Cepat Dan Pemrograman)

d. JP0, 3 *pin jumper*

Ketika posisi 2-3, *board* pada keadaan *serial enabled* (X1 *connector* dapat digunakan). Ketika posisi 1-2 *board* pada keadaan *serial disabled* (X1 *connector* tidak berfungsi) dan eksternal *pull-down resistor* pada *pin*0 (RX) dan *pin*1 (TX) dalam keadaan aktif, resistor *pull-down* untuk mencegah *noise* dari RX.

e. JP4

Ketika pada posisi 1-2, *board* dapat mengaktifkan fungsi *auto-reset*, yang berfungsi ketika meng-*upload* program pada *board* tanpa perlu menekan *reset* S1.

## f. S1

Adalah *push button* yang berfungsi sebagai tombol *reset*.

## g. LED

*POWERLED* : menyala ketika Arduino dinyalakan dengan diberi tegangan dari DC1.

*RX LED* : berkedip ketika menerima data melalui komputer lewat komunikasi serial.

*TX LED* : berkedip ketika mengirim data melalui komputer lewat komunikasi serial.

*L LED* : terhubung dengan digital *pin13*. Berkedip ketika *bootloading*.

h. Digital *pinout in/out*

8 digital *pin inputs/outputs* : *pin 0-7* (terhubung pada *port D* dari ATMEGA). *Pin-0(RX)* dan *pin-1(TX)* dapat digunakan sebagai *pin* komunikasi. Untuk ATmega168/328 *pin 3, 5* dan *6* dapat digunakan sebagai *output PWM*. Enam (6) *pin inputs/outputs* digital: *pin8-13* (terhubung pada *port B*). *pin10(SS)*, *pin11 (MOSI)*, *pin12 (MISO)*, *pin13 (SCK)* yang bisa digunakan sebagai SPI (Serial peripheral interface). *Pin9, 10, dan 11* dapat digunakan sebagai *output PWM* untuk ATmega8 dan ATmega168/328.

i. Analog *pinout input*

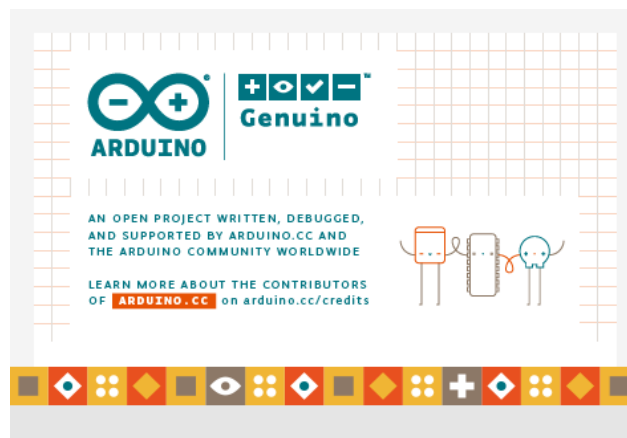
Enam (6) analog *input analog* : *pin0-5 (A0-A5)* (terhubung pada *port C*). *pin4 (SDA)* dan *pin5 (SCL)* yang dapat digunakan sebagai I2c (*two-wire serial bus*). *Pin analog* ini dapat digunakan sebagai *pin digital14 (A0)* sampai *pin digital pin19 (A5)*.

### 2.1.2.2 Aplikasi Program Arduino IDE

Arduino IDE (*Integrated Development Environment*) merupakan aplikasi yang mencakup *editor*, *compiler*, dan *uploader* dapat menggunakan semua seri modul keluarga Arduino seperti Arduino Duemilanove, Uno, Bluetooth, Mega (Istiyanto, 2015: 46).

*Editor* program adalah jendela yang memungkinkan pengguna menulis program menggunakan bahasa pemrograman. *Compiler* adalah sebuah modul yang mengubah kode program menjadi kode biner. Sedangkan *uploader* adalah modul yang memuat kode biner dari komputer ke dalam memori di dalam papan Arduino.

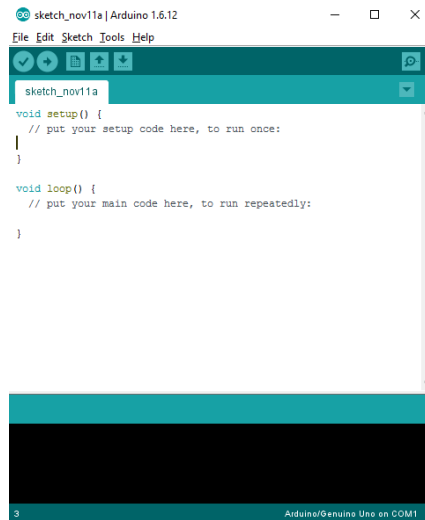
Bahasa pemrograman pada Arduino IDE menggunakan bahasa C. Serangkaian kode program yang dibuat pada Arduino IDE disebut *sketch*, di mana *sketch* ini akan disimpan sebagai *file* berformat *hex* dan *file* ini yang di-*upload* ke papan Arduino.



**Gambar 2.3 Tampilan Awal Arduino IDE 1.6.12**

Perangkat lunak ini dapat dijalankan pada komputer yang menggunakan sistem operasi *Windows*, *Mac OS X*, dan *Linux*. Sampai saat ini Arduino IDE telah banyak melakukan pembaharuan versi demi perbaikan. Arduino IDE yang

digunakan pada penelitian ini adalah Arduino IDE versi 1.6.12. *Form splash* dari arduino 1.6.12 diperlihatkan pada Gambar 2.3 dan contoh jendela penulisan *sketch*-nya pada Gambar 2.4.



**Gambar 2.4 Software IDE Arduino**

### 2.1.3 Bahasa Pemrograman *Visual basic*.NET 2010

Menurut Hirin (2011: 2) *Visual Basic* adalah salah satu bahasa pemrograman berbasis desktop yang dikeluarkan (diproduksi) oleh perusahaan perangkat lunak komputer terbesar yaitu Microsoft.

*Visual Basic* merupakan salah satu bahasa pemrograman paling laris dan paling sukses di dunia. Dimana tercatat sampai pada tahun 2015 *Visual Basic* merupakan bahasa pemrograman yang paling banyak dipakai oleh para programmer bahkan diyakini sampai saat ini. Menjadi pilihan berbagai kalangan tentunya *Visual Basic* memiliki berbagai hal yang patut dijadikan alasan, selain bahasa pemrograman yang sangat (paling) mudah dipelajari oleh berbagai kalangan baik awam maupun ahli, *Visual Basic* yang didukung penuh oleh produsennya (Microsoft) selalu dikembangkan dan disesuaikan dengan kebutuhan

zaman seperti penyesuaian model pemrograman modern yang berbasis OOP (*Object Oriented Programming*).

*Visual Basic* 2010 yang sudah mengusung pemrograman berorientasi objek. Dimana sebelum munculnya versi ini Microsoft sudah merilis berbagai versi pendahulunya, berikut kronologi versi *Visual basic* yang pernah dirilis oleh Microsoft ke publik sesuai tahun rilisnya.

**Tabel 2.2 Versi *Visual basic***

<b>Versi</b>	<b>Tahun Rilis</b>	<b>.Net Framework</b>
<i>Visual basic</i> 1.0	Mei, 1991	-
<i>Visual basic</i> 2.0	November, 1992	-
<i>Visual basic</i> 3.0	Juni, 1993	-
<i>Visual basic</i> 4.0	Agustus, 1995	-
<i>Visual basic</i> 5.0	Februari, 1996	-
<i>Visual basic</i> 6.0	Oktober , 1998	-
<i>Visual basic</i> 7.0 (vb.net 2002)	Februari, 2002	<i>Framework</i> 1.0
<i>Visual basic</i> 7.0 (vb.net 2003)	Februari, 2003	<i>Framework</i> 1.5
<i>Visual basic</i> 8.0 (vb.net 2005)	Februari, 2005	<i>Framework</i> 2.0
<i>Visual basic</i> 9.0 (vb.net 2008)	Februari, 2007	<i>Framework</i> 3.5
<i>Visual basic</i> 10 (vb.net 2010)	Februari, 2010	<i>Framework</i> 4.0

(Sumber: Belajar Tuntas (Dari Dasar Sampai Mahir) VB.NET 2010.)

Setelah *Visual Basic* versi 6.0, Microsoft melakukan perubahan besar pada bahasa pemrograman *Visual Basic* versi selanjutnya. Di mana ditambahkan suatu pustaka-pustaka yang terangkai menjadi suatu kesatuan. Yang disebut dengan .net (dotnet) *framework*. Selain itu ditambahkan (diperkuat) pula pemodelan pemrograman berorientasi objek.

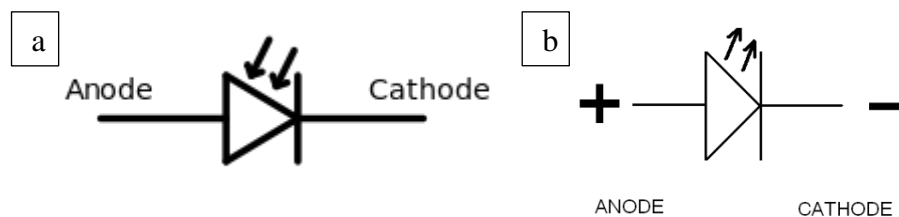
Keuntungan adanya *framework* ini pembuatan program *Visual Basic* terkesan lebih mudah dan singkat karena dalam *framework* telah terbungkus berbagai komponen dan *class* yang siap pakai sehingga kita tidak perlu menulis kode yang terlalu panjang untuk melakukan berbagai fungsi tertentu.



Kekurangannya tentu saja program yang dibuat akan memakan lebih banyak tempat (memori) baik memori *Hardisk* maupun RAM karena *framework* juga harus didistribusikan dengan terpasang pada komputer.

#### 2.1.4 Sensor *Photo Infrared*

Menurut Andrianto (2016: 95) *infrared* (dioda *infrared*) adalah suatu komponen yang memancarkan sinar pada daerah panjang gelombang infra merah dan biasanya digunakan sebagai pemancar gelombang infra merah (TX). *Photodiode* adalah suatu komponen yang peka terhadap cahaya (gelombang infra merah, cahaya tampak, sinar x), prinsip kerja fotodiode jika sebuah sambungan-pn dibias mundur dan diberikan cahaya arus akan bertambah cukup besar.



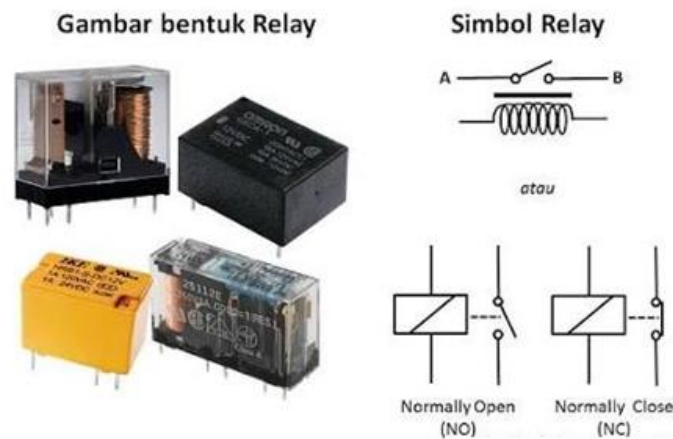
**Gambar 2.5 Simbol (a) *Photodiode*, (b) *Infrared LED*.**

Simbol skematik *photodiode* dan *infrared LED* hanya memiliki sedikit perbedaan, yaitu terletak pada tanda panah yang menghadap ke dalam seperti diperlihatkan pada Gambar 2.5.

#### 2.1.5 *Relay*

Menurut Jatmiko (2015: 15) *relay* adalah saklar (*switch*) yang dioperasikan secara listrik dan merupakan komponen elektromekanikal yang terdiri dari 2 bagian utama yakni elektromagnet (*coil*) dan mekanikal (seperangkat kontak saklar / *switch*).

*Relay* menggunakan prinsip elektromagnetik untuk menggerakkan kontak saklar sehingga dengan arus listrik yang kecil dapat menghantarkan listrik yang bertegangan lebih tinggi. Sebagai contoh, dengan *relay* yang menggunakan 5V dan 50 mA mampu menggerakkan armature *relay* (yang berfungsi sebagai saklarnya) untuk menghantarkan listrik 220V 2A.



**Gambar 2.6 Relay**

Kontak poin *relay* terdiri dari 2 jenis yaitu :

1. *Normally Close* (NC) yaitu kondisi awal sebelum diaktifkan akan selalu berada di posisi tertutup.
2. *Normally Open* (NO) yaitu kondisi awal sebelum diaktifkan akan selalu berada di posisi terbuka.

### 2.1.6 Motor DC Gearbox

Motor listrik merupakan suatu peralatan yang mampu mengubah energi listrik menjadi energi mekanik berupa putaran pada poros (Bintoro, 2000: 23). Motor DC adalah motor listrik yang memerlukan suplai tegangan arus searah pada kumparan medan untuk diubah menjadi energi gerak mekanik. Kumparan medan pada motor DC disebut stator (bagian yang tidak berputar) dan kumparan jangkar

disebut rotor (bagian yang berputar). *Gearbox* adalah perangkat yang mengkonversi kecepatan dan torsi. Jadi motor DC *gearbox* dapat diartikan sebagai motor listrik arus searah yang menggunakan perangkat *gearbox*, sehingga torsi dari putaran motor menjadi besar. Contoh motor *gearbox* dapat dilihat pada Gambar 2.7.

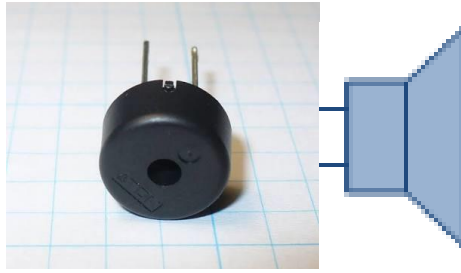


**Gambar 2.7 Motor DC *Gearbox***

### **2.1.7 *Buzzer***

Menurut Anggraini (2010: 5) *buzzer* adalah suatu komponen yang dapat merubah energilistrik menjadi energi suara yang terdiri dari dua lempengan logam, yaitu lempengan logam tipis dan lempengan logam tebal. *Buzzer* biasanya digunakan sebagai indikator dalam suatu sistem sebagai penanda atau peringatan.

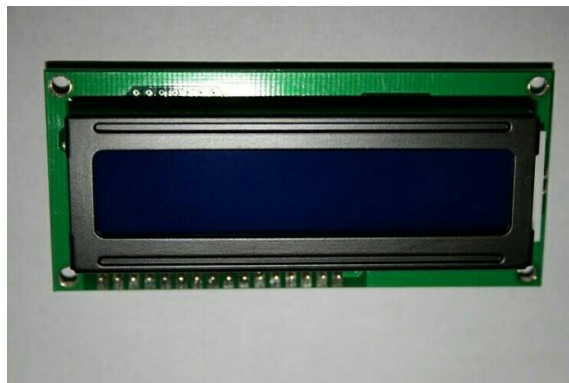
Bila *buzzer* mendapatkan tegangan maka lempengan logam 1 dan 2 bermuatan bermuatan listrik. Dengan adanya muatan listrik maka terdapat beda potensial di kedua lempengan, beda potensial akan menyebabkan lempengan 1 bergerak saling bersentuhan dengan lempengan 2. Diantara lempengan 1 dan 2 terdapat rongga udara, sehingga apabila terjadi proses getaran di rongga udara maka *buzzer* akan menghasilkan bunyi dengan frekuensi tinggi.



**Gambar 2.8 Bentuk Dan Simbol *Buzzer***

### 2.1.8 LCD 2x16

Menurut Kadir (2013: 196) *Liquid Crystal Display* (LCD) adalah komponen yang dapat menampilkan tulisan. Salah satu jenisnya memiliki dua baris dengan setiap baris terdiri atas 16 karakter. LCD seperti itu biasa disebut LCD 2x16 yang diperlihatkan pada Gambar 2.9.



**Gambar 2.9 LCD 2x16**

LCD 2x16 biasa digunakan pada peralatan-peralatan berbasis mikrokontroler yang dapat menampilkan tulisan berupa angka atau huruf sebagai suatu informasi. LCD 2x16 memiliki 16 *pin* yang memiliki fungsi-fungsi tertentu. Fungsi dari *pin-pin* tersebut adalah sebagai berikut :

- a. *Pin* 1 dan 2 merupakan *pin* yang tersambung dengan catu daya. Vdd terhubung dengan tegangan positif catu daya, sedangkan vss tersambung pada 0V atau *ground*.

- b. *Pin 3* merupakan *pin* yang digunakan untuk mengatur kontras *display*. Idealnya *pin* ini dihubungkan dengan tegangan yang bisa diubah untuk memungkinkan pengaturan tingkatan kontras *display* sesuai kebutuhan.
- c. *Pin 4* merupakan *register select*, masukan yang pertama dari tiga *command control input*. Dengan membuat (*RS*) menjadi *HIGH*, data karakter dapat ditransfer dari dan menuju modulnya.
- d. *Pin 5* merupakan *Read/Write (R/W)*. Cara memfungsikan perintah *write* adalah *R/W LOW* untuk menulis karakter ke modul. *R/W HIGH* untuk membaca data karakter atau informasi status registernya.
- e. *Pin 6* merupakan *enable (E)*. *Input* ini digunakan untuk transfer aktual perintah-perintah atau karakter antara modul dengan hubungan data. Ketika menulis ke *display*, data ditransfer hanya pada perpindahan *HIGH* ke *LOW*. Namun ketika membaca dari *display*, data akan lebih cepat tersedia setelah perpindahan dari *LOW* ke *HIGH* dan tetap tersedia hingga sinyal *LOW* kembali
- f. *Pin 7* sampai *14* merupakan adalah 8 jalur data (*D0* sampai *D7*) dimana data dapat ditransfer ke dan dari *display*.
- g. *Pin 15* atau *A (+)* mempunyai *level DC +5V* dan berfungsi sebagai *LED backlight +*, sedangkan *pin 16* atau *k (-)* memiliki level *0V* dan berfungsi sebagai *LED backlight -*

Pada *Arduino IDE* terdapat *library* yang memudahkan untuk melakukan pengaksesan atau pengendalian *LCD 2x16* yaitu *library* “*LiquidCrystal.h*”.

## 2.2 Kerangka Berpikir

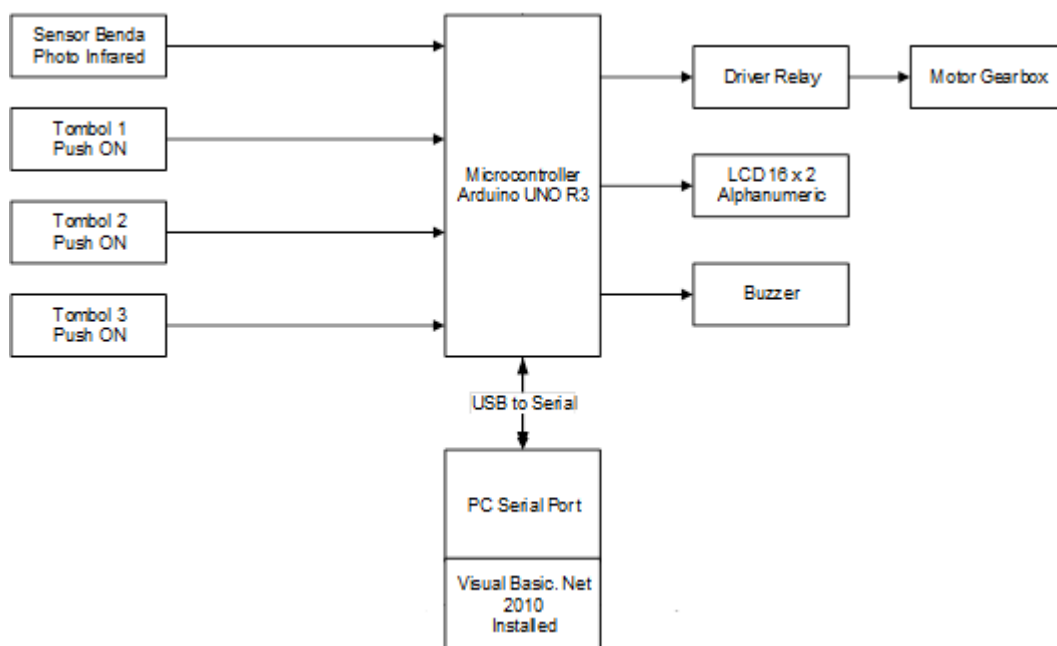
Metode penelitian yang digunakan dalam menyelesaikan penelitian ini adalah menggunakan metode rekayasa teknik meliputi perancangan, analisa kebutuhan, perancangan, pengujian, implementasi *hardware* dan *software*.

Adapun penelitian terkait dari jurnal, berdasarkan penelitian ini akan menjadi konsep dalam pembuatan alat penghitung kerupuk. Menurut Mr. Richard D. Primdahl dan Mr. Donald E. Raterman (1990). Karyano (2011), Nur Widiyanto (2014), dengan alat yang diberi nama *Coin Sorter With Counter And Brake Mechanism*. Alat tersebut memiliki tujuan untuk menghitung dan memisahkan uang logam secara otomatis. Lalu contoh model kedua yang dilakukan oleh Karyano (2011), dalam tugas akhirnya beliau merancang alat dengan nama Prototipe Mesin Penyeleksi dan Penghitung Uang Logam Berbasis Mikrokontroler ATMEL AVR AT90S8515. Alat tersebut memiliki tujuan yang sama dengan contoh pertama namun terdapat perbedaan pada mikrokontrolernya. Dan contoh tugas akhir yang ketiga dibuat oleh Widiyanto (2014), beliau membuat Rancang Bangun Otomatis Alat Penghitung Benih Ikan Menggunakan Arduino. Alat tersebut bertujuan untuk menghitung jumlah benih ikan yang diinginkan menggunakan mikrokontroler arduino, sensor optocoupler, aktuator motor DC, dan *output* tampilan LCD.

Berdasarkan teori-teori yang telah dibahas, maka dapat dirancang sebuah alat untuk menghitung kerupuk putih pada industri kerupuk putih, di mana dalam sistem tersebut menggunakan Arduino UNO sebagai perangkat kendali.

### 2.2.1 Blok Diagram

Blok diagram digunakan untuk menentukan komponen penyusunan dari suatu alat yang akan dibuat, sehingga hasil akhirnya sesuai dengan yang diinginkan.



**Gambar 2.10 Blok diagram sistem alat**

Alat penghitung kerupuk ini menggunakan sistem kendali Arduino UNO R3. . *input* dari alat ini yaitu sensor benda *photo infrared* yang digunakan untuk mendeteksi kerupuk putih, dan tombol untuk *switching*. Sedangkan *output* dari alat ini adalah motor DC sebagai aktuator, LCD 16x2 yang menampilkan secara visual hasil penghitungan kerupuk putih, dan *buzzer* untuk indikator suara (jika diperlukan). Adapun *software* yang digunakan alat ini yaitu Arduino IDE untuk memprogram Arduino dan *Visual basic.NET* 2010 berfungsi sebagai GUI (*Graphic User Interface*). Alat ini mendapatkan sumber tegangan dari *powersupply* 8V 3A untuk Arduino UNO, dan *powersupply* 5V 5A untuk *driver*

*relay* dan motor DC. Hal ini dikarenakan motor DC memerlukan tegangan dan arus yang stabil.

Dengan menggabungkan sistem kerja dari beberapa komponen *hardware* dan *software* yang telah disebutkan di atas, maka proses penghitungan kerupuk putih akan lebih efektif dan efisien, karena prosesnya dapat dikontrol secara otomatis.

### **2.2.2 Alur Kerja Sistem**

Sistem pengendalian alat penghitung kerupuk ini bekerja pada saat saklar utama dinyalakan, kabel USB arduino terkoneksi ke PC. *Control* melalui *Visual basic* akan disimpan oleh *database*, Pada *main menu* LCD tekan *switch* kuning untuk pilih *set*, maka pengguna dapat mengubah jumlah target kerupuk pada kolom satuan, puluhan, ratusan, dan ribuan sesuai pada data VB. Setelah sesuai tekan *switch* merah untuk kembali ke menu utama dan data target tadi tersimpan. Tekan *switch* hijau untuk mengubah ke mode run, maka pengguna dapat melihat jumlah target yang sudah diubah, dan kolom kosong untuk *counting* kerupuk. Proses *counting* kerupuk terjadi ketika *switch* kuning di tekan, maka Motor DC *gearbox* akan berputar untuk menggerakkan kerupuk agar dapat melewati lubang *output*, sensor *photo infrared* mengidentifikasi kerupuk putih yang lewat atau menghalangi *photodiode*. Secara otomatis motor akan berhenti ketika jumlah kerupuk yang terhitung sudah memenuhi target. Data pada VB di simpan dengan format xcl, maka pengguna masih dapat meng-*edit* tabel sebelum akhirnya di-*print*.



## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Tempat dan Waktu Penelitian**

Penelitian dilaksanakan di gedung L, Jurusan Teknik Elektro, Kampus A, Universitas Negeri Jakarta. Penelitian ini dilaksanakan pada bulan Januari s/d Agustus 2017.

#### **3.2 Alat dan Bahan Penelitian**

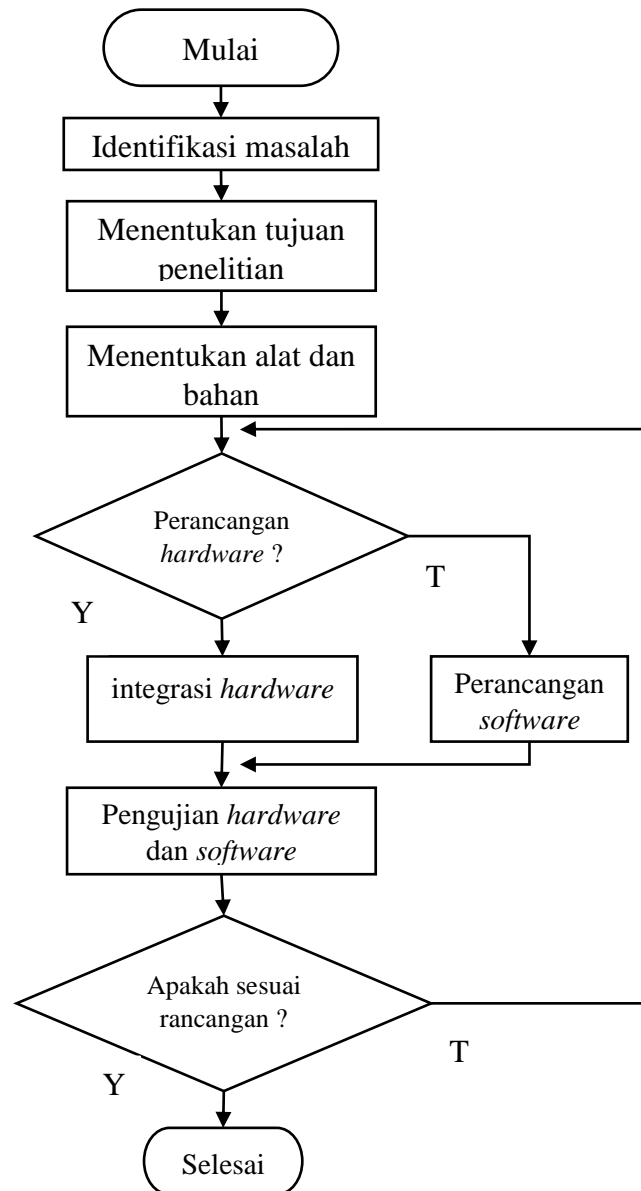
Alat dan bahan yang digunakan pada penelitian ini adalah alat dan bahan yang diperlukan dalam melakukan pengujian pada alat penghitung kerupuk putih. Adapun alat dan bahan tersebut yaitu :

1. Perangkat lunak :
  - a. Perangkat lunak Arduino IDE yang digunakan untuk memprogram board arduino
  - b. Perangkat lunak *Visual basic* yang digunakan untuk memprogram kontrol via VB
  - c. Perangkat lunak simulasi elektronik *Everycircuit* yang digunakan dalam mensimulasikan rangkaian elektronik yang akan digunakan
  - d. Perangkat lunak *Eagle* yang digunakan untuk membuat gambar skematik dan layout rangkaian pada PCB.
  - e. Perangkat lunak *Sketchup make* yang digunakan untuk membuat gambar disain alat
  - f. Microsoft *Word* 2016 yang digunakan untuk penulisan
  - g. Microsoft *Excel* yang digunakan untuk edit tabel

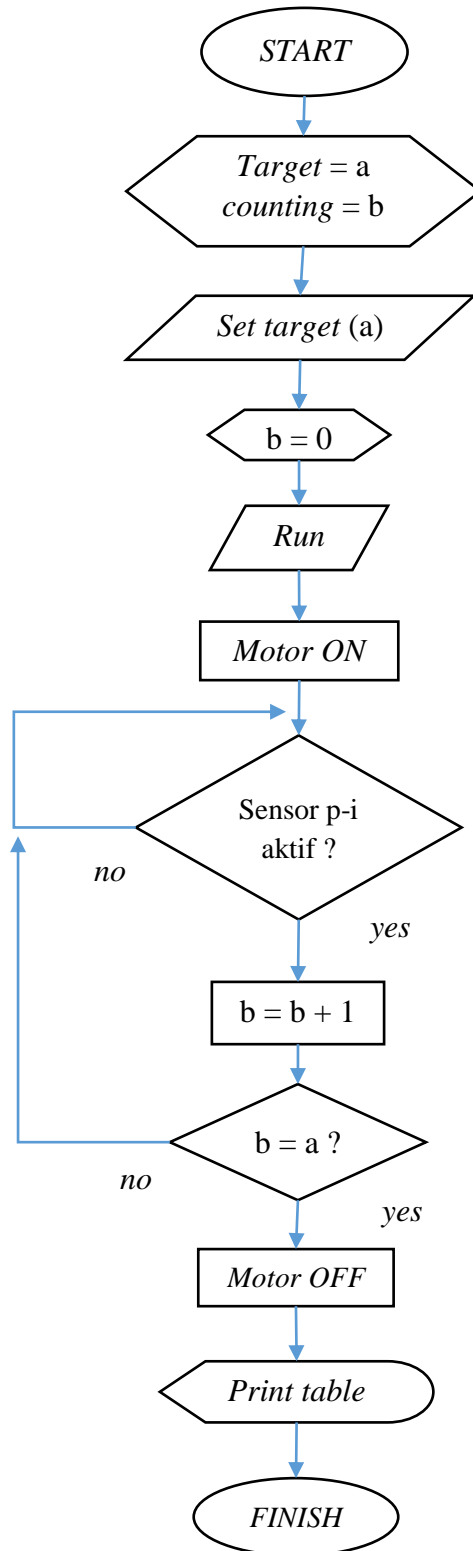
2. Alat Ukur :

- a. AVO Meter, digunakan untuk mengukur besar tegangan, arus, hambatan, dan mengecek komponen maupun jalur rangkaian

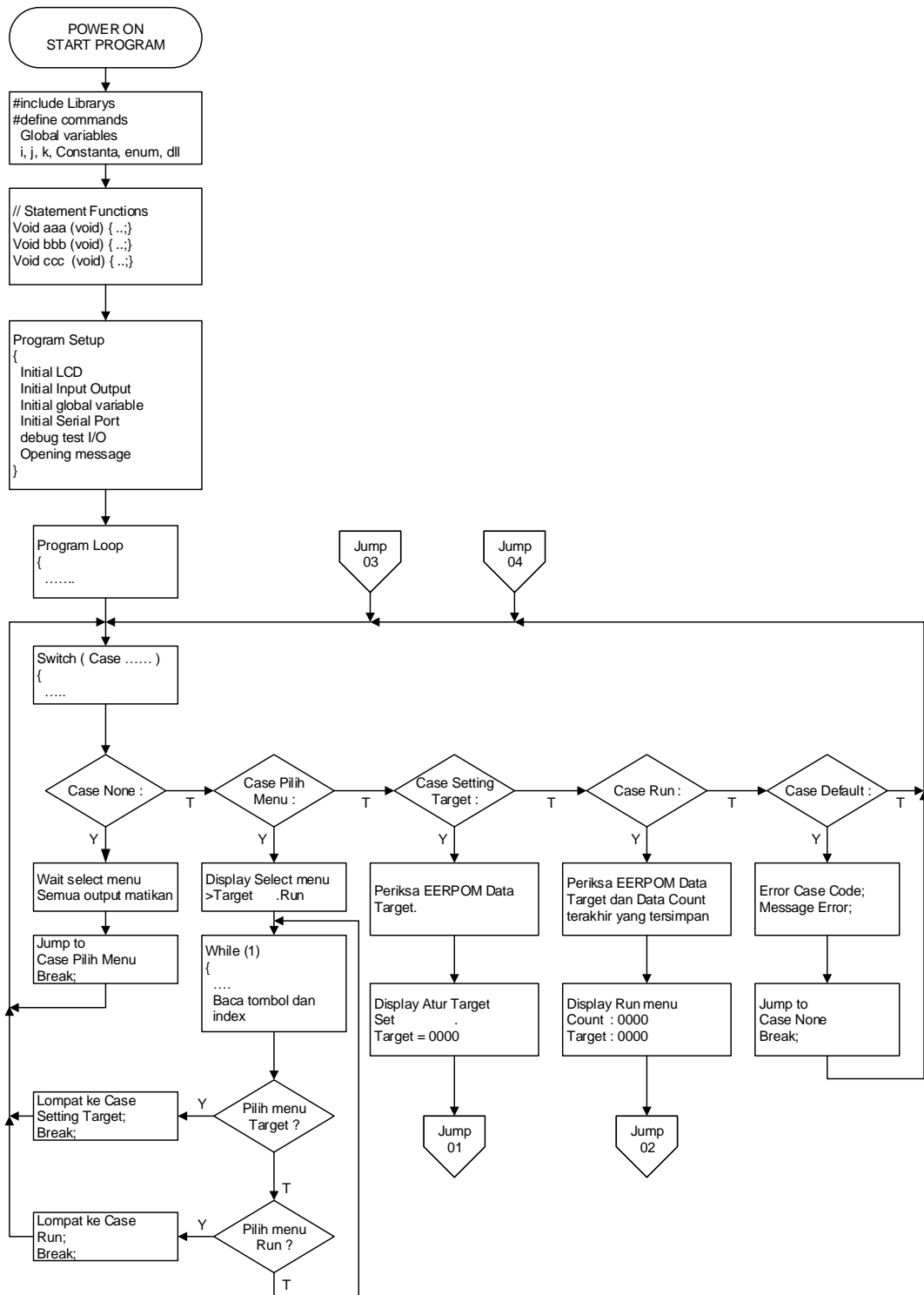
### 3.3 Diagram Alir Penelitian

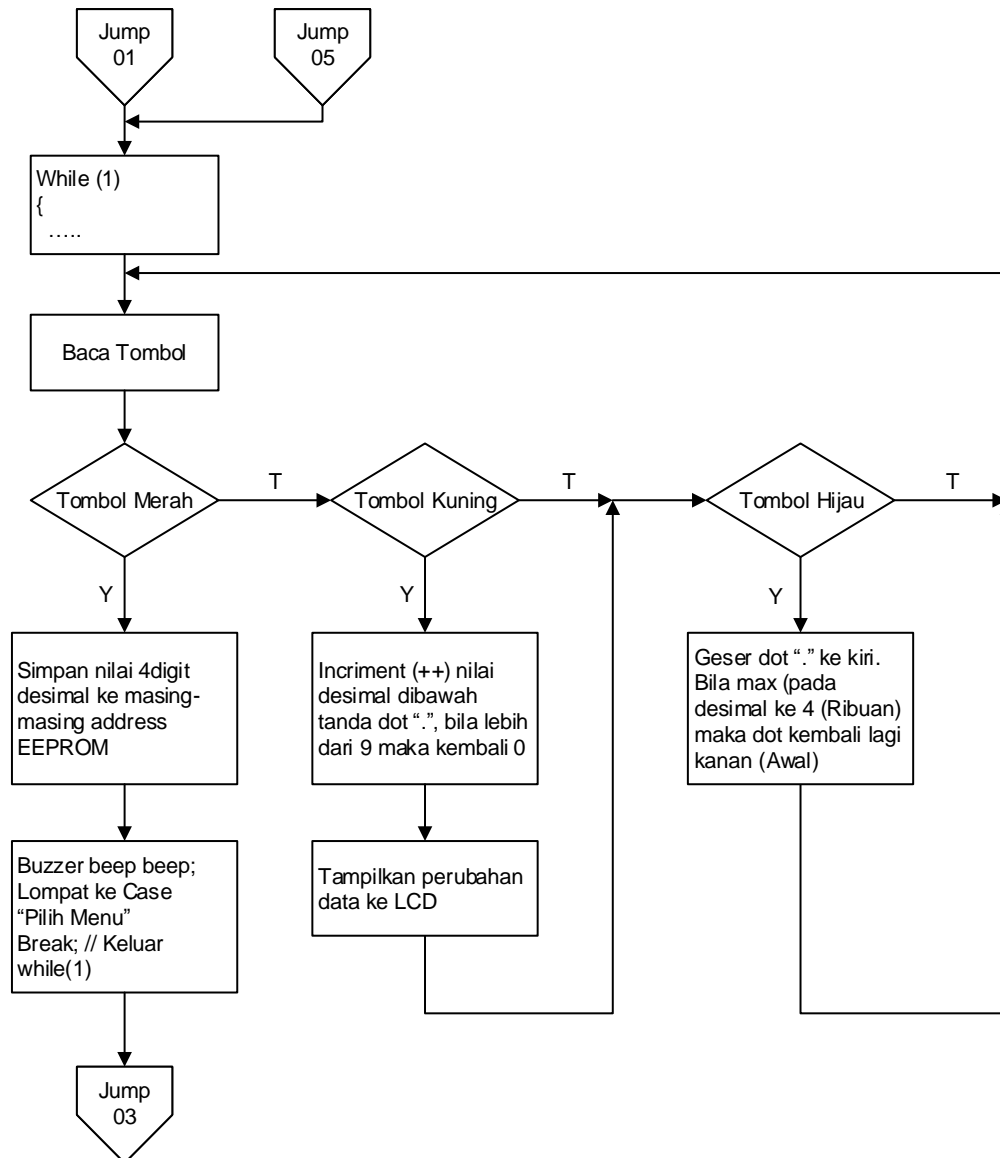


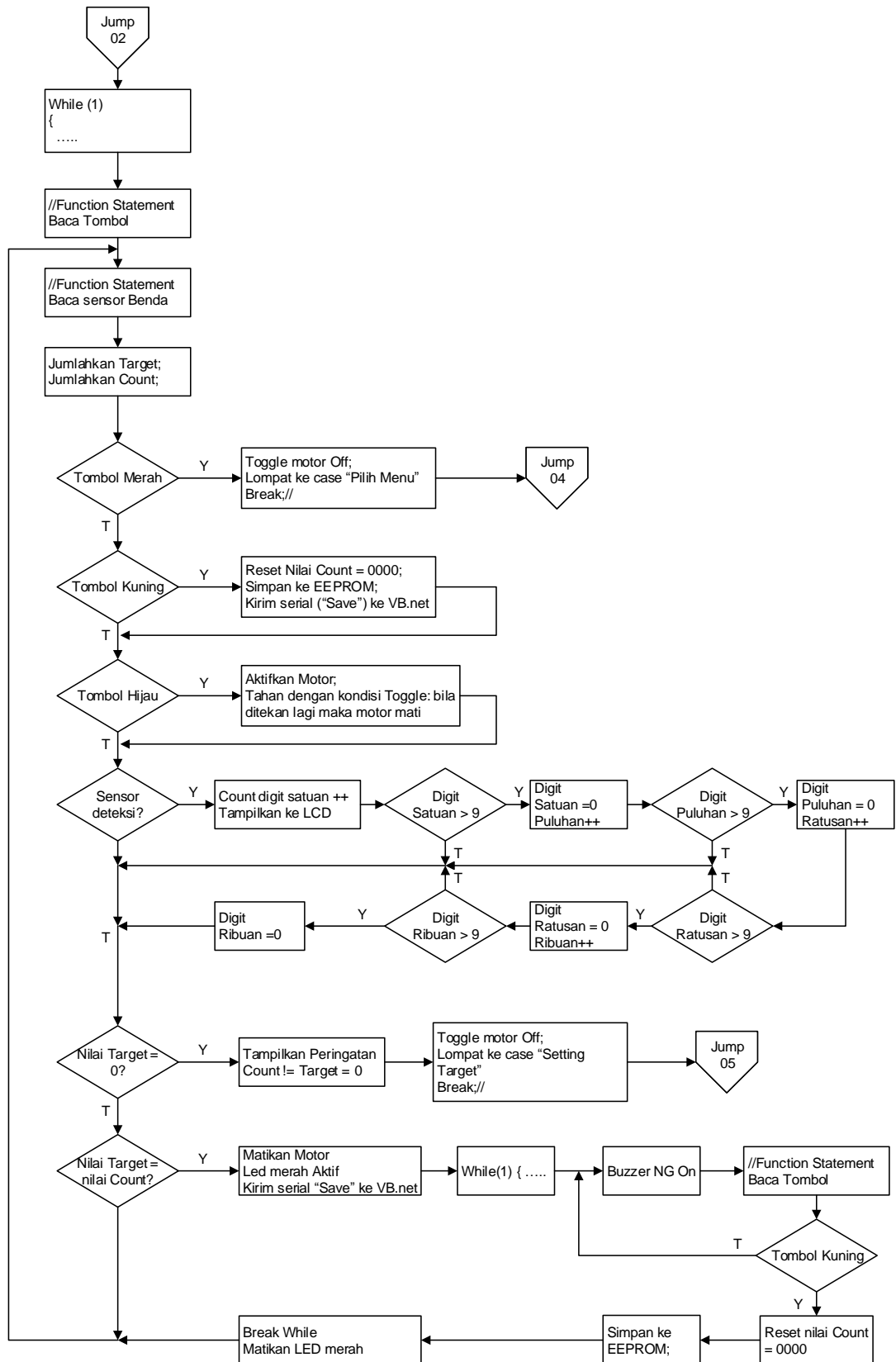
**Gambar 3.1** Flowchart Perancangan Alat



**Gambar 3.2** *Flowchart* Cara Kerja Alat







Gambar 3.3 Flowchart Program

### **3.3.1 Perancangan Penelitian**

Perancangan penelitian ini merupakan suatu rencana atau gagasan yang komprehensif dan mempunyai suatu tujuan yang terarah agar penelitian dapat berjalan dengan baik. Perancangan alat penghitung dan pemilah kerupuk putih pada *home industry* berbasis mikrokontroler adalah sebagai berikut:

#### **3.3.1.1 Menentukan Sistem Kendali**

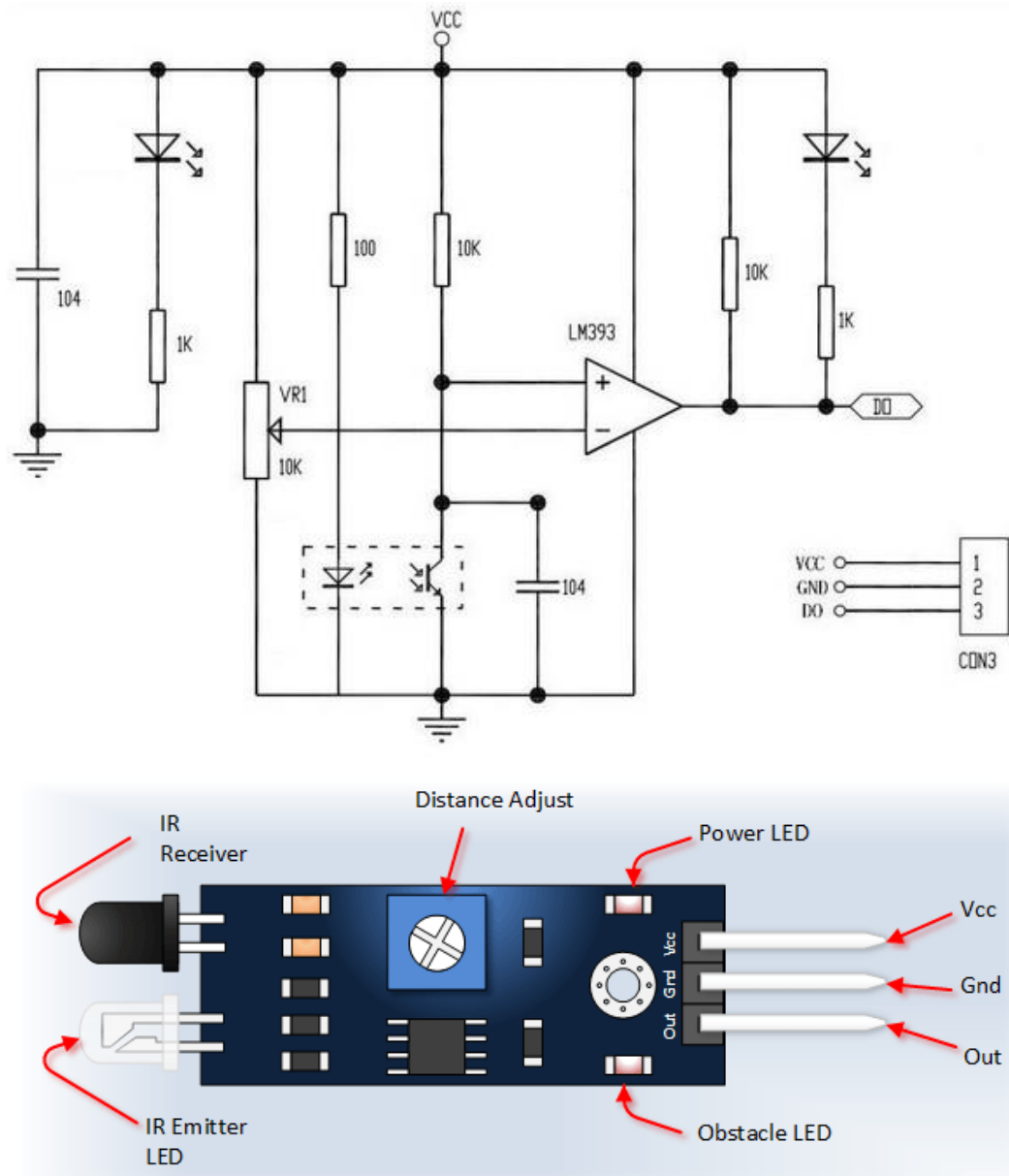
Sistem kendali yang digunakan untuk pembuatan alat ini adalah Arduino UNO yang dapat dilihat pada Gambar 2.2.

#### **3.3.1.2 Menentukan Motor DC**

Alat ini menggunakan motor DC *gearbox* dengan tegangan 12V agar memiliki torsi yang cukup untuk menggerakkan maket. Gambar motor DC *gearbox* dapat dilihat pada Gambar 2.7.

#### **3.3.1.3 Menentukan Sensor *Photo Infrared***

Alat ini menggunakan sensor *photo infrared* untuk mendeteksi kerupuk putih yang menghalangi sensor. Sensor *photo infrared* itu sendiri berupa *photodiode* dan LED *infrared*.



**Gambar 3.4** Sensor *Photodiode Infrared*

#### 3.3.1.4 Menentukan *Powersupply*

*Powersupply* berfungsi untuk membagikan tegangan ke semua komponen agar dapat bekerja. Untuk memudahkan dalam pengkabelannya, peneliti menggunakan *powersupply switching* yang dapat dilihat pada Gambar 3.5.

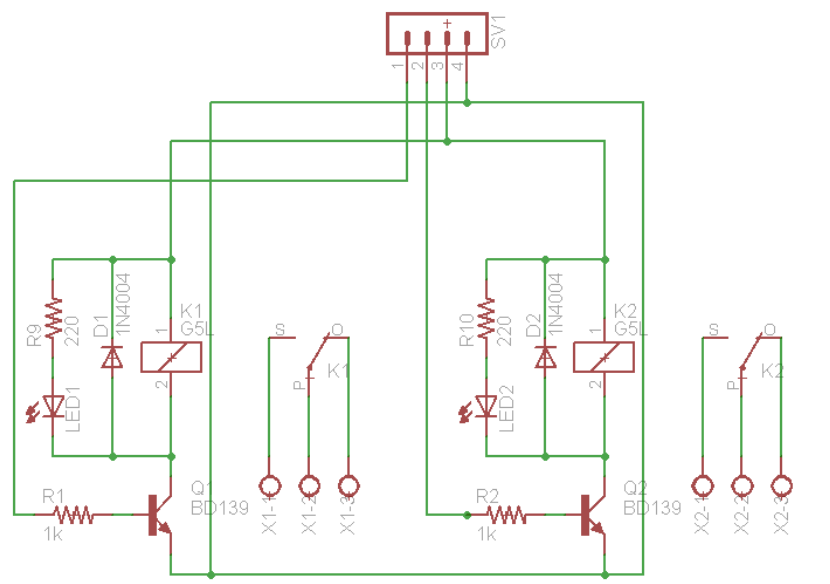




**Gambar 3.5 Powersupply Switching**

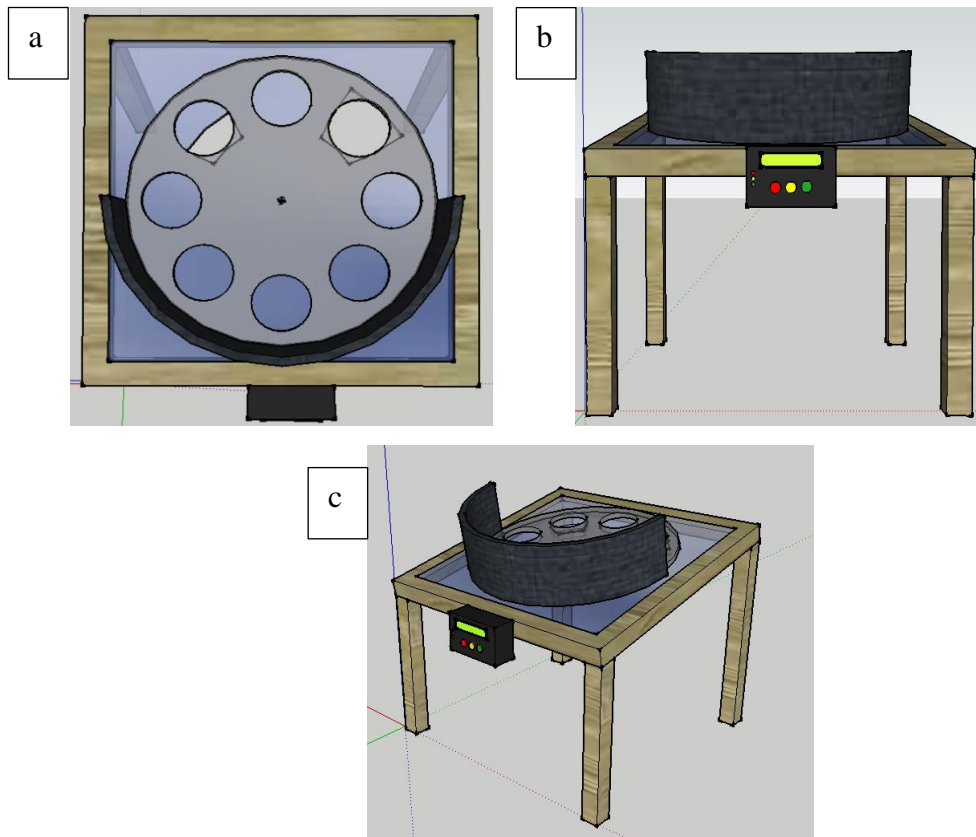
### 3.3.1.5 Menentukan *Driver Motor Relay*

Rangkaian *relay* pada alat ini berfungsi untuk memungkinkan penggunaan tegangan dan arus yang kecil didapat dari Arduino untuk mengontrol tegangan dan arus ke motor DC. Rangkaian *relay* ini terdiri dari 2 *channel*, yang masing-masing *channel*-nya dapat digunakan sebagai saklar *on/off*. *Channel* satu digunakan untuk mensaklarkan motor DC *Gearbox* yang *input*-nya dikendalikan oleh Arduino UNO. Rangkaian *relay* dapat dilihat pada Gambar 3.6 di bawah ini:



**Gambar 3.6 Driver Relay 2 Channel**

### 3.3.1.6 Perancangan Maket



**Gambar 3.7 Disain Alat Penghitung Kerupuk (a) Tampak Atas, (b) Tampak Depan, (c) Tampak Samping.**

Dalam pembuatan dan pengujian alat, disain alat ini berbentuk sebuah meja yang dibuat dengan bahan utama akrilik dan kayu. Peneliti menggunakan kayu pada bahan rangka karena kayu bisa didapatkan dengan mudah dan tidak sulit dalam proses pembentukan. Di bagian atas *body* terdapat penyangga yang digunakan untuk menahan kerupuk agar tidak terjatuh ke depan. Kerupuk yang berada pada silo atas nantinya akan disortir menggunakan lubang yang ada di bagian atas akrilik. Di bagian bawah akrilik terdapat sebuah sensor *infrared* yang digunakan untuk menghitung jumlah kerupuk yang lewat melalui lubang *rejector*. Dan motor DC *gearbox* yang digunakan untuk menggerakkan akrilik. Untuk rangkaian kendali diletakkan dibagian depan maket, yaitu berbentuk *balckbox*

yang dilengkapi dengan 3 tombol *switch* untuk mengendalikan alat. Untuk lebih jelasnya dapat dilihat pada Gambar 3.7.

### 3.3.1.7 Perancangan Arduino IDE

Arduino IDE merupakan salah satu bawaan *software* sendiri dari perangkat arduino di mana pemrogramannya menggunakan bahasa C/C++. Arduino IDE yang beroperasi di komputer berfungsi untuk menghasilkan sebuah *file* yang berformat .ino yang akan unggah pada papan arduino. Dalam hal ini Arduino IDE digunakan untuk membuat program alat sistem *counting* kerupuk pada bagian Arduino UNO yang nantinya digunakan sebagai pengendali perangkat keras.

Berdasarkan fungsi dari aplikasi arduino IDE di atas, peneliti menggunakannya sebagai pemrograman mikrokontroler Arduino UNO untuk alat. Beberapa *input* dan *output* yang dihubungkan menggunakan parameter data untuk memasukan program. Berikut ini adalah beberapa *input* dan *output* yang dihubungkan menggunakan parameter data untuk memasukkan program yang digunakan pada Arduino UNO menggunakan perangkat lunak Arduino IDE.

1. *Input* berupa tombol hijau, kuning, merah, dan sensor *photo infrared*. *pin* yang digunakan pada arduino Uno dapat dilihat pada Tabel 3.1 berikut:

**Tabel 3.1 *Input* Arduino UNO**

<b>JENIS</b>	<b><i>PIN</i> PERANGKAT <i>INPUT</i></b>	<b><i>PIN</i> ARDUINO</b>
Tombol hijau		A0
Tombol kuning		A1
Tombol merah		A2
Sensor <i>photo infrared</i>	VCC	5V
	<i>GROUND</i>	GND
	DATA	A3

2. *Output* berupa *relay*, *buzzer*, LED hijau, kuning, merah, dan LCD. *Pin* yang digunakan pada Arduino UNO dapat dilihat pada Tabel 3.2 berikut:

**Tabel 3.2 *Output* Arduino UNO**

<b>JENIS</b>	<b><i>PIN</i> PERANGKAT <i>OUTPUT</i></b>	<b><i>PIN</i> ARDUINO</b>
<i>Relay</i>	<i>Output 1</i>	3
<i>Buzzer</i>	<i>Output 2</i>	4
LED Hijau	<i>Output 3</i>	5
LED Kuning	<i>Output 4</i>	6
LED Merah	<i>Output 5</i>	7
LCD	<i>Output 6</i>	8,9,10,11,12,13

### 3.4 Teknik dan Prosedur Pengumpulan Data

Prosedur penelitian dalam merancang dan pengujian alat penghitung kerupuk berbasis arduino terdiri dari beberapa tahapan yaitu :

1. Perakitan modul dimulai dari pemilihan komponen, pemasangan, penyolderan, dan pengkabelan.
2. Mengecek (uji subsistem) secara *manual* apakah *relay* dapat digunakan atau tidak.
3. Mengecek secara *manual driver* motor dapat digunakan atau tidak.
4. Mengecek secara *manual* apakah motor DC dapat digunakan atau tidak.
5. Mengecek sensor *photo infrared* dapat digunakan atau tidak untuk mendeteksi kerupuk putih.
6. Mengecek *powersupply* apakah dapat digunakan atau tidak.
7. Merancang maket dengan menggunakan akrilik dan kayu.
8. Mengecek disain maket apakah dapat digunakan atau tidak.
9. Setelah semua subsistem bekerja dengan baik, pengujian dilakukan dengan menggabungkan semua subsistem dalam alat.

10. Jika integrasi semua subsistem sesuai dengan rancangan selanjutnya dilakukan uji coba.
11. Menyimpulkan hasil penelitian dari hasil pengujian ini.

### **3.5 Teknik Analisis Data**

Teknik analisis data merupakan penelitian yang diperlukan peneliti untuk mendapatkan data dari keseluruhan sistem alat. Pengujian dilakukan untuk mengetahui apakah semua sistem berhasil atau gagal. Penelusuran referensi atau literatur di perpustakaan dan instansi sebagai bahan pelengkap, pendukung dan pembanding serta konsep dalam mencari solusi permasalahan. Studi pustaka tersebut berupa buku, *leaflet*, brosure, dan laporan penelitian terdahul. Referensi tersebut antara lain diperoleh dari buku-buku, jurnal, majalah, koran, dan internet (Sugiyono, 2006).

Untuk mendapatkan hasil yang diinginkan pada alat penghitung dan pemilah kerupuk putih pada *home industry* berbasis mikrokontroler harus dilakukan pengujian. Berikut adalah kriteria pengujian alat tersebut dengan parameter keberhasilan yang dilakukan.

#### **3.5.1 Pengujian Powersupply**

Pengujian *powersupply* dilakukan untuk mengetahui tegangan keluaran yang dihasilkan oleh *powersupply* yang digunakan pada alat penghitung dan pemilah kerupuk. Pegujian dilakukan dengan cara *powersupply* (a) 5V 5A dan (b) 12V 3A diberi tegangan sumber PLN, kemudian diukur tegangan *output*-nya menggunakan AVO meter. Pengujian *powersupply* dilihat pada Tabel 3.3

**Tabel 3.3 Pengujian Powersupply**

<b>Kondisi PSU</b>	<b>Kriteria Pengujian</b>	<b>Hasil Pengukuran</b>	<b>Tampilan</b>
Tidak aktif	0V		
PSU (a) aktif	5V		
PSU (b) aktif	12V		

### 3.5.2 Pengujian Sensor *Photo Infrared*

Pengujian sensor *photo infrared* ini dilakukan untuk mengetahui apakah sensor *photo infrared* dapat mendeteksi kerupuk yang menghalangi *photodiode* dan berapa tegangan yang dihasilkan oleh sensor *photo infrared* tersebut. Pengujian dilakukan dengan cara memberikan tegangan 5V ke sensor *photo infrared*, kemudian setelah diadjust sensor *photo infrared* dites dengan menghalangi dan tidak menghalangi *photodidanya* menggunakan kerupuk putih. Pengujian sensor *photo infrared* dilihat pada Tabel 3.4.

**Tabel 3.4 Pengujian Sensor *Infrared***

<b>Kondisi Sensor</b>	<b>Kriteria Pengujian</b>	<b>Hasil Pengukuran</b>	<b>Tampilan</b>
Tidak terhalang	5V		
terhalang	0V		

### 3.5.3 Pengujian Rangkaian *Driver Relay*

Pengujian *driver relay* dilakukan untuk mengetahui tegangan keluaran yang dihasilkan oleh *driver relay* pada kondisi aktif dan tidak aktif ketika diberi sumber tegangan *powersupply* 5V. pengujian *driver relay* dilihat pada Tabel 3.5.

**Tabel 3.5 Pengujian *Driver Relay***

<b>Kondisi Relay</b>	<b>Kriteria Pengujian</b>	<b>Hasil Pengukuran</b>	<b>Tampilan</b>
Tidak aktif	0V		
aktif	5V		

### 3.5.4 Pengujian LCD

Pengujian LCD ini dilakukan untuk mengetahui tegangan keluaran yang dihasilkan oleh LCD 2x16 ketika diberi tegangan *powersupply* 5V. pengujian dilihat pada Tabel 3.6 .

**Tabel 3.6 Pengujian LCD**

<b>Kondisi LCD</b>	<b>Kriteria Pengujian</b>	<b>Tampilan</b>
Tidak aktif	0V	
aktif	5V	

### 3.5.5 Pengujian Maket

Pengujian ini dilakukan untuk mengetahui keberhasilan mekanik pada maket pada saat *counting* kerupuk putih. Pengujian dilihat pada Tabel 3.7.

**Tabel 3.7 Pengujian Maket**

<b>Jumlah Kerupuk Putih</b>	<b>Pembacaan Alat</b>					<b>Hasil Pengujian</b>
	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	
1 keping						
2 keping						
4 keping						
8 keping						
16 keping						
32 keping						
64 keping						

## BAB IV

### HASIL PENELITIAN

#### 4.1 Deskripsi Hasil Penelitian

Berdasarkan rancangan penelitian yang telah dijabarkan pada penjelasan sebelumnya, maka peneliti akan mendeskripsikan hasil penelitian berupa prinsip kerja dan langkah kerja alat.

##### 4.1.1 Prinsip Kerja Alat

Pada alat penghitung dan pemilah kerupuk putih pada *home industry* berbasis mikrokontroler ini memiliki prinsip kerja untuk menghitung jumlah kerupuk yang diinginkan secara otomatis. Alat ini mempunyai dua lubang *rejector*, *rejector* (a) untuk membuang kerupuk yang rusak atau tidak sempurna, dan *rejector* (b) untuk memisahkan kerupuk dengan kondisi baik. Alat ini menggunakan motor DC untuk memindahkan kerupuk putih agar jatuh ke lubang *rejector*, motor tersebut akan menggerakkan roda akrilik supaya kerupuk putih dapat dipindahkan dari silo atas ke silo bawah. Untuk mendeteksi kerupuk putih alat pemilah dan penghitung kerupuk menggunakan sensor *photo infrared* dimana akan mendeteksi kerupuk yang melewati lubang *rejector* (b). Pada prinsip kerja alat penghitung kerupuk ini membutuhkan sumber daya listrik konstan menggunakan *powersupply* 5V 5A dan 12V 3A.

##### 4.1.2 Langkah Kerja Alat

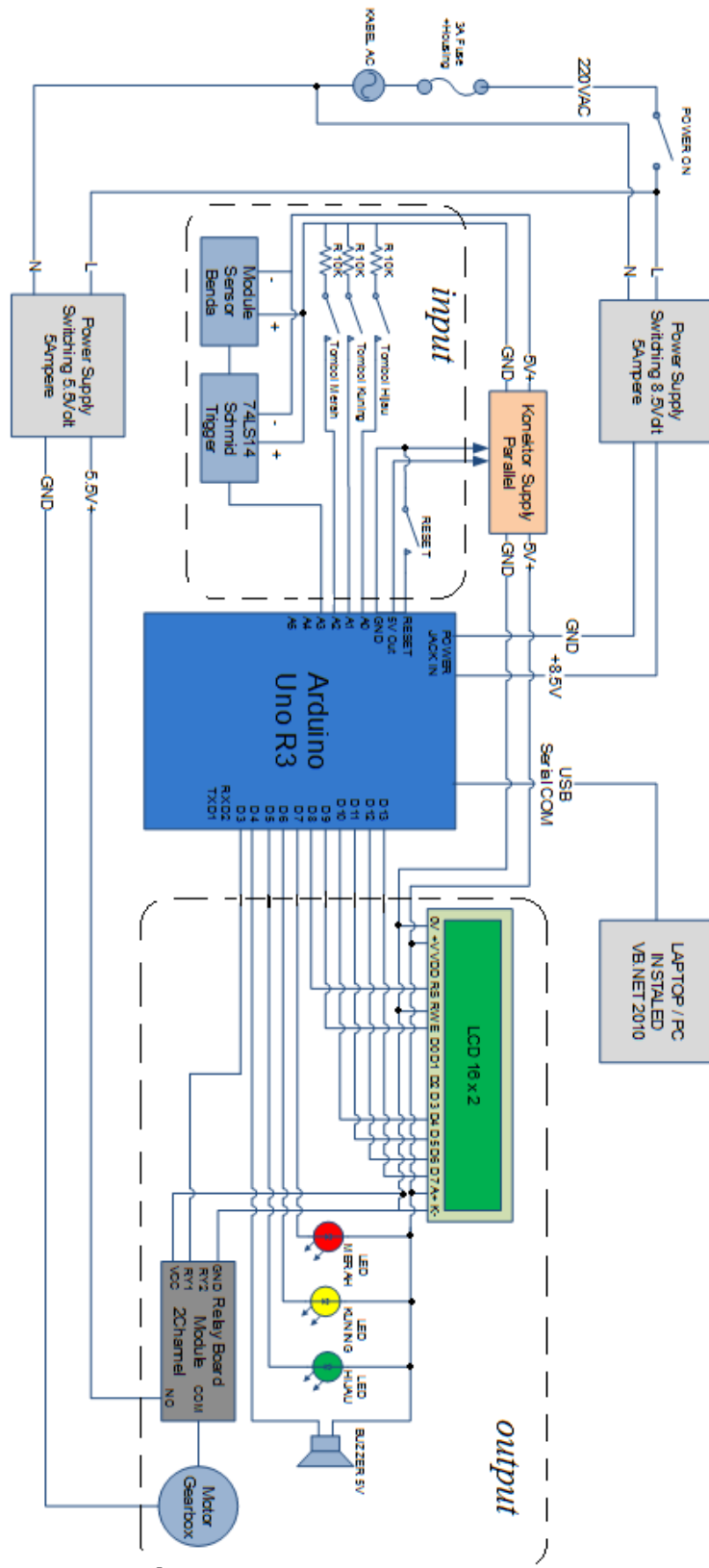
Berikut adalah langkah – langkah kerja alat penghitung kerupuk putih putih berbasis Arduino :



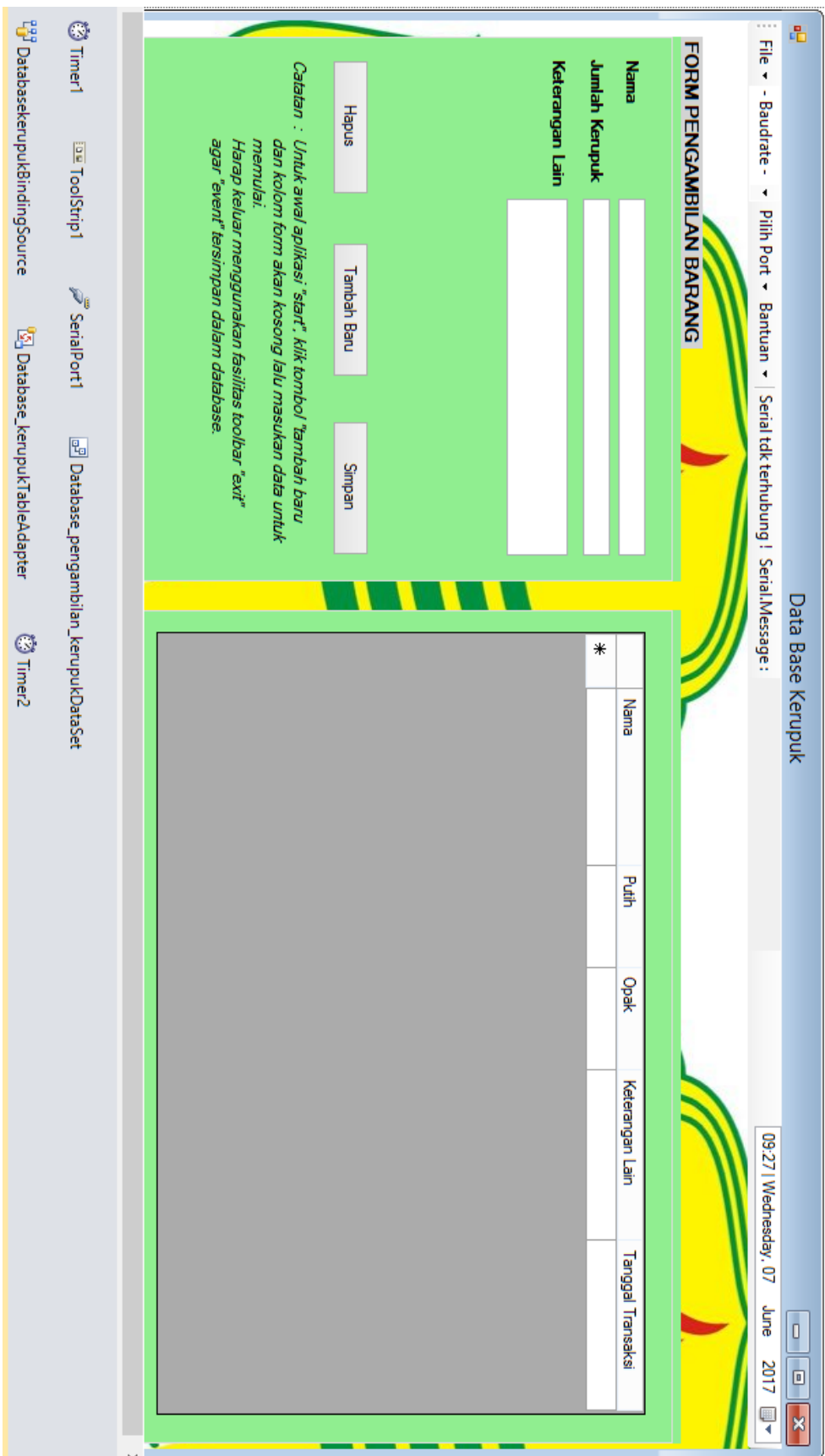
1. Nyalakan saklar utama pada alat, dan PC.
2. Jalankan program VB, lalu sambungkan kabel USB dari Arduino ke PC.
3. Lakukan pengecekan VB sudah tersambung dengan Arduino.
4. *Input* nama pedagang, jumlah kerupuk yang akan disetorkan, dan keterangan pembayarannya menggunakan PC, klik tambah baru.
5. Setelah data tabel pada VB bertambah, tekan *switch* pada alat untuk mengatur jumlah target agar sama dengan data pada VB sebelumnya.
6. Tekan *switch* pada alat untuk melakukan *running* alat sehingga proses *counting* dimulai.
7. Motor DC dan sensor *photo infrared* akan aktif selama proses *counting* berlangsung, jika *counting* selesai atau memenuhi target maka motor DC akan otomatis berhenti dan *buzzer* akan berbunyi.
8. Tekan *switch* untuk mematikan *buzzer*.
9. Ulangi langkah 4, 5, 6, 7, 8 hingga data pedagang selesai, jika ingin menghapus data sebelumnya klik hapus pada VB.
10. Jika sudah selesai klik simpan pada VB ke dalam format xcl.
11. *Edit* tabel pada excel sesuai yang dibutuhkan, lalu print.

#### **4.2 Analisis Hasil Penelitian**

Berdasarkan blok diagram dan *flowchart* yang telah dirancang pada penjelasan sebelumnya maka alat diimplementasikan oleh peneliti pada Gambar 4.1; Gambar 4.2; Gambar 4.3; dan Gambar 4.4.



Gambar 4.1 Diagram Wiring Alat



**Gambar 4.2** GUI (*Graphical User Interface*) Alat



**Gambar 4.3 Alat Penghitung Kerupuk Putih Berbasis Mikrokontroler**



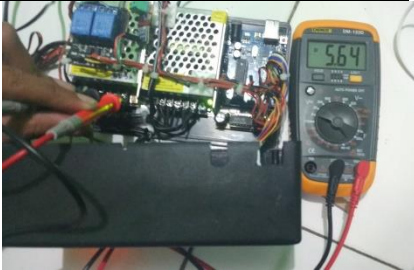
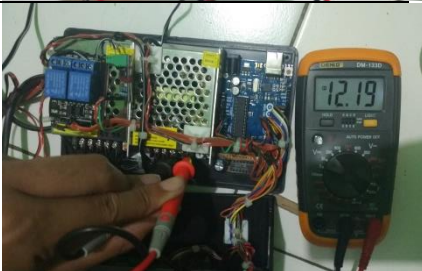
**Gambar 4.4 Console Alat**

Hasil pengujian yang telah dilakukan terhadap indikator-indikator penelitian dari alat penghitung kerupuk berbasis arduino. Pengujian ini dilakukan dengan perangkat keras dan perangkat lunak yang diprogram oleh Arduino IDE.

#### 4.2.1 Hasil Pengujian Powersupply

Hasil pengujian *powersupply* (PSU) dilakukan dengan mengukur tegangan pada kondisi aktif dan tidak aktif yaitu pada Tabel 4.1.

**Tabel 4.1 Hasil Pengujian Powersupply**

Kondisi PSU	Kriteria Pengujian	Hasil Pengukuran	Tampilan
Tidak aktif	0V	0V	0V
PSU (a) aktif	5V	5,64V	
PSU (b) aktif	12V	12,19V	

#### 4.2.2 Hasil Pengujian Sensor Photo Infrared

Pengujian Sensor *photo infrared* menggunakan program Arduino IDE yang ditunjukkan pada Gambar 4.5.

```

Praktik13  Praktik16  Praktik20
void setup() {
  Serial.begin(9600);
  // inialisasi pin13 sebagai output (tersambung ke LED)
  pinMode(13, OUTPUT);
}

unsigned int sensor;
int batasNyala=800;



void loop() {
  // baca nilai sensor di PIN A0 (Analog) dan kirim ke serial
  sensor=analogRead(A0);
  Serial.print("Nilai: ");
  Serial.println(sensor);
  // Jika jarak sensor dgn dinding makin dekat,nilai sensor
  // turun,besar nilai tsb dapat di pantau via serial monitor
  if(sensor<=batasNyala)
  {
    digitalWrite(13, HIGH); // LED 13 dinyalakan
  }
  else
  {
    digitalWrite(13, LOW); // LED 13 dipadamkan
  }
  delay(500); // delay 0,5 detik
}

```

**Gambar 4.5 Testing Program Sensor Photo infrared**

Hasil pengujian *sensor infrared* dilakukan dengan mengukur tegangan pada kondisi terhalang dan tidak terhalang oleh kerupuk yaitu pada Tabel 4.2.

**Tabel 4.2 Hasil Pengujian Sensor Infrared**

Kondisi Sensor	Kriteria pengujian	Hasil pengukuran	Tampilan
Tidak terhalang	5V	4,91V	
terhalang	0V	0,10V	

### 4.2.3 Hasil Pengujian Rangkaian *Driver Relay*

Pengujian *Driver Relay* menggunakan program Arduino IDE yang ditunjukkan pada Gambar 4.6.

```

Praktik13  Praktik16  Praktik20  Praktik21
const int pinKa = 10; //pinKanan Motor
const int pinKi = 11; //pinKiri Motor

void setup () {
  // inisialisasi digitalpin 10,11 sebagai output
  pinMode(pinKa, OUTPUT);
  pinMode(pinKi, OUTPUT);
  digitalWrite(pinKa,LOW);
  digitalWrite(pinKi,LOW);
}


void loop() {
  //Putar searah jarum jam 4 detik
  digitalWrite(pinKa,HIGH);
  digitalWrite(pinKi,LOW);
  delay(4000);
  //Stop 1 detik
  digitalWrite(pinKa,LOW);
  digitalWrite(pinKi,LOW);
  delay(1000);
  //Putar berlawanan arah jarum jam 4 detik
  digitalWrite(pinKa,LOW);
  digitalWrite(pinKi,HIGH);
  delay(4000);
  //Stop 1 detik
  digitalWrite(pinKa,LOW);
  digitalWrite(pinKi,LOW);
}


```

**Gambar 4.6** *Testing Program Relay*

Hasil pengujian *relay* untuk switch motor DC dilakukan dengan mengukur tegangan *output* ketika *relay* aktif dan tidak aktif, yaitu pada Tabel 4.3.

**Tabel 4.3** Hasil Pengujian Rangkaian *Driver Relay*

Kondisi <i>relay</i>	Kriteria pengujian	Hasil pengujian	Tampilan
Tidak aktif	0V	0,08V	

aktif	5V	4,86V	
-------	----	-------	--

#### 4.2.4 Hasil Pengujian LCD

Pengujian LCD menggunakan program Arduino IDE yang ditunjukkan pada Gambar 4.7.

```

Praktik13  Praktik16  Praktik20
#include <LiquidCrystal.h>

// inisialisasi koneksi pin LCD dengan pin Arduino
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
int sensorT=A0;
float suhu;

void setup() {
  // inisialisasi LCD yang digunakan:
  lcd.begin(16, 2);
  lcd.print("Temperatur:");
}

void loop() {
  // baca temperatur dari LM35
  int nilDigital=analogRead(sensorT);
  // konversi ke derajat Celcius
  suhu=(5.0 * nilDigital * 100.0)/1024.0;
  // set kolom 0 dan baris 1 LCD dan tampilkan
  lcd.setCursor(0, 1);
  lcd.print(suhu);
  lcd.print("Celcius");
  delay(300); // delay 300ms
}



```

**Gambar 4.7 Testing Program LCD**

Hasil pengujian LCD dilakukan dengan mengukur tegangan *input* ketika LCD aktif dan tidak aktif, yaitu pada Tabel 4.4.



Tabel 4.4 Hasil Pengujian LCD 16x2

Kondisi LCD	Kriteria pengujian	Hasil pengujian	Tampilan
Tidak aktif	0V	0V	
aktif	5V	5V	

#### 4.2.5 Hasil Pengujian Maket

```

Penghitung_Kerupuk_Mawar sub_Controloutput sub_InitIO sub_Readinput$ sub_Stepprogram
#include <LiquidCrystal.h>
#include <EEPROM.h>

//          RS,E,D4,D5,D6,D7
LiquidCrystal lcd (8,9,10,11,12,13);

//__ DEFINE SHORTCUT
#define buzzOn(n)    digitalWrite(bzz, LOW);    delay(n);
#define buzzOff(n)  digitalWrite(bzz, HIGH);    delay(n);
#define buzzOn      digitalWrite(bzz, LOW);
#define buzzOff     digitalWrite(bzz, HIGH);
#define lredOn      digitalWrite(lred, LOW);
#define lredOff     digitalWrite(lred, HIGH);
#define lylwOn      digitalWrite(lylw, LOW);
#define lylwOff     digitalWrite(lylw, HIGH);
#define lgrnOn      digitalWrite(lgrn, LOW);
#define lgrnOff     digitalWrite(lgrn, HIGH);
#define ry1On       digitalWrite(ry1, LOW);
#define ry1Off      digitalWrite(ry1, HIGH);

//__ ENUM SCAN TOMBOL
enum scanInputs { btNone = 0, btRed, btYlw , btGrn, countCr, };
scanInputs Tombol;
enum scanSensor { ssNone = 0, ssCapt,};
scanSensor Sensor;

```

Gambar 4.8 Testing Program Maket

Hasil pengujian maket dilakukan untuk mendapatkan hasil perbandingan *input* dan *output* perhitungan, yaitu pada Tabel 4.5.

Tabel 4.5 Hasil Pengujian Maket

Jenis Uji Coba	Jumlah Kerupuk Putih (i)	Hasil Pembacaan Alat (O <sub>n</sub> )				
		1	2	3	4	5
A	1 keping	1	1	1	1	1
B	2 keping	2	2	2	2	2
C	4 keping	4	4	4	4	4
D	8 keping	8	8	8	8	8
E	16 keping	16	14	16	16	18
F	32 keping	32	29	35	38	40
G	64 keping	64	73	76	79	82

Setelah dilakukan penghitungan kerupuk dengan alat didapatkan persentase *error* pada Tabel 4.6. Penghitungan rumus persentase *error* menggunakan rumus:

$$E = \frac{\left(\frac{O_1 - i}{i}\right) + \left(\frac{O_2 - i}{i}\right) + \left(\frac{O_3 - i}{i}\right) + \left(\frac{O_4 - i}{i}\right) + \left(\frac{O_5 - i}{i}\right)}{\text{Banyak uji coba}}$$

Keterangan: E = Nilai *Error*

O<sub>n</sub> = hasil pembacaan alat

i = jumlah kerupuk

Tabel 4.7 Persentase *Error* Penghitungan Alat

Jenis Uji Coba	Jumlah kerupuk putih	Hasil Pembacaan Alat Dan Persentase <i>Error</i>					Rata-rata <i>error</i>
		1	2	3	4	5	
A	1 keping	1 (0%)	1 (0%)	1 (0%)	1 (0%)	1 (0%)	0%
B	2 keping	2 (0%)	2 (0%)	2 (0%)	2 (0%)	2 (0%)	0%
C	4 keping	4 (0%)	4 (0%)	4 (0%)	4 (0%)	4 (0%)	0%
D	8 keping	8 (0%)	8 (0%)	8 (0%)	8 (0%)	8 (0%)	0%
E	16 keping	16 (0%)	14 (12%)	16 (0%)	17 (6%)	18 (12%)	6%
F	32 keping	32 (0%)	29 (9%)	35 (9%)	38 (18%)	40 (25%)	12.2%
G	64 keping	64 (0%)	73 (14%)	76 (18%)	79 (23%)	82 (28%)	16.6%

### 4.3 Pembahasan

Pengujian alat secara keseluruhan dilakukan dengan menggabungkan semua subsistem, mulai dari sensor *photo infrared*, switch, rangkaian realy pensaklar motor DC, motor DC penggerak, LCD, *buzzer*, LED, Arduino sebagai mikrokontroler. Berdasarkan seluruh pengujian yang telah dilakukan, diketahui bahwa alat penghitung dan pemilah kerupuk putih pada home industry berbasis mikrokontroler ini berfungsi sesuai dengan perancangan. Hasil pengujian didapat melalui pengujian *powersupply*, pengujian sensor *photo infrared*, pengujian rangkaian *driver*, pengujian LCD, dan pengujian maket.

Pada tabel 4.1 hasil pengujian terhadap *powersupply* dengan tegangan keluaran lebih dari atau sama dengan 12 volt dapat membagi tegangan ke semua komponen elektronik dan modul pada alat.

Pada tabel 4.2 hasil pengujian terhadap sensor *infrared* ketika mendeteksi kerupuk putih. Tegangan keluaran saat terhalang adalah 0V dan ketika tidak ada yang menghalangi sensor adalah 5V. Sensor *infrared* dapat mendeteksi kerupuk yang melintas di depannya dengan jarak 1cm yang dapat di-adjust dengan memutar trimpot menggunakan obeng plus.

Pada tabel 4.3 hasil pengujian terhadap *driver relay* ketika diaktifkan dan dinon aktifkan. dengan tegangan keluaran saat diaktifkan sebesar 5V dapat mensaklar motor DC *gearbox* sehingga dapat berputar.

Pada tabel 4.4 hasil pengujian terhadap LCD ketika diaktifkan dan dinonaktifkan. Pada saat diaktifkan, LCD menampilkan karakter huruf dan angka. Ketika saklar *power* di-non aktifkan maka LCD otomatis mati.

Pada Tabel 4.5 hasil pengujian terhadap maket ketika program running. Pengujian jumlah kerupuk dilakukan sebanyak 5 kali untuk mendapatkan nilai *error* yang ditunjukkan pada Tabel 4.7.

#### **4.4 Aplikasi Hasil Penelitian**

Berdasarkan hasil pengujian yang telah dilakukan dan dijabarkan pada penjelasan sebelumnya maka alat penghitung dan pemilah kerupuk putih pada *home industry* berbasis mikrokontroler dapat diaplikasikan sebagai alat *counting* kerupuk pada *home industry* kerupuk putih, selain itu dapat dimanfaatkan untuk salah satu solusi dan pengembangan dalam pemanfaatan teknik elektronika di bidang industri pangan.

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Berdasarkan hasil pengujian dan penelitian pada alat penghitung dan pemilah kerupuk putih pada *home industry* berbasis mikrokontroler dapat disimpulkan bahwa:

1. Perancangan sistem alat dapat terealisasi sesuai dengan perencanaan menggunakan metode rekayasa teknik meliputi perencanaan, analisa kebutuhan, perancangan, pengujian, implementasi *hardware* dan *software*.
2. Alat mampu melakukan set target jumlah kerupuk yang diinginkan oleh pengguna. Ketika proses *running*, motor DC aktif memindahkan kerupuk putih ke silo, bersamaan dengan sensor *photo infrared* aktif menghitung jumlah kerupuk yang sudah dimasukkan ke silo. Kerupuk putih yang tidak sempurna berhasil dipisahkan melalui lubang *rejector*, dan hasil penghitungan nilai rata-rata terbesar adalah 16.6%.

#### 5.2 Saran

Dalam proses perancangan alat peneliti menyadari bahwa masih banyak kekurangan dan kelemahan maka dari itu peneliti memberikan saran sebagai berikut kepada pihak-pihak yang ingin mengembangkan penelitian ini:

1. Konstruksi maket harus lebih presisi agar tidak ada kerupuk yang tersangkut.
2. Pemilihan torsi motor *gearbox* sebagai penggerak harus sesuai dengan beban kerupuk.
3. Sensor *photo infrared* harus selalu di-*adjust* sebelum memulai proses *counting*.

## DAFTAR PUSTAKA

- [FT] Fakultas Teknik. (2015). Buku Panduan Penyusunan Skripsi Dan Non Skripsi. Jakarta: Fakultas Teknik, Universitas Negeri Jakarta.
- A.G, Aziz. (2017). Protoype Robot Line Follower Pengangkut Sampah Dengan Memanfaatkan Solar Cell Berbasis Mirokontroler Arduino Mega 2560 [Skripsi]. Jakarta: Fakultas Teknik, Universitas Negeri Jakarta.
- A.M, Hirin. (2011). Belajar Tuntas (Dari Dasar Sampai Mahir) VB.NET 2010. Kendal: Prestasi Pustaka Publisher.
- Agromedia, Redaksi. (2007). 22 Peluang Bisnis Makanan Untuk *Home Industry*. Jakarta: Agromedia.
- Andrianto, Heri & Darmawan, Aan. (2016). Arduino Belajar Cepat Dan Pemrograman. Bandung: Informatika.
- Jatmiko, P. (2015). PLC, HMI and Industrial Part Volume 1 of Industry. Bogor: Kartanagari.
- Kadir, Abdul. (2012). Panduan Praktis Mempelajari Aplikasi Mikrokontroler Dan Pemrogramannya Menggunakan Arduino. Yogyakarta: Andi.
- Karyanto, Sari. (2011). Prototipe Mesin Penyeleksi Dan Penghitung Uang Logam Berbasis Mikrokontroler ATMEL AT90S8515. [Skripsi]. Universitas Diponegoro.
- Koswara, Eko. (2013). *Visual basic For Beginner*. Yogyakarta: Mediakom.
- Petruzella, Frank D. 2001. Elektronik Industri. Terjemahan sumanto. Edisi kedua. Yogyakarta: Andi.
- Primdahl. (1990). *Coin Sorter With Counter And Break Mechanism*. United State Patent, 1-6.
- Sugiyono. (2006). Metode Penelitian Kuantitatif Dan R&D. Bandung: Alfabeta.
- Widianto, Nur. (2015). Rancang Bangun Otomatis Alat Penghitung Benih Ikan. [Tugas Akhir]. Universitas Mercubuana.
- Winoto, Ardi. (2010). Mikrokontroler AVR ATmega8/32/16/8535 dan Pemrogramannya dengan Bahasa C Pada WinAVR. Bandung: Informatika.

## LAMPIRAN

### Data List Code

```
'----- Bismillahhirrohmanirrohim -----  
' _____ Library-library _____  
Imports System.IO ' Library Serial port  
Imports System.IO.Ports ' Library Serial Input  
Output  
Imports System.Threading  
Imports System.ComponentModel  
Imports System.IO.Directory  
Imports System.Data.OleDb ' Library untuk database  
Imports Microsoft.Office.Interop.Excel ' library refference  
untuk export database ke excel  
Imports Microsoft.Office.Interop ' library refference  
untuk export database ke excel  
  
Public Class Form1  
  
    ' _____ define function _____  
    Dim Baud_selected As String  
    Delegate Sub setTextCallBack(ByVal [Text] As String)  
    Public Delegate Sub SerialDataReceivedEventHandler _  
        (ByVal sender As Object, ByVal e As SerialDataReceivedEventArgs)  
  
    Dim myPorts As String()  
    Dim Port_select As String  
  
    Dim Received_data As String = ""  
    Dim buffer_data As String = ""  
    Dim Count_Pelanggan As Integer  
  
    ' _____ Main form awal aplikasi di run  
  
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles MyBase.Load  
        'TODO: This line of code loads data into the  
'Database_pengambilan_kerupukDataSet.Database_kerupuk' table. You can  
move, or remove it, as needed.  
  
Me.Database_kerupukTableAdapter.Fill(Me.Database_pengambilan_kerupukDataS  
et.Database_kerupuk)  
  
        Timer1.Enabled = True ' Timer 1 Open scanning  
port input  
        Timer2.Enabled = False  
        Connect_Port.Enabled = True ' Tombol Connect port  
izinkan  
        Disconnect_port.Enabled = False ' Tombol disconnect port  
tidak diizinkan  
        Tanggal_Capture.Text = Date.Now ' Simpan tanggal sekarang  
ke textbox. Textbox kondisi enable = false dan dihidden dibelakang tabel  
        Me.Addnew_button.PerformClick() ' Otomatis tambah baru  
database action klik  
  
    End Sub  
  
    ' _____ Binding source dari datagridview ke database ms.access  
  
    Private Sub DBkerupuk1BindingNavigatorSaveItem_Click(ByVal sender As  
System.Object, ByVal e As System.EventArgs)  
  
        Me.Validate()  

```

```

        Me.DatabasekerupukBindingSource.EndEdit()

Me.Database_kerupukTableAdapter.Update(Me.Database_pengambilan_kerupukData
aSet)

    End Sub

    ' _____ Fungsi menambahkan cell (baris dan kolom) baru pada
datagridview _____
    Private Sub Addnew_button_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Addnew_button.Click

        DatabasekerupukBindingSource.AddNew()           ' Tambahkan cell
baru pada datagridview
        Tanggal_Capture.Text = Date.Now                 ' Masukkan data
string tanggal hari ini pada textbox
        TextBox1.Focus()                                ' Setelah
dimasukan fokuskan pengisian pada textbox 1

    End Sub

    ' _____ Fungsi untuk menghapus cell terpilih pada datagridview,
menghapus hanya untuk per-cell per action _____
    Private Sub Delete_button_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Delete_button.Click

        DatabasekerupukBindingSource.RemoveCurrent()

    End Sub

    ' _____ Fungsi untuk menyimpan data baru dari form isi ke
datagridview _____
    Private Sub Save_button_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles Save_button.Click

        Try
            If (((TextBox1.Text = Nothing) And (TextBox2.Text = Nothing)
And (TextBox7.Text = Nothing)) Then

                MsgBox("Mohon untuk mengisi Form, tidak menyimpan form
kosong")

            Else

                DatabasekerupukBindingSource.EndEdit()

                Database_kerupukTableAdapter.Update(Database_pengambilan_kerupukData
aSet.Database_kerupuk)
                MessageBox.Show("Data Tersimpan !")
                Me.Addnew_button.PerformClick()
            End If

            Catch ex As Exception
                MessageBox.Show("An error occured !")
            Exit Sub

        End Try

    End Sub

    ' _____ Fungsi "exit button toolbar" untuk keluar atau menghentikan
aplikasi secara sistematis (bukan force close [x]) _____
    Private Sub ExitToolStripMenuItem_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ExitToolStripMenuItem.Click

        Dim result_caption As DialogResult

```



```

        TextBox7.Text = "Terakhir akses aplikasi"
    ' Isi textbox dengan pesan info

        result_caption = MessageBox.Show("Anda Yakin Keluar ?", "PILIHAN",
        MessageBoxButtons.YesNo) ' Tampilkan jendela popup peringatan untuk
        menghentikan aplikasi
        If result_caption = DialogResult.Yes Then
    ' Bila tombol "yes" ditekan, maka ....
            Save_button.Enabled = True
            DatabasekerupukBindingSource.EndEdit()
    ' Database pada datagridview akhiri editing

    Database_kerupukTableAdapter.Update(Database_pengambilan_kerupukDataSet.D
    atabase_kerupuk) ' Update data base pada datagridview sebelum aplikasi
    tutup

        Try
    ' Tes
            SerialPort1.Close()
    ' Tutup serial port apapun
            Disconnect_port.Enabled = False
    ' Matikan fungsi akses tombol "Disconnect"
            Connect_Port.Enabled = True
    ' Matikan fungsi akses tombol "Connect"

        Catch ex As Exception
    ' maka
            Exit Sub

        End Try
        ' _____ Close Serial Port _____

        Me.Close()
    ' Tutup aplikasi
        ElseIf result_caption = DialogResult.No Then
    ' Bila tombol "no" ditekan maka nothing

        End If

    End Sub

    ' _____ Fungsi untuk menampilkan aplikasimaker

```

---

```

    Private Sub CreditToolStripMenuItem_Click(ByVal sender As
    System.Object, ByVal e As System.EventArgs) Handles
    CreditToolStripMenuItem.Click

        Dim credit_caption As DialogResult

        credit_caption = MessageBox.Show("MAULANA .. NIM 5215125336",
    "Credit Program", MessageBoxButtons.OK)
        If credit_caption = DialogResult.OK Then
            End If

        End Sub

    ' _____ Fungsi untuk export seluruh data dari datagridview ke dalam
    format excel file

```

---

```

    Private Sub SaveAsToolStripMenuItem_Click(ByVal sender As
    System.Object, ByVal e As System.EventArgs) Handles
    SaveAsToolStripMenuItem.Click

        'On Error GoTo errorhandler

```

```

        If ((DataGridView1.Columns.Count = 0) Or
(DataGridView1.Rows.Count = 0)) Then
            Exit Sub
        End If

        Dim dSet As New DataSet
        dSet.Tables.Add()
        Dim count As Integer = -1
        Dim dr1 As DataRow

        For i As Integer = 0 To DataGridView1.RowCount - 1
            dr1 = dSet.Tables(0).NewRow

            For j As Integer = 0 To DataGridView1.Columns.Count - 1
                If DataGridView1.Columns(j).Visible = True Then
                    count = count + 1

                    dr1(count) = DataGridView1.Rows(i).Cells(j).Value
                End If
            Next

            count = -1
            dSet.Tables(0).Rows.Add(dr1)
        Next

        Dim newExcel As New Excel.Application
        Dim wBook As Excel.Workbook
        Dim wSheet As Excel.Worksheet

        wBook = newExcel.Workbooks.Add()
        wSheet = wBook.ActiveSheet()

        Dim dt As System.Data.DataTable = dSet.Tables(0)
        Dim DC As System.Data.DataColumn
        Dim dr As System.Data.DataRow
        Dim colIndex As Integer = 0
        Dim rowIndex As Integer = 0

        For Each DC In dt.Columns
            colIndex = colIndex + 1
            newExcel.Cells(1, colIndex) = DC.ColumnName
        Next

        For Each dr In dt.Rows
            rowIndex = rowIndex + 1
            colIndex = 0
            For Each DC In dt.Columns
                colIndex = colIndex + 1
                newExcel.Cells(rowIndex + 1, colIndex) =
dr(DC.ColumnName)
            Next
        Next

        wSheet.Columns.AutoFit()

        Count_Pelanggan += 1

        Dim saveFileDialog1 As New SaveFileDialog()
        ' saveFileDialog1.Filter = "Excel Workbook|*.xls|Excel
Workbook 2010-2011|*.xlsx"
        saveFileDialog1.Filter = "Excel Workbook 2010-2011|*.xlsx|Excel
Workbook 2003|*.xls"
        saveFileDialog1.Title = "Save database ke Excel File"

```

```

'          saveFileDialog1.FileName = "File Baru " &
Now.ToShortDateString & ".xlsx"
saveFileDialog1.FileName = "File Baru 00" & Count_Pelanggan &
" .xlsx"
saveFileDialog1.ShowDialog()
' show directory
saveFileDialog1.InitialDirectory = "D:/"
If saveFileDialog1.FileName <> "" Then
    Dim fs As System.IO.FileStream =
CType(saveFileDialog1.OpenFile(), System.IO.FileStream)
    fs.Close()

End If

Dim strFilename As String = saveFileDialog1.FileName
Dim blnFileopen As Boolean = False

Try
    Dim fileTemp As System.IO.FileStream =
System.IO.File.OpenWrite(strFilename)
    fileTemp.Close()
Catch ex As Exception
    blnFileopen = False
    Exit Sub
End Try

If System.IO.File.Exists(strFilename) Then
    System.IO.File.Delete(strFilename)
End If

Exit Sub

'errorhandler:

'          MsgBox(Err.Description)
'Exit Sub

End Sub

' _____ Fungsi "save button toolbar" untuk menyimpan data yang telah
diisi pada form ke database _____
Private Sub Save_database_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Save_database.Click

    Me.Save_button.PerformClick()          ' Jalankan aplikasi melalui
penekanan tombol save_button database secara tidak langsung

End Sub

' _____ Fungsi "Koneksi button toolbar" untuk menyambungkan fungsi
serial port aplikasi dan microcontroller _____
Private Sub Connect_Port_Click_1(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Connect_Port.Click

    Try
        SerialPort1.PortName = sub_pilihport.Text          ' ambil data
baudrate dari toolbar dropdown combo baud
        SerialPort1.BaudRate = Combo_baudtool.Text          ' ambil data
port com yang terbaca dan terpilih dari dropdown toolbar
        SerialPort1.Open()          ' Buka serial
port comm
        Serial_status.Text = "Serial Terhubung!"          ' Ubah text
notifikasi pada toolbar
        Connect_Port.Enabled = False          ' akses
tombol "port connect" dimatikan
    
```

```

        Disconnect_port.Enabled = True           ' akses
tombol "port disconnect" diaktifkan
        Timer2.Enabled = True                   ' Jalankan
fungsi timer2
        Serial_tools.Text = Port_select         ' Ubah nama
index toolbar menjadi nama comm yang terbaca pada komputer

        Catch ex As Exception
            MsgBox("Pilih baudrate dahulu !")   ' Bila ada
error maka akan mengarah pada pilih baudrate terlebih dahulu
            Combo_baudtool.Focus()             ' Fokuskan
editing pada kolom combo

        End Try

    End Sub

' _____ Fungsi "Memutuskan button toolbar" untuk menutup fungsi dari
serial port aplikasi _____
    Private Sub Disconnect_port_Click_1(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles Disconnect_port.Click

        Try
            SerialPort1.Close()                 ' Tutup
portserial
            Serial_status.Text = "Serial Diputus" ' Ubah label
text serial menjadi diputus
            Connect_Port.Enabled = True         ' Button
"connect" akses yes
            Disconnect_port.Enabled = False     ' Button
"disconnect" akses no
            Timer2.Enabled = False             ' Matikan
Timer 2. Akses serial tidak bisa dipakai
            Serial_tools.Text = "Pilih Port"    ' Ubag label
toolbar text kembali menjadi "Pilih Port"

        Catch ex As Exception

            MsgBox("Komunikasi Port hilang atau terputus, koneksi
ulang !")
            Connect_Port.Enabled = False
            Disconnect_port.Enabled = True
            Serial_tools.Text = "Pilih Port"    ' Ubag label
toolbar text kembali menjadi "Pilih Port"
        End Try

    End Sub

' _____ Fungsi timer1 dengan statement-statement
_____
    Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick

        Me.myPorts = IO.Ports.SerialPort.GetPortNames() '
Baca com usb yang terbaca

        For Each Me.Port_select In myPorts        '
step for setiap port yang ada pada com usb
            Me.sub_pilihport.Text = Port_select  '
Ubah text subpilihport menjadi nama port yang terbaca oleh Serial port
        Next Port_select

    End Sub

```

```

' _____ Subfungsi untuk mengolah data serial yang masuk dan
mengkomparasinya _____
Function ReceiveSerialData() As String

    Dim Incoming As String
    Try
        'Incoming = SerialPort1.ReadLine()
        Incoming = SerialPort1.ReadExisting()      ' Baca
seluruh isi data pada buffer serial
        If Incoming Is Nothing Then
            Return "Nothing" & vbCrLf
        Else
            Return Incoming
        End If
    Catch ex As TimeoutException
        Return "Tidak ada data masuk"

    End Try
End Function

' _____ Fungsi timer2 untuk membaca portserial yang terbaca pada USB
dan mengupdatenya pada toolbar _____
Private Sub Timer2_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer2.Tick

    If SerialPort1.IsOpen = True Then
        Received_data = SerialPort1.ReadExisting()      '
Pindahkan data buffer serial ke "received data" format string
        ts_serialmessage.Text = Received_data          ' Cetak
sting ke textbox read only true

        If Received_data = "Save" & vbCrLf Then      '
Komparasi data serial bila "save" maka
            'Me.Save_button.PerformClick()
            Savedata_serialCapture()                  '
Jalankan statement subfungsi "Savedata_serialCapture()" yang sudah atau
tidak diisi ada form
        End If
    End If

End Sub

' _____ Fungsi untuk menyimpan data khusus pada fungsi panggil dari
command serial microcontroller _____
Private Sub Savedata_serialCapture()

    Count_Pelanggan += 1      ' Increment nilai Count_Pelanggan
sebanyak 1 decimal

    If (((TextBox1.Text = Nothing) And (TextBox2.Text = Nothing) And
(TextBox7.Text = Nothing)) Then      ' Bila form isi kosong maka ....
        TextBox1.Text = "Pelanggan " & Count_Pelanggan      ' Isi
form textbox1 otomatis
        TextBox2.Text = "0"      ' Isi
form textbox2 otomatis
        TextBox7.Text = "Simpan Otomatis dari Microcontroller"      ' Isi
form textbox7 otomatis
    End If

    DatabasekerupukBindingSource.EndEdit()      '
Akhiri editing database

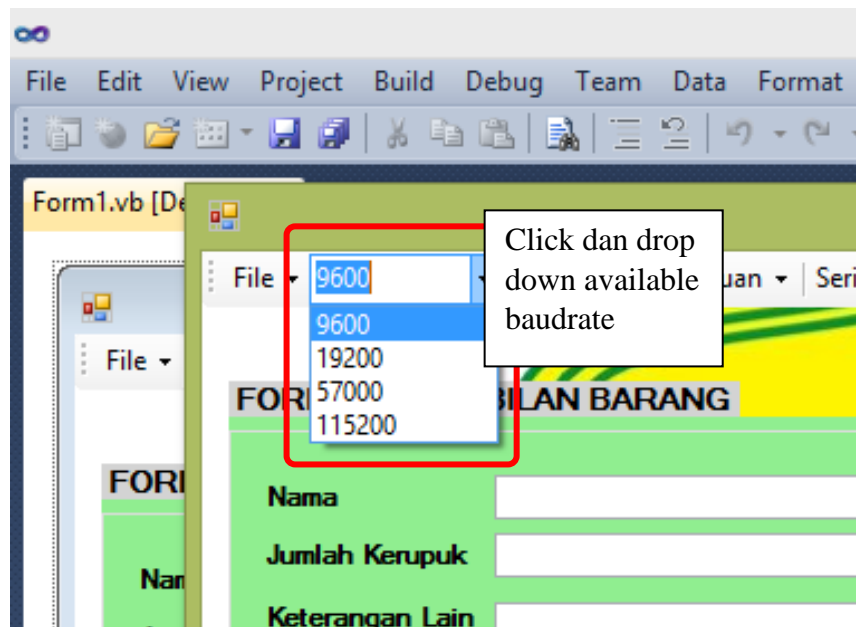
    Database_kerupukTableAdapter.Update(Database_pengambilan_kerupukDataSet.D
atabase_kerupuk)      'update database pada datagridview

```

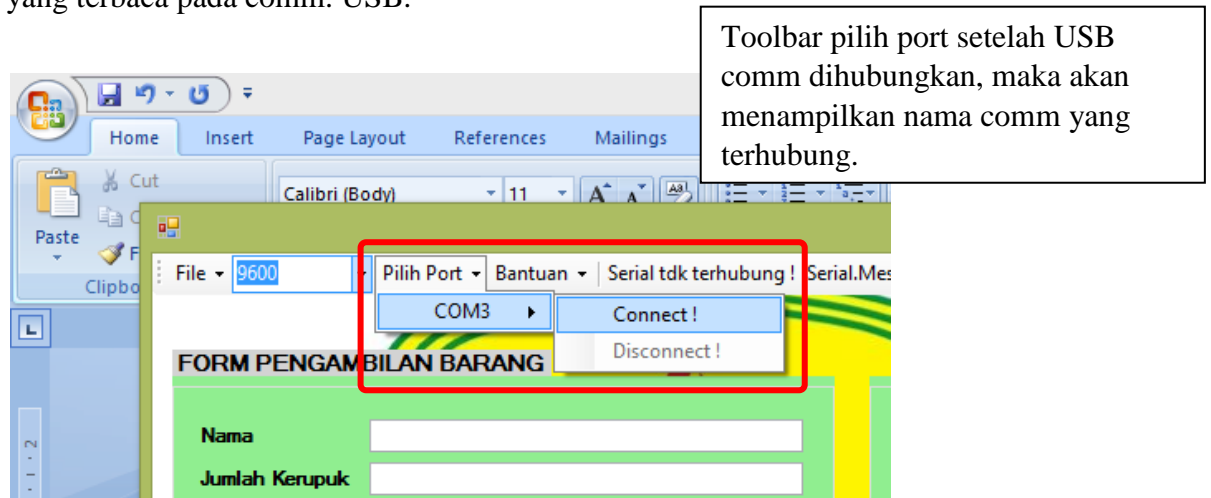
```
Me.Addnew_button.PerformClick()  
Shortcut klik sendiri tombol "tambah baru"  
End Sub  
  
End Class
```

## PANDUAN PENGGUNAAN ALAT PEMILAH DAN PENGHITUNG KERUPUK PUTIH BERBASIS MIKROKONTROLER

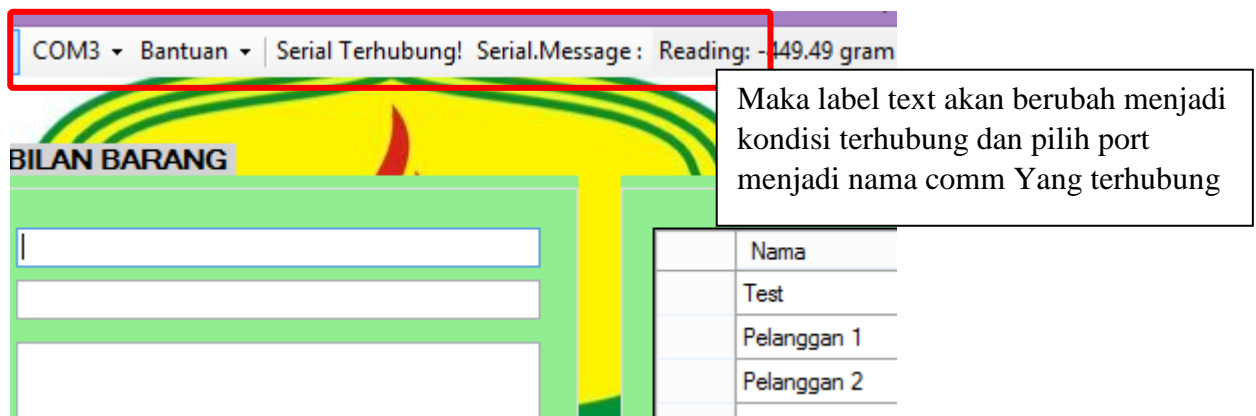
Pilih *baudrate* komunikasi serial, agar data dapat diterima dan dibaca dengan normal.



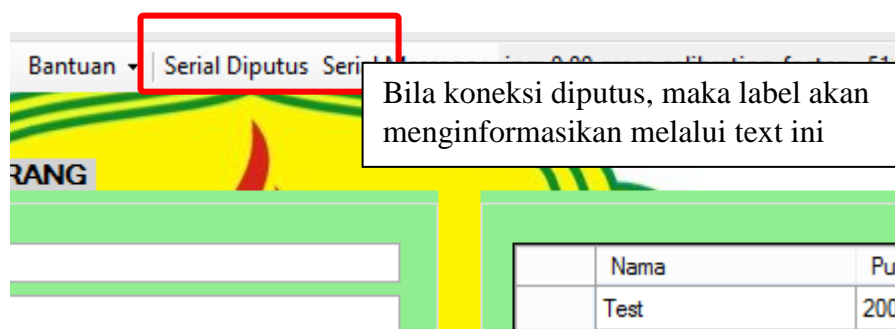
Pilih port Action perform click : dropdowntoolbar akan menampilkan port serial yang terbaca pada comm. USB.



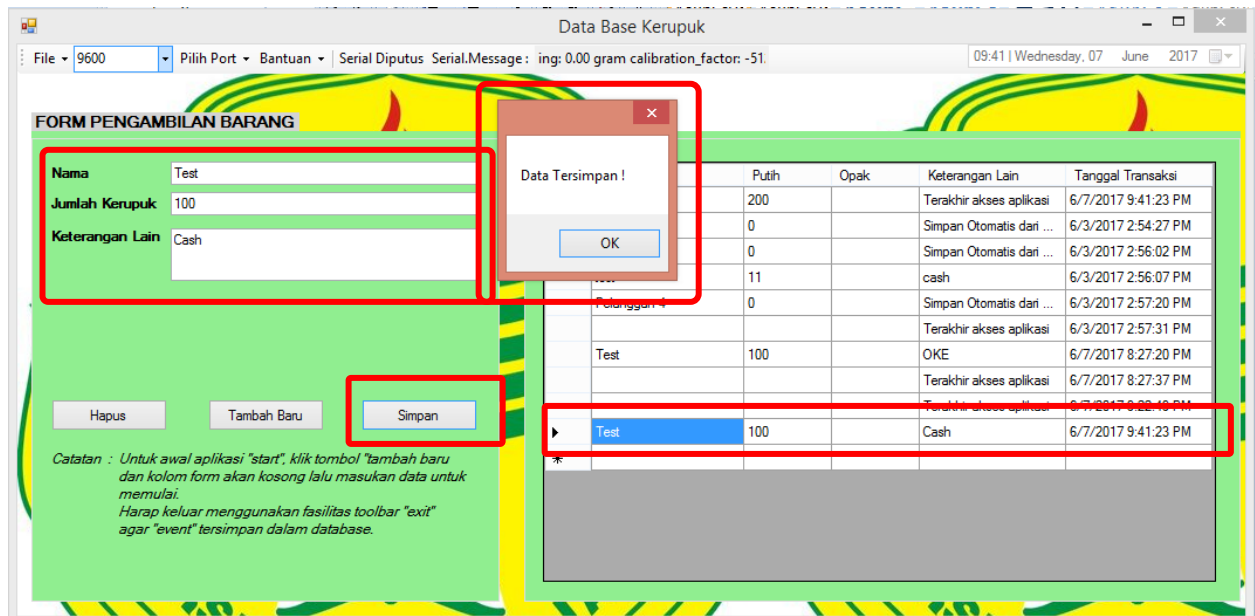
Apabila di klik *connect* maka portserial akan terhubung dan serial status akan muncul.



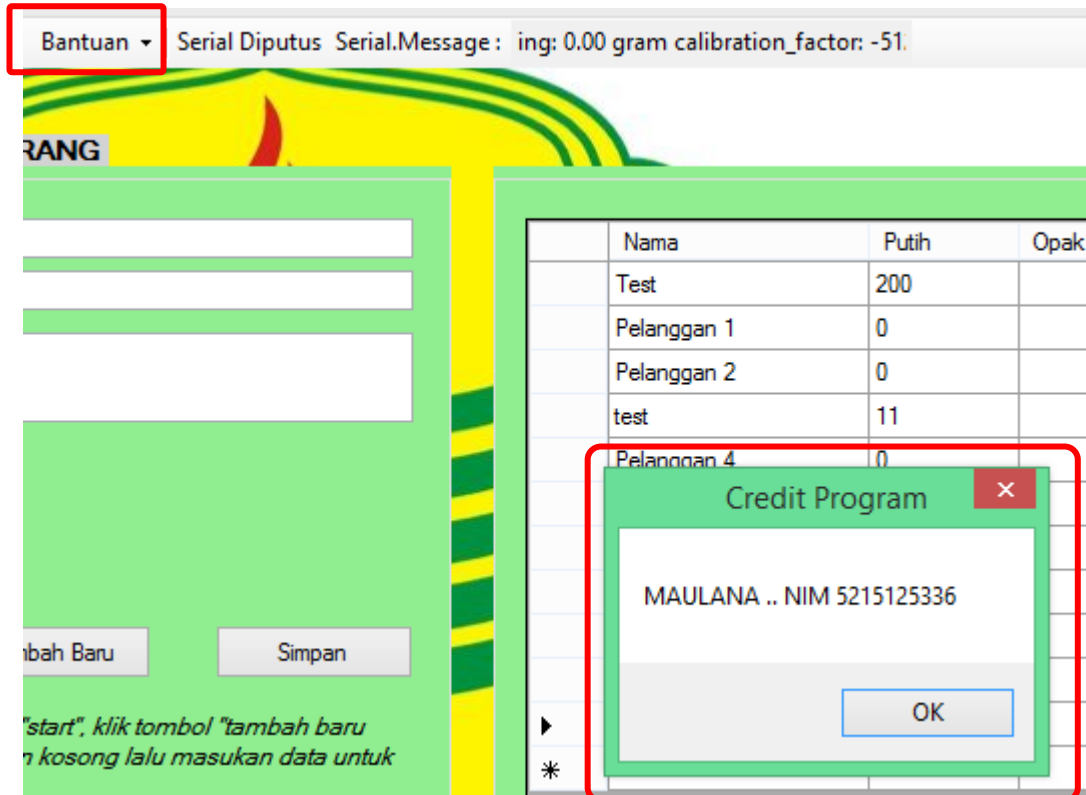
Dan apabila serialport di disconnect maka serial status akan berubah menjadi serial diputus



Pengisian form. Apabila tombol klik ditekan maka data akan tersimpan pada database dan jendela notifikasi telah tersimpan akan muncul.



Bila toolbar bantuan ditekan maka akan muncul jendela popup credit software







Tekan tombol *power* yang ada di sebelah kanan atas hingga LED indikator menyala dan LCD menampilkan “Select Menu”. Tekan *switch* MODE (hijau)/BACK (merah) untuk memindahkan kursor. Tekan *switch* SET (kuning) untuk memilih “Target”.



Jika memilih menu “Target” maka tampilannya seperti gambar di atas. Tekan SET untuk menambah 1 angka pada angka yang diberikan pointer di atasnya, angka akan bertambah dari 0-9 dan kembali ke 0. Tekan MODE untuk

memindahkan pointer dari satuan, puluhan, ratusan, ribuan (kanan ke kiri). Tekan BACK untuk kembali ke “Select Menu”.



Jika memilih menu “Run” maka tampilan LCD seperti gambar di atas. Tekan SET untuk mereset *counting* mulai dari 0000. Tekan MODE hingga indikator Motor ON (LED hijau) menyala untuk menngerakkan Motor *Gearbox*. Tekan MODE hingga indikator Motor OFF (LED merah) menyala untuk mematikan motor. Jika proses *counting* selesai maka motor otomatis berhenti dan *buzzer* aktif, tekan SET untuk memulai *counting* lagi. Tekan BACK untuk kembali ke “Select Menu”.

## RIWAYAT HIDUP



Achmad Maulana, lahir pada tanggal 29 Juli 1994 dari pasangan Bapak Dede Basuki dan Ibu Juhariah sebagai anak kedua dari tiga bersaudara. Bertempat tinggal di Jalan G1 RT 02/02 No. 11D, Kecamatan Palmerah, Kelurahan Slipi, Jakarta Barat. Penulis menyelesaikan pendidikan formal dimulai dari Sekolah Dasar di SD Swasta Barunawati II pada tahun 2000 dan lulus pada tahun 2006. Kemudian melanjutkan ke Pendidikan Sekolah Menengah Pertama di SMP Negeri 88 Jakarta pada tahun 2006 dan lulus pada tahun 2009. Penulis meneruskan ke Pendidikan Sekolah Menengah Atas di SMA Negeri 65 Jakarta pada tahun 2009 dan lulus pada tahun 2012. Setelah tamat SMA penulis melanjutkan pendidikan ke Universitas Negeri Jakarta pada tahun 2012 untuk Program Studi Pendidikan Teknik Elektronika, Jurusan teknik Elektro, Fakultas Teknik, Universitas Negeri Jakarta.