

**PERENCANAAN DAN PEMBUATAN ALAT PENIMBANG
PERHIASAN EMAS DILENGKAPI DENGAN
PENCETAK SERTIFIKAT PERHIASAN**

**SKRIPSI
KONSENTRASI TEKNIK ELEKTRONIKA**

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik



Disusun oleh:
DWI JOKOWALOYO
NIM. 0810632005-63

**KEMENTERIAN PENDIDIKAN NASIONAL
UNIVERSITAS BRAWIJAYA
FAKULTAS TEKNIK
MALANG
2010**

LEMBAR PERSETUJUAN

PERENCANAAN DAN PEMBUATAN ALAT PENIMBANG PERHIASAN EMAS DILENGKAPI DENGAN PENCETAK SERTIFIKAT PERHIASAN

SKRIPSI KONSENTRASI TEKNIK ELEKTRONIKA

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik



Disusun oleh:

DWI JOKOWALOYO

NIM. 0810632005-63

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Ir. Nurussa'adah, MT.
NIP. 19680706 199203 2 001

Mochammad Rif'an, ST., MT.
NIP. 19710301 200012 1 001

LEMBAR PENGESAHAN

PERENCANAAN DAN PEMBUATAN ALAT PENIMBANG PERHIASAN EMAS DILENGKAPI DENGAN PENCETAK SERTIFIKAT PERHIASAN

SKRIPSI KONSENTRASI TEKNIK ELEKTRONIKA

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik

Disusun oleh:

DWI JOKOWALOYO

NIM. 0810632005-63

Sripsi ini telah diuji dan dinyatakan lulus pada
Tanggal 27 Desember 2010

Dosen Penguji

Ir. M. Julius St., MS.
NIP. 19540720 198203 1 002

Ir. Ponco Siwindarto, M.Eng.Sc.
NIP. 19590304 198903 1 001

Dr. Agung Darmawansyah, ST., MT.
NIP. 19721218 199903 1 002

Mengetahui,
Ketua Jurusan Teknik Elektro

Rudy Yuwono ST., M.Sc.
NIP. 19710615 199802 1 003

ABSTRAK

Dwi Jokowaloyo, 2010. *Perencanaan Dan Pembuatan Alat Penimbang Perhiasan Emas Dilengkapi Dengan Pencetak Sertifikat Perhiasan*. Skripsi, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Brawijaya.

Dosen Pembimbing : Ir. Nurussa'adah, MT. dan Mochammad Rif'an, ST., MT.

Timbangan perhiasan emas analog maupun digital yang terdapat di pasaran selama ini hanya menunjukkan besaran berat. Untuk itu dalam penelitian ini, dibuat timbangan perhiasan emas digital yang dapat menampilkan berat, menentukan harga dan dapat mencetak sertifikat perhiasan.

Dalam perancangan timbangan perhiasan emas ini digunakan sebuah rangkaian mikrokontroler AT89S52 yang berfungsi untuk mengatur dan mengolah data dari beberapa komponen seperti LCD (*Liquid Crystal Display*), ADC (*Analog to Digital Converter*), RTC (*Real Time Clock*), EEPROM (*Electrically Erasable and Programmable - Read Only Memory*) dan keypad. Data berat perhiasan emas diambil melalui sebuah sensor berat (*load cell*). Timbangan ini dilengkapi dengan sebuah keypad untuk memasukkan harga dan memilih jenis perhiasan emas, sebuah *printer* untuk mencetak sertifikat perhiasan.

Hasil dari penelitian ini berupa timbangan perhiasan emas digital yang dapat menampilkan data berat dalam satuan gram pada LCD, dapat menentukan harga perhiasan emas dan dapat memilih jenis perhiasan emas yang diinginkan, serta dapat mencetak sertifikat perhiasan. Nilai kesalahan rata-rata pengujian rangkaian pengkondisi sinyal sensor berat adalah 3,36% sedangkan nilai kesalahan rata-rata pengujian alat secara keseluruhan adalah 3,82%.

Kata kunci: Timbangan, Sertifikat perhiasan, mikrokontroler AT89S52.



PENGANTAR

Puji syukur kehadirat Allah SWT atas rahmat, hidayah, ridho, nikmat, dan karunia-Nya serta kekuatan-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul *“Perencanaan Dan Pembuatan Alat Penimbang Perhiasan Emas Dilengkapi Dengan Pencetak Sertifikat Perhiasan”* dengan lancar dan baik.

Penyusunan skripsi ini diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Teknik di jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya. Selama penyelesaian skripsi ini penulis menyadari adanya dorongan dan bantuan dari berbagai pihak, untuk itu penulis menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

- 1) Bapak Rudy Yuwono, ST., M.Sc. Ketua Jurusan Teknik Elektro,
- 2) Bapak M. Aziz Muslim, ST., MT., Ph.D. Sekretaris Jurusan Teknik Elektro,
- 3) Bapak Ir. Moch. Julius St., MT., Ketua Kelompok Dosen Keahlian Program Studi Paket B (Teknik Elektronika) Jurusan Teknik Elektro,
- 4) Ibu Ir. Nurussa'adah, MT. pembimbing I yang telah banyak memberikan pengarahan dan bantuan dalam menyelesaikan skripsi ini,
- 5) Bapak Moch. Rif'an, ST., MT. pembimbing II yang telah banyak memberikan masukan dan bantuan dalam menyelesaikan skripsi ini,
- 6) Orang tuaku tercinta, terima kasih atas doa, kasih sayang, perhatian, serta dukungan baik materi maupun non-materi yang tak ternilai yang telah diberikan,
- 7) Semua pihak yang tidak dapat penulis sebutkan satu per satu.

Penulis menyadari ketidaksempurnaan dalam penyusunan skripsi ini, sehingga penulis sangat mengharapkan kritik dan saran dari pembaca untuk dapat memperbaiki segala kekurangan di dalam skripsi ini. Akhirnya, semoga skripsi ini dapat memberikan manfaat dan berguna bagi pembaca.

Malang, 28 Desember 2010

Penulis



DAFTAR ISI

	halaman
HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN	iii
ABSTRAK	iv
PENGANTAR	v
DAFTAR ISI	vi
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
DAFTAR LAMPIRAN	xiii
BAB I PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah	2
1.4. Tujuan Penelitian	2
1.5. Sistematika Penulisan	2
BAB II TINJAUAN PUSTAKA	
2.1. <i>Load Cell</i>	4
2.1.1. <i>Strain Gauge</i>	5
2.1.2. <i>Jembatan Wheatstone</i>	5
2.2. <i>Op-Amp (Operational Amplifier)</i>	6
2.2.1. <i>Penguat Diferensial (Differential Amplifier)</i>	7
2.2.2. <i>Penguat Penyangga (Buffer Amplifier)</i>	8
2.2.3. <i>Penguat Instrumentasi (Instrumentation Amplifier)</i>	9
2.3. <i>ADC PCF8591</i>	10
2.4. <i>Protokol I²C</i>	11
2.5. <i>Mikrokontroler AT89S52</i>	12
2.5.1. <i>Konfigurasi Pin Mikrokontroler AT89S52</i>	13
2.5.2. <i>Arsitektur Mikrokontroler AT89S52</i>	15
2.5.3. <i>Pewaktu Mikrokontroler AT89S52</i>	15
2.5.4. <i>Reset</i>	16

2.6.	RTC DS1307.....	17
2.7.	EEPROM AT24C16	18
2.8.	LCD	19
2.9.	Keypad 4x4	22
2.10.	Printer LX - 800	22
	2.10.1. Sistem Komunikasi Paralel	23

BAB III METODOLOGI PENELITIAN

3.1.	Studi Literatur	26
3.2.	Penentuan Spesifikasi Alat	26
3.3.	Perencanaan Alat	26
3.4.	Pembuatan Alat	27
3.5.	Pengujian Alat.....	27

BAB IV PERENCANAAN DAN PEMBUATAN ALAT

4.1.	Spesifikasi Alat	28
4.2.	Perencanaan Sistem	28
4.3.	Prinsip Kerja Sistem	30
4.4.	Perancangan Perangkat Keras.....	30
	4.4.1. Perancangan Sensor <i>load cell</i>	31
	4.4.2. Perancangan Rangkaian Pengkondisi Sinyal	32
	4.4.3. Perancangan Rangkaian ADC.....	34
	4.4.4. Perancangan Rangkaian Mikrokontroler	36
	4.4.5. Perancangan Rangkaian RTC	39
	4.4.6. Perancangan Rangkaian EEPROM.....	40
	4.4.7. Perancangan Konfigurasi Pin SDA dan SCL.....	41
	4.4.8. Perancangan Antarmuka Modul LCD.....	43
	4.4.9. Perancangan Antarmuka Keypad.....	44
	4.4.10. Perancangan Antarmuka Port Paralel	45
4.5.	Perancangan Perangkat Lunak (<i>software</i>)	45

BAB V PENGUJIAN DAN PEMBAHASAN

5.1.	Pengujian Antarmuka Keypad	48
	5.1.1. Tujuan Pengujian	48
	5.1.2. Peralatan Pengujian.....	48
	5.1.3. Prosedur Pengujian	49

5.1.4.	Hasil Pengujian	49
5.2.	Pengujian Antarmuka Modul LCD.....	50
5.2.1.	Tujuan Pengujian	50
5.2.2.	Peralatan Pengujian.....	50
5.2.3.	Prosedur Pengujian	51
5.2.4.	Hasil Pengujian	51
5.3.	Pengujian Rangkaian RTC.....	51
5.3.1.	Tujuan Pengujian	51
5.3.2.	Peralatan Pengujian.....	52
5.3.3.	Prosedur Pengujian	52
5.3.4.	Hasil Pengujian	52
5.4.	Pengujian Rangkaian EEPROM.....	53
5.4.1.	Tujuan Pengujian	53
5.4.2.	Peralatan Pengujian.....	53
5.4.3.	Prosedur Pengujian	53
5.4.4.	Hasil Pengujian	54
5.5.	Pengujian Rangkaian ADC.....	54
5.5.1.	Tujuan Pengujian.....	54
5.5.2.	Peralatan Pengujian.....	54
5.5.3.	Prosedur Pengujian	55
5.5.4.	Hasil Pengujian	55
5.6.	Pengujian Rangkaian Pengkondisi Sinyal	56
5.6.1.	Tujuan Pengujian	56
5.6.2.	Peralatan Pengujian.....	56
5.6.3.	Prosedur Pengujian	56
5.6.4.	Hasil Pengujian	57
5.7.	Pengujian Antarmuka Port Paralel.....	58
5.7.1.	Tujuan Pengujian	58
5.7.2.	Peralatan Pengujian.....	58
5.7.3.	Prosedur Pengujian	58
5.7.4.	Hasil Pengujian	59
5.8.	Pengujian Sistem secara Keseluruhan	60
5.8.1.	Tujuan Pengujian	60
5.8.2.	Peralatan Pengujian.....	60

5.8.3. Prosedur Pengujian 60
5.8.4. Hasil Pengujian 61

BAB VI KESIMPULAN DAN SARAN

6.1. Kesimpulan 62
6.2. Saran 62

DAFTAR PUSTAKA

LAMPIRAN



DAFTAR TABEL

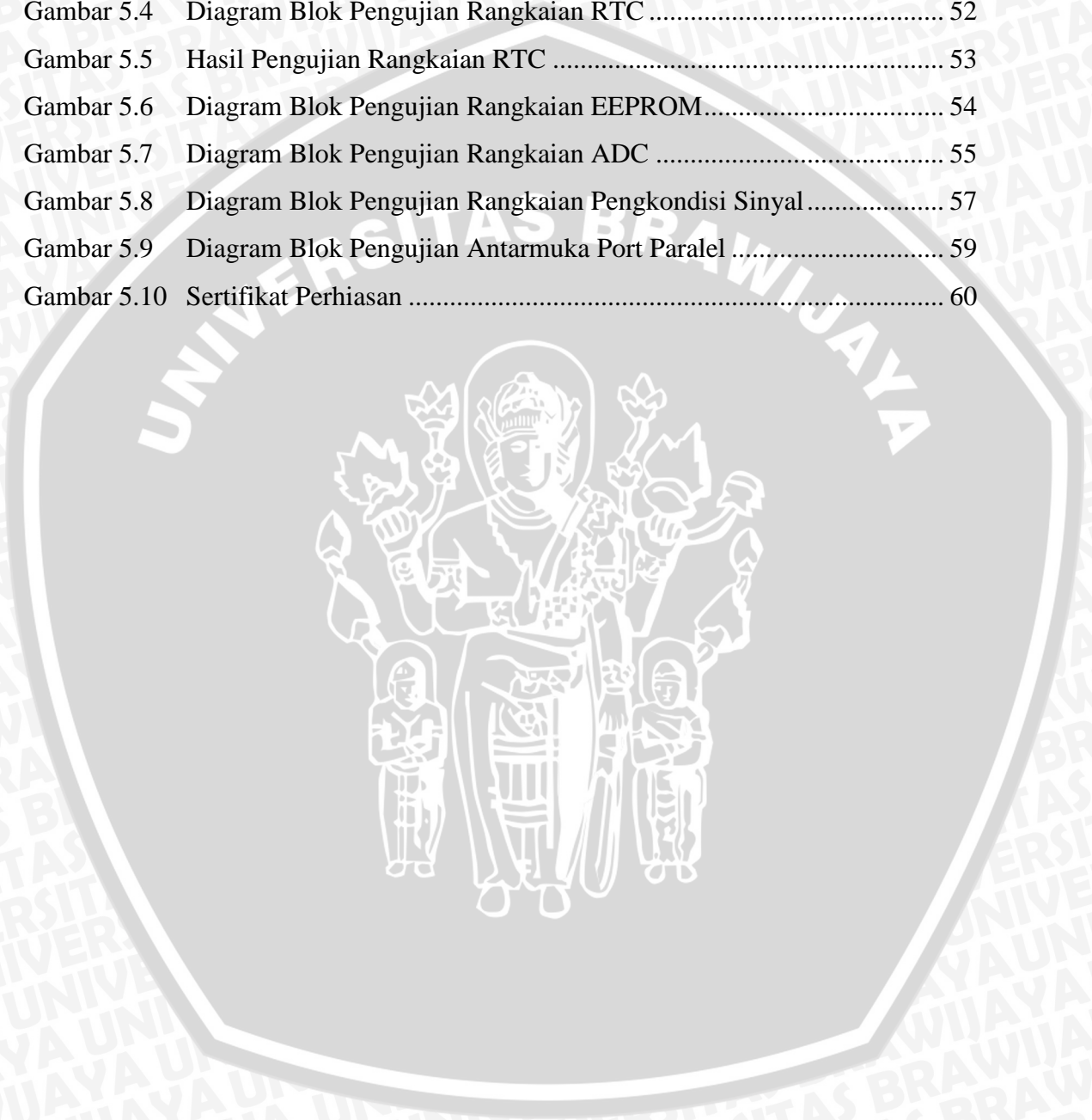
	Halaman
Tabel 2.1 Alamat Register RTC.....	18
Tabel 2.2 Fungsi pin-pin Modul LCD M1632.....	20
Tabel 2.3 Instruksi pada Modul LCD	21
Tabel 2.4 Nama Pin dari Konektor <i>Parallel Port</i> DB25 dan <i>Centronics</i>	25
Tabel 4.1 Kebutuhan Memori untuk Setiap Tansaksi.....	41
Tabel 5.1 Hasil Pengujian Antarmuka Keypad.....	50
Tabel 5.2 Hasil Pengujian Rangkaian RTC	53
Tabel 5.3 Data Hasil Pengujian Rangkaian EEPROM	54
Tabel 5.4 Hasil Pengujian Rangkaian ADC.....	55
Tabel 5.5 Hasil Pengujian Rangkaian Pengkondisi Sinyal Sensor Berat	57
Tabel 5.6 Hasil Pengujian Antarmuka Port Paralel	59
Tabel 5.7 Data Hasil Pengujian Timbangan Alat Secara Keseluruhan.....	61



DAFTAR GAMBAR

Gambar 2.1	<i>Load cell Bending Beam</i>	4
Gambar 2.2	Konfigurasi Susunan <i>Strain Gauge</i>	5
Gambar 2.3	Rangkaian Jembatan <i>Wheatstone</i>	6
Gambar 2.4	Simbol Penguat Operasional	7
Gambar 2.5	Rangkaian Penguat Diferensial	7
Gambar 2.6	Rangkaian Penguat Penyangga	9
Gambar 2.7	Konfigurasi Pin ADC PCF8591.....	10
Gambar 2.8	<i>Timing Diagram</i> Protokol I ² C	12
Gambar 2.9	Konfigurasi Pin Mikrokontroler AT89S52	13
Gambar 2.10	Arsitektur Mikrokontroler AT89S52	15
Gambar 2.11	Osilator Eksternal.....	16
Gambar 2.12	Rangkaian <i>Power On Reset</i>	16
Gambar 2.13	Konfigurasi Pin RTC DS1307	17
Gambar 2.14	Konfigurasi Pin EEPROM AT24C16	19
Gambar 2.15	Konfigurasi Pin LCD M1632.....	20
Gambar 2.16	Keypad 4×4	22
Gambar 2.17	Printer Epson LX-800	23
Gambar 2.18	Konfigurasi Pin Port Paralel.....	25
Gambar 4.1	Diagram Blok Keseluruhan Sistem	29
Gambar 4.2	Rangkaian Pengkondisi Sinyal.....	33
Gambar 4.3	Pengalamatan IC ADC PCF8591	34
Gambar 4.4	Rangkaian ADC PCF8591	35
Gambar 4.5	Rangkaian Mikrokontroler AT89S52	36
Gambar 4.6	Rangkaian Osilator.....	38
Gambar 4.7	Rangkaian <i>Reset</i>	38
Gambar 4.8	Rangkaian RTC DS1307	40
Gambar 4.9	Rangkaian EEPROM AT24C16	40
Gambar 4.10	Konfigurasi Pin SDA dan SCL	41
Gambar 4.11	Antarmuka Modul LCD	44
Gambar 4.12	Antarmuka Keypad	44

Gambar 4.13	Antarmuka Port Paralel	45
Gambar 4.14	Diagram Alir Program Utama	46
Gambar 5.1	Diagram Blok Pengujian Antarmuka Keypad	49
Gambar 5.2	Diagram Blok Pengujian Antarmuka Modul LCD	51
Gambar 5.3	Hasil Pengujian Antarmuka Modul LCD	51
Gambar 5.4	Diagram Blok Pengujian Rangkaian RTC	52
Gambar 5.5	Hasil Pengujian Rangkaian RTC	53
Gambar 5.6	Diagram Blok Pengujian Rangkaian EEPROM.....	54
Gambar 5.7	Diagram Blok Pengujian Rangkaian ADC	55
Gambar 5.8	Diagram Blok Pengujian Rangkaian Pengkondisi Sinyal	57
Gambar 5.9	Diagram Blok Pengujian Antarmuka Port Paralel	59
Gambar 5.10	Sertifikat Perhiasan	60



DAFTAR LAMPIRAN

LAMPIRAN I PERANGKAT KERAS

I-a Skema Rangkaian Keseluruhan

I-b Foto Alat

I-c Sertifikat Perhiasan

LAMPIRAN II PERANGKAT LUNAK

II-a *Flowchart* Pemilihan Jenis Perhiasan Emas

II-b *Flowchart* Kalibrasi

II-c *Flowchart* Entry Nama

II-d *Flowchart* Entry Nilai Tukar

II-e *Flowchart* Pengukuran

II-f *Listing* Program

LAMPIRAN III DATASHEET KOMPONEN

III-a *Datasheet* Load Cell

III-b *Datasheet* LM324

III-c *Datasheet* PCF8591

III-d *Datasheet* AT89S52

III-e *Datasheet* DS1307

III-f *Datasheet* AT24C16

III-g *Datasheet* LCD M1632

III-h *Datasheet* SMBus And I²C Bus Design

III-i *Datasheet* Parallel Port

BAB I

PENDAHULUAN

1.1. Latar Belakang

Perkembangan ilmu pengetahuan dan teknologi terutama dibidang elektronika membawa perubahan besar dengan memberikan berbagai macam alat atau sarana yang dapat mempermudah segala macam aktivitas manusia. Penekanan pada faktor kemudahan operasional serta kepraktisan membuat peralihan teknologi dari sistem analog ke sistem digital. Keadaan ini didasarkan pada kondisi kehidupan manusia yang ingin memenuhi kebutuhannya secara praktis untuk mendapatkan hasil yang maksimal.

Timbangan adalah sebuah alat bantu yang digunakan untuk mengetahui berat suatu benda. Dalam pemanfaatannya timbangan digunakan diberbagai bidang, dari bidang perdagangan, industri sampai dengan perusahaan jasa. Jenis timbangan yang digunakan bermacam-macam, mulai dari timbangan analog, timbangan mekanik hingga timbangan digital.

Timbangan digital mempunyai tingkat kepresisian yang lebih baik dan pengoperasian yang lebih efisien dari pada timbangan analog. Akan tetapi pemanfaatan kedua jenis timbangan ini hanya untuk mengukur besaran berat saja. Sedangkan proses pencatatan transaksi dan proses perhitungan biaya dilakukan secara manual dan relatif membutuhkan waktu yang lama. Hal ini memungkinkan terjadi kesalahan pencatatan transaksi dan kesalahan perhitungan harga.

Berdasarkan permasalahan tersebut, maka dirancang suatu alat yang mampu mengukur berat perhiasan emas yang dilengkapi dengan pencetak sertifikat perhiasan agar kemungkinan kesalahan dalam proses penimbangan, penulisan transaksi dan perhitungan biaya dapat dihindarkan. Perancangan dan pembuatan alat ini bertujuan untuk mempermudah para penjual dan pembeli perhiasan emas dalam proses penimbangan, penulisan transaksi dan perhitungan harga dan sebagai upaya untuk memberikan peningkatan pelayanan pada para konsumen.

1.2. Rumusan Masalah

Mengacu pada permasalahan yang diuraikan pada latar belakang, maka rumusan masalah dapat ditentukan pada:

- 1) Bagaimana merancang rangkaian pengkondisi sinyal sensor berat?,
- 2) Bagaimana mencetak sertifikat perhiasan?,
- 3) Berapakah nilai kesalahan dalam pengukuran menggunakan timbangan alat dibandingkan pengukuran dengan timbangan digital?

1.3. Batasan Masalah

Untuk mengarahkan pembahasan agar sesuai dengan tujuan, maka ditentukan ruang lingkup pembahasan sebagai berikut:

- 1) Penentuan besar nilai karat perhiasan emas digantikan dengan nilai tukar perhiasan emas,
- 2) Pada alat ini tidak disediakan fasilitas atau menu untuk menghitung laba/rugi penjualan perhiasan emas,
- 3) Data transaksi yang dapat disimpan maksimal 30 transaksi.

1.4. Tujuan Penelitian

Tujuan penelitian ini adalah mengimplementasikan rangkaian elektronika untuk mempermudah para penjual dan pembeli perhiasan emas dalam proses penimbangan dan perhitungan harga.

1.5. Sistematika Penulisan

Sistematika penulisan yang digunakan dalam penyusunan skripsi ini adalah sebagai berikut:

BAB I PENDAHULUAN

Menjelaskan tentang latar belakang masalah, rumusan masalah, pembatasan masalah, tujuan penelitian dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Menjelaskan teori dasar yang berisi penjelasan *Load cell*, Op-amp, ADC, Protokol I²C, Mikrokontroler, RTC, EEPROM, LCD, Keypad dan Printer.

BAB III METODOLOGI PENELITIAN

Menjelaskan tentang metodologi penelitian, spesifikasi alat, perencanaan dan pembuatan alat dan cara pengujian alat.

BAB IV PERANCANGAN DAN PEMBUATAN ALAT

Menjelaskan tentang diagram blok perancangan dan pembuatan alat yang meliputi prinsip kerja, spesifikasi alat, perancangan *hardware*, dan perancangan *software*.

BAB V PENGUJIAN DAN PEMBAHASAN

Menjelaskan tentang pengujian dan pembahasan yang meliputi pengujian tiap-tiap bagian dan pengujian secara keseluruhan.

BAB VI PENUTUP

Berisi kesimpulan dan saran.

UNIVERSITAS BRAWIJAYA



BAB II

TINJAUAN PUSTAKA

Penjelasan dan uraian teori penunjang yang digunakan dalam penelitian ini diperlukan untuk mempermudah pemahaman tentang cara kerja rangkaian maupun dasar-dasar perencanaan alat ini.

Teori-teori penunjang yang dijelaskan dalam bab ini meliputi:

- 1). *Load cell*,
- 2). Op-amp,
- 3). ADC (*Analog to Digital Converter*),
- 4). Protokol I²C,
- 5). Mikrokontroler,
- 6). RTC (*Real Time Clock*),
- 7). EEPROM (*Electrically Erasable and Programmable -ROM*)
- 8). LCD (*Liquid Crystal Display*),
- 9). Keypad,
- 10). Printer.

2.1. *Load cell*

Load cell sering digunakan pada setiap pengukuran berat secara elektronik. *Load cell* dikategorikan sebagai sensor gaya. Sensor ini berfungsi merubah besaran berat atau gaya menjadi sinyal listrik dalam bentuk tegangan. Bentuk fisik *load cell bending beam* ditunjukkan dalam Gambar 2.1



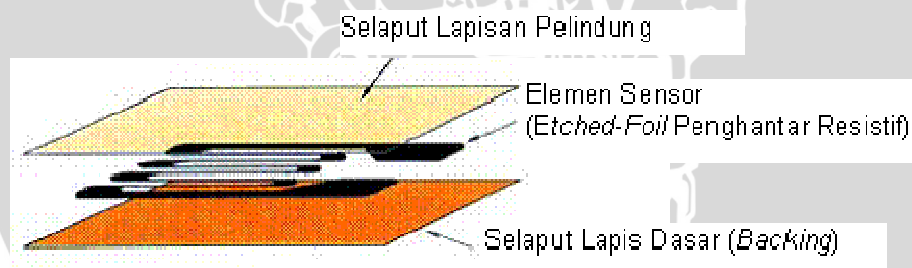
Gambar 2.1 *Load cell Bending Beam*
Sumber: www.vishaypgloadcell.com

Load cell sendiri dibentuk dari *tranduser* atau sensor tekan yang biasa disebut *strain gauge*. Sebuah *load cell* biasanya terdiri dari dua sampai empat *strain gauge* yang disusun menjadi jembatan *wheatstone*.

2.1.1. *Strain Gauge*

Gaya yang diberikan pada suatu benda logam (material ferrit / konduktif) selain menimbulkan deformasi bentuk fisik juga menimbulkan perubahan sifat resistansi elektrik benda tersebut. Dalam hal ini, dengan menempelkan jenis material tersebut pada suatu benda uji menggunakan suatu perekat yang isolatif terhadap arus listrik, maka material tersebut akan menghasilkan perubahan resistansi yang nilainya sebanding terhadap deformasi bentuknya. Karena adanya kesebandingan tersebut maka dapat dikatakan material ini sebagai sensor. Sensor/material inilah yang disebut *strain gauge*.

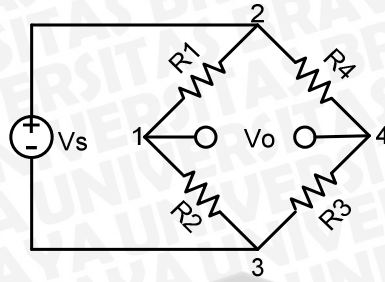
Prinsip kerja *strain gauge* adalah ketika terjadi regangan pada suatu benda uji yang telah dipasang *strain gauge*, maka regangan tersebut akan terhantarkan melalui alas *gauge* (isolatif) pada *foil* atau penghantar resistif di dalam *gauge* tersebut. Sehingga *foil* atau penghantar halus akan mengalami perubahan nilai resistansinya. Perubahan resistansi ini berbanding lurus terhadap besarnya regangan. *Strain gauge* dibuat dari sehelai kertas logam resistif yang dikikis tipis (*etched-foil*) dan berbentuk kisi (*grid*) – sebagai elemen utama (sensor) – serta dilapisi sepasang selaput sebagai pelindung sekaligus isolator. Kemudian ditambahkan sepasang kawat timah (*lead-gauge*) yang terhubung pada kedua ujung elemen sensor. Konfigurasi Susunan *Strain Gauge* ditunjukkan dalam Gambar 2.2.



Gambar 2.2 Konfigurasi Susunan *Strain Gauge*
Sumber: www.wikipedia.com

2.1.2. *Jembatan Wheatstone*

Rangkaian jembatan *wheatstone* digunakan untuk mengkonversi perubahan resistansi *strain gauge* menjadi tegangan atau arus. Jembatan *wheatstone* yang umum digunakan adalah jembatan *wheatstone* dengan satu elemen aktif, dua elemen aktif dan empat elemen aktif. Rangkaian jembatan *wheatstone* ditunjukkan dalam Gambar 2.3.



Gambar 2.3 Rangkaian Jembatan *Wheatstone*

Sumber: *Measuring Strain with Strain Gauge*, National Instrumens Corp

Jembatan dikatakan setimbang bila beda potensial pada V_o adalah 0 V. Kondisi ini terjadi bila tegangan dari node 1 ke node 2 sama dengan tegangan dari node 4 ke 2. Atau dengan mendasarkan pada terminal lainnya, jika tegangan dari node 1 ke 3 sama dengan tegangan dari node 4 ke 3. Jadi jembatan dalam keadaan setimbang jika:

$$I_1 R_1 = I_4 R_4 \quad (2-1)$$

$$I_1 = I_2 = \frac{V}{R_1 + R_2} \quad (2-2)$$

$$I_3 = I_4 = \frac{V}{R_3 + R_4} \quad (2-3)$$

Dengan mensubstitusikan Persamaan (2-2) dan (2-3) ke Persamaan (2-1) diperoleh:

$$\frac{V}{R_1 + R_2} R_1 = \frac{V}{R_3 + R_4} R_4$$

$$\frac{R_1}{R_1 + R_2} = \frac{R_4}{R_3 + R_4}$$

$$R_1 (R_3 + R_4) = R_4 (R_1 + R_2)$$

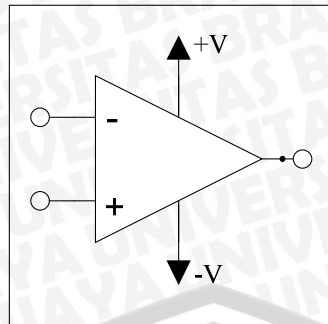
$$R_1 R_3 + R_1 R_4 = R_4 R_1 + R_4 R_2$$

$$R_1 R_3 = R_4 R_2 \text{ atau } \frac{R_1}{R_2} = \frac{R_4}{R_3} \quad (2-4)$$

Persamaan (2-4) merupakan bentuk yang telah dikenal dalam kesetimbangan jembatan *wheatstone*.

2.2. Op-amp (Operational Amplifier)

Rangkaian op-amp atau yang sering dikenal dengan nama operational *amplifier* merupakan penguat DC dengan *gain* yang tinggi (Coughlin, 1992: 10). Pada op-amp terdapat satu terminal keluaran dan dua terminal masukan. Terminal masukan yang bertanda (-) dinamakan terminal masukan pembalik (*inverting*) sedangkan terminal masukan yang bertanda (+) dinamakan terminal masukan tak pembalik (*non inverting*). Simbol op-amp ditunjukkan dalam Gambar 2.4.



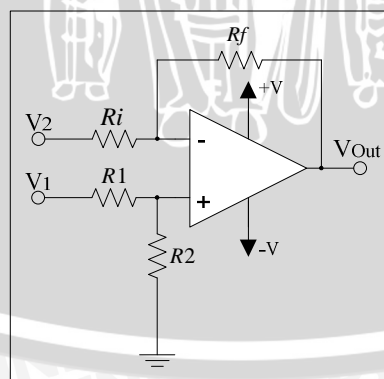
Gambar 2.4 Simbol Penguat Operasional
Sumber: Coughlin, 1992:13

Op-amp yang ideal mempunyai sifat-sifat dasar sebagai berikut:

- Resistansi masukan $R_i = \infty$
- Resistansi keluaran $R_o = 0$
- Perolehan tegangan $A_v = -\infty$
- Lebar pita = ∞
- $V_o = 0$, jika $V_+ = V_-$, tidak tergantung pada besarnya V_+

2.2.1. Penguat Diferensial (*Differential Amplifier*)

Penguat yang paling berguna untuk pengukuran atau pengendalian adalah penguat diferensial. Penguat diferensial disusun oleh sebuah op-amp dan beberapa tahanan presisi, yang membuat rangkaiannya sangat stabil dan berguna apabila ketelitian merupakan hal yang penting. Penguat diferensial bisa untuk mengukur maupun memperkuat sinyal-sinyal kecil yang terpendam menjadi sinyal yang jauh lebih besar. Rangkaian penguat diferensial ditunjukkan dalam Gambar 2.5.



Gambar 2.5 Rangkaian Penguat Diferensial
Sumber: Robert F. Coughlin, 1992: 162

Dalam Gambar 2.5, untuk mengetahui besar V_o , digunakan teorema superposisi, V_{out1} merupakan tegangan keluaran op-amp jika $V_1 = 0$ dan V_{out2} merupakan tegangan keluaran op-amp jika $V_2 = 0$

$$V_O = V_{out1} + V_{out2} \quad (2-5)$$

Jika $V_1 = 0$, maka rangkaian menjadi penguat pembalik, sehingga:

$$V_{out1} = -\frac{R_f}{R_i} V_2 \quad (2-6)$$

Namun, bila $V_2 = 0$, maka:

$$V_{out} = \left(\frac{R_2}{R_1 + R_2} \right) V_1 \quad (2-7)$$

Sehingga V_{out2} adalah

$$V_{out2} = \left(\frac{R_2}{R_1 + R_2} \right) \left(\frac{R_f + R_i}{R_i} \right) V_1 \quad (2-8)$$

Kembali ke Persamaan (2-5), maka V_o adalah

$$V_{out} = V_{out1} + V_{out2} \\ V_{out} = \left\{ -\frac{R_f}{R_i} V_2 \right\} + \left\{ \left(\frac{R_2}{R_1 + R_2} \right) \left(\frac{R_f + R_i}{R_i} \right) V_1 \right\} \quad (2-9)$$

Jika $\frac{R_f}{R_i} = \frac{R_2}{R_1}$, maka:

$$\frac{R_f + R_i}{R_i} = \frac{R_i}{R_i} + \frac{R_f}{R_i}$$

$$\frac{R_f + R_i}{R_i} = 1 + \frac{R_f}{R_i}$$

Substitusikan $\frac{R_2}{R_1}$ ke $\frac{R_f}{R_i}$, sehingga

$$\frac{R_f + R_i}{R_i} = 1 + \frac{R_2}{R_1}$$

$$\frac{R_i}{R_i} + \frac{R_f}{R_i} = \frac{R_1}{R_1} + \frac{R_2}{R_1}$$

Maka Persamaan (2-9) menjadi

$$V_{out} = \left\{ -\frac{R_f}{R_i} V_2 \right\} + \left\{ \left(\frac{R_2}{R_1 + R_2} \right) \left(\frac{R_f + R_i}{R_i} \right) V_1 \right\} \\ V_{out} = \left\{ -\frac{R_f}{R_i} V_2 \right\} + \left\{ \left(\frac{R_2}{R_1 + R_2} \right) \left(\frac{R_2 + R_1}{R_1} \right) V_1 \right\} \\ V_o = \frac{R_f}{R_i} (V_1 - V_2) \quad (2-10)$$

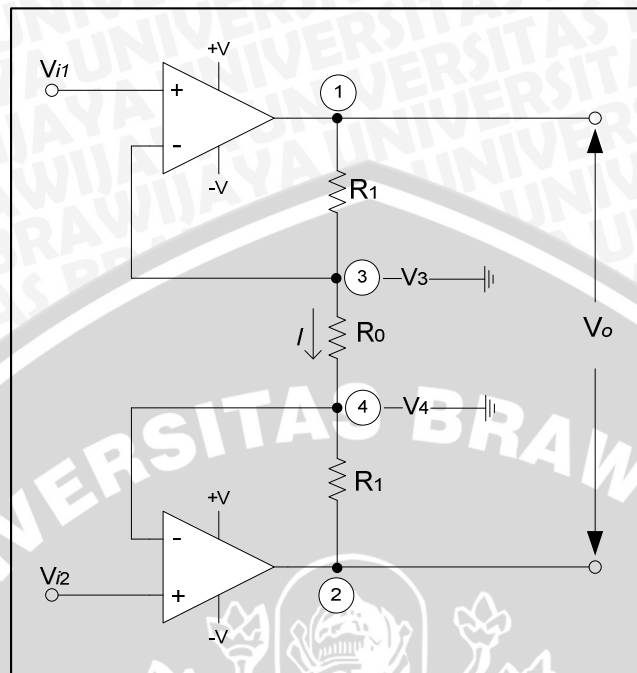
Sehingga penguatan diferensial (A_d) adalah:

$$A_d = \frac{R_f}{R_i} \quad (2-11)$$

2.2.2. Penguat Penyangga (*Buffer Amplifier*)

Rangkaian penguat penyangga (*buffer amplifier*) adalah rangkaian yang masukannya sama dengan hasil keluarannya. Rangkaian penguat penyangga ditunjukkan dalam Gambar 2.6. Pada rangkaian penguat penyangga terdapat sebuah

tahanan variabel, karena masing-masing tegangan di titik 3 dan 4 sama dengan V_{i1} dan V_{i2} terhadap ground ($V_3=V_{i1}$ dan $V_4=V_{i2}$).



Gambar 2.6 Rangkaian Penguat Penyangga
Sumber: Robert F. Coughlin, 1992: 167

Tegangan pada R_0 adalah $V_{i1} - V_{i2}$, bila $R_1=R_2$ maka resistor variabel R_0 digunakan untuk pengaturan penguatan (*gain*). Arus yang mengalir pada R_0 sebesar:

$$I = \frac{V_{i1} - V_{i2}}{R_0} \quad (2-12)$$

Karena I merupakan arus yang melalui R_1 dan R_2 , maka $I = i_1 = i_2$

$$\frac{V_3 - V_4}{R_0} = \frac{V_1 - V_2}{R_1 + R_0 + R_2} \quad (2-13)$$

Maka tegangan keluaran V_{out} adalah

$$(V_1 - V_2) = \frac{R_1 + R_0 + R_2}{R_0} (V_{i1} - V_{i2}) \quad (2-14)$$

Sehingga penguatan tegangan rangkaian penguat penyangga (A_b) adalah

$$A_b = \frac{(V_1 - V_2)}{(V_{i1} - V_{i2})} = \left(\frac{R_1 + R_0 + R_2}{R_0} \right) \quad (2-15)$$

2.2.3. Penguat Instrumentasi (*Instrumentation Amplifier*)

Penguat instrumentasi merupakan penguat yang digunakan untuk memperkuat tegangan dalam orde kecil (mV) dan mempunyai impedansi masukan yang tinggi. Penguat instrumentasi terdiri dari penguat penyangga dan penguat diferensial. Persamaan penguat instrumentasi sebagai berikut (sumber: Frank R. Dungan, 1991: 78):

$$A = \left(1 + \frac{2R_1}{R_0}\right) \left(\frac{R_f}{R_i}\right) \quad (2-16)$$

Sehingga tegangan keluaran (V_{out}) penguat instrumentasi adalah

$$V_{out} = Av(V_1 - V_2) \quad (2-17)$$

Sedangkan CMRR (*Common Mode Rejection Ratio*) penguat instrumentasi adalah

$$CMRR = \frac{A_d}{A_c}$$

Keterangan:

A_d = Penguatan mode diferensial

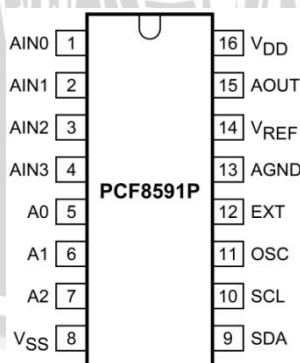
A_c = Penguatan mode bersama

$$CMRR(\text{dB}) = 20 \log |CMRR| \quad (2-18)$$

2.3. ADC PCF8591

ADC merupakan sebuah piranti yang dirancang dalam satu *chip* untuk mengubah sinyal analog menjadi sinyal digital. Penggunaan ADC adalah untuk mendapatkan data digital hasil konversi sinyal analog dari sensor berat. Resolusi ADC menyatakan jumlah nilai diskrit yang bisa dihasilkan dari *range* tegangan tertentu, yang dinyatakan dalam bit. Misalnya ADC yang mengubah masukan analog menjadi satu dari nilai diskrit antara 0-255 memiliki resolusi sebesar 8 bit. Resolusi tegangan ADC merupakan hasil bagi antara seluruh nilai *range* tegangan dengan jumlah nilai diskrit.

Dalam penelitian ini diperlukan pengubah sinyal analog ke sinyal digital (ADC). Sinyal analog dari keluaran pengkondisi sinyal akan diubah menjadi sinyal digital oleh ADC PCF8591. ADC jenis ini memiliki satu keluaran tegangan analog dengan perangkat bus serial I²C. Konfigurasi pin ADC PCF8591 ditunjukkan dalam Gambar 2.7.



Gambar 2.7 Konfigurasi Pin ADC PCF8591

Sumber: Philips, 2003: 4

IC ADC PCF8591 merupakan ADC yang menggunakan metode pendekatan berturut-turut (*successive-approximation* ADC), yang prinsip kerjanya yaitu membandingkan nilai tegangan masukan dengan keluaran DAC internal sampai

mencapai nilai pendekatan terbaik. Pada setiap step proses, nilai biner dari hasil pendekatan disimpan pada SAR (*successive-approximation register*). SAR menggunakan tegangan referensi sebagai perbandingan. Metode ini memerlukan waktu konversi satu *clock-cycle* untuk setiap bit. Alamat, control dan data dari dan ke mikrokontroler dikirim secara serial melalui dua kabel secara *bidirectional I²C*.

Nilai pembacaan dari ADC setelah pengkonversian dapat ditentukan dengan Persamaan (2-19), dan hasil ini tergantung dari nilai tegangan masukan ADC.

$$\text{Nilai ADC} = \frac{V_{in}}{V_{ref}} \cdot 2^n \quad (2-19)$$

Keterangan:

V_{in} : Tegangan masukan (volt)

V_{ref} : Tegangan referensi (volt)

n : Jumlah bit

Kenaikan tiap langkah untuk kenaikan keluaran biner (resolusi tegangan ADC) ditentukan dengan Persamaan (2-20), yaitu sebagai berikut:

$$V_{step} = \frac{V_{ref}}{2^n} \quad (2-20)$$

Keterangan:

V_{ref} : Tegangan referensi (volt)

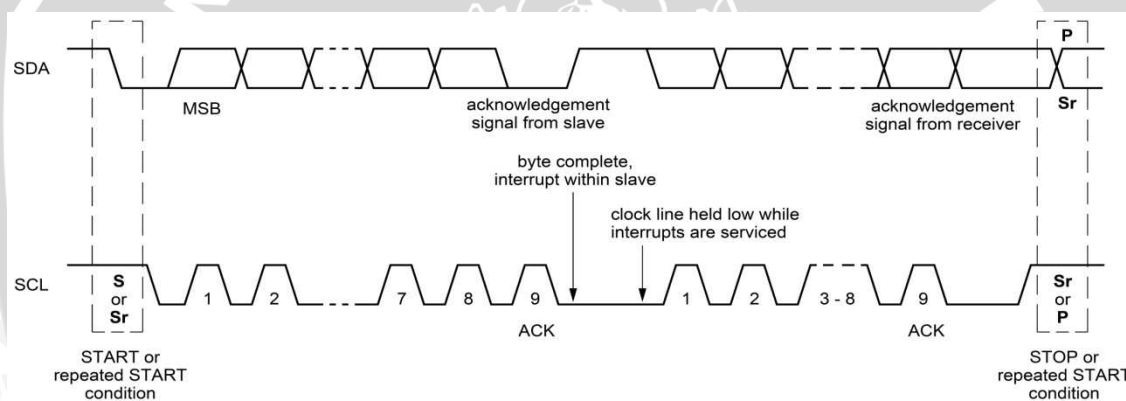
n : Jumlah bit

2.4. Protokol I²C

Komunikasi data secara I²C dilakukan melalui dua saluran, masing-masing adalah saluran data secara seri (SDA) dan saluran *clock* secara seri (SCL). Kedua saluran ini dikenal sebagai bus I²C yang digunakan untuk menghubungkan banyak IC I²C untuk berbagai macam keperluan. IC I²C dibedakan menjadi *master* dan *slave*, yang dimaksud dengan *master* adalah peralatan I²C yang memulai transfer data dan yang membangkitkan *clock* (SCK). Yang bertindak sebagai *master* umumnya mikroprosesor/mikrokontroler yang bertugas mengendalikan I²C bus.

Pada kondisi normal, SDA bisa berubah selama SCL dalam kondisi *low*, pengecualian saat kondisi *start* dan *stop*. Kondisi *start* menandakan *master* akan memulai proses pertukaran data dengan ditandai transisi *high* ke *low* pada saluran SDA saat saluran SCL *high*. Sedangkan kondisi *stop* menandakan *master* akan menghentikan proses pertukaran data dengan ditandai transisi *low* ke *high* pada saluran SDA saat saluran SCL *high*.

Lebar data yang dikirimkan melalui SDA harus sebanyak 8 bit atau satu byte. Banyaknya byte yang dapat dikirimkan tidak dibatasi. Masing-masing byte harus diikuti bit *acknowledge* oleh pihak penerima data dengan menahan saluran SDA pada kondisi *low* saat pulsa *acknowledge* sedang dalam periode *high*. Jika pihak penerima tidak memberikan *acknowledge*, berarti pihak penerima tidak sedang dalam kondisi siap bertukar data. Data ditransfer dengan MSB terlebih dahulu. Jika *slave* tidak dapat menerima atau mengirimkan byte data selama melakukan beberapa fungsi lain. Misalnya melayani interupsi internal, *slave* dapat memaksa *master* ke kondisi *wait* dengan menahan saluran SCL pada kondisi *low*. Perpindahan data kemudian dilanjutkan jika *slave* siap untuk bertukar data dan melepaskan saluran SCL. Proses pengiriman data dapat dihentikan dengan memberikan kondisi *stop* walaupun proses pengiriman belum lengkap satu byte. Dalam kasus ini, tidak ada sinyal *acknowledge*. *Timing diagram* protokol I²C ditunjukkan dalam Gambar 2.8.



Gambar 2.8 *Timing Diagram* Protokol I²C
Sumber: Philips, 2000: 10

2.5. Mikrokontroler AT89S52

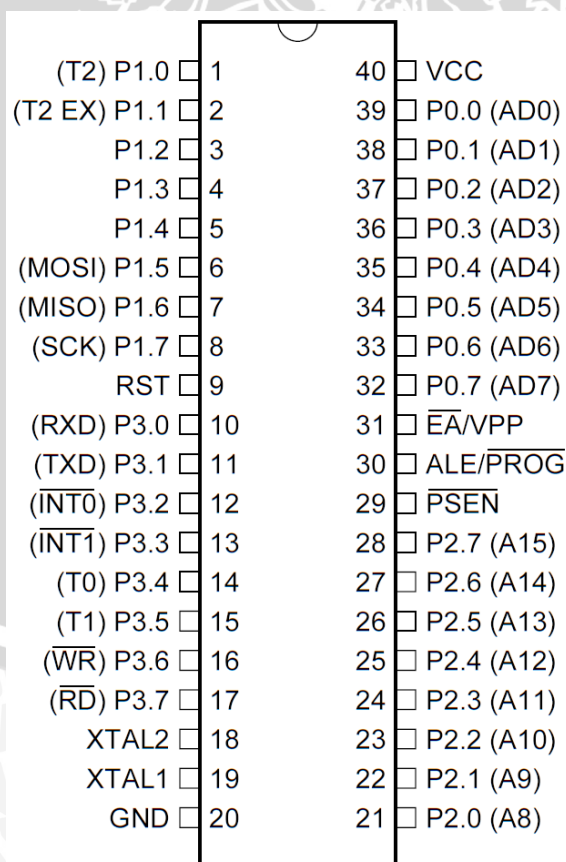
AT89S52 merupakan mikrokontroler produk dari Atmel dengan 8k byte flash PEROM (*Programmable and Erasable Read Only Memory*). AT89S52 mempunyai memori dengan teknologi *nonvolatile memory*, isi memori tersebut dapat diisi ulang atau dihapus berkali-kali. Memori ini digunakan untuk menyimpan instruksi berstandar MCS-51 *code* sehingga memungkinkan mikrokontroler untuk bekerja dalam mode *single chip operation* yang tidak memerlukan *external memory* untuk menyimpan *source code* tersebut. Secara umum *feature* yang dimiliki mikrokontroler AT89S52 adalah sebagai berikut:

- Sebuah CPU 8 bit dengan menggunakan teknologi dari Atmel,
- Memiliki memori baca-tulis (RAM) sebesar 256 byte,

- ☑ Memiliki 32 saluran masukan/keluaran,
- ☑ Jalur dua arah (*bidirectional*) yang digunakan sebagai saluran masukan atau keluaran,
- ☑ Sebuah port serial dengan kontrol *full duplex* UART (*Universal Asynchronous Receiver Transmitter*),
- ☑ Tiga buah *timer/counter* 16 bit,
- ☑ Osilator internal dan rangkaian pewaktu,
- ☑ Flash EPROM yang besarnya 8 kbyte untuk memori program,
- ☑ Kemampuan melaksanakan operasi perkalian, pembagian dan operasi *Boolean*,
- ☑ Mampu beroperasi 0 sampai 33 MHz.

2.5.1. Konfigurasi Pin Mikrokontroler AT89S52

Konfigurasi pin mikrokontroler AT89S52 digolongkan menjadi sumber tegangan, pin isolator, pin kontrol, pin masukan/keluaran dan pin untuk interupsi luar. Konfigurasi pin mikrokontroler AT89S52 ditunjukkan dalam Gambar 2.9.



Gambar 2.9 Konfigurasi Pin Mikrokontroler AT89S52

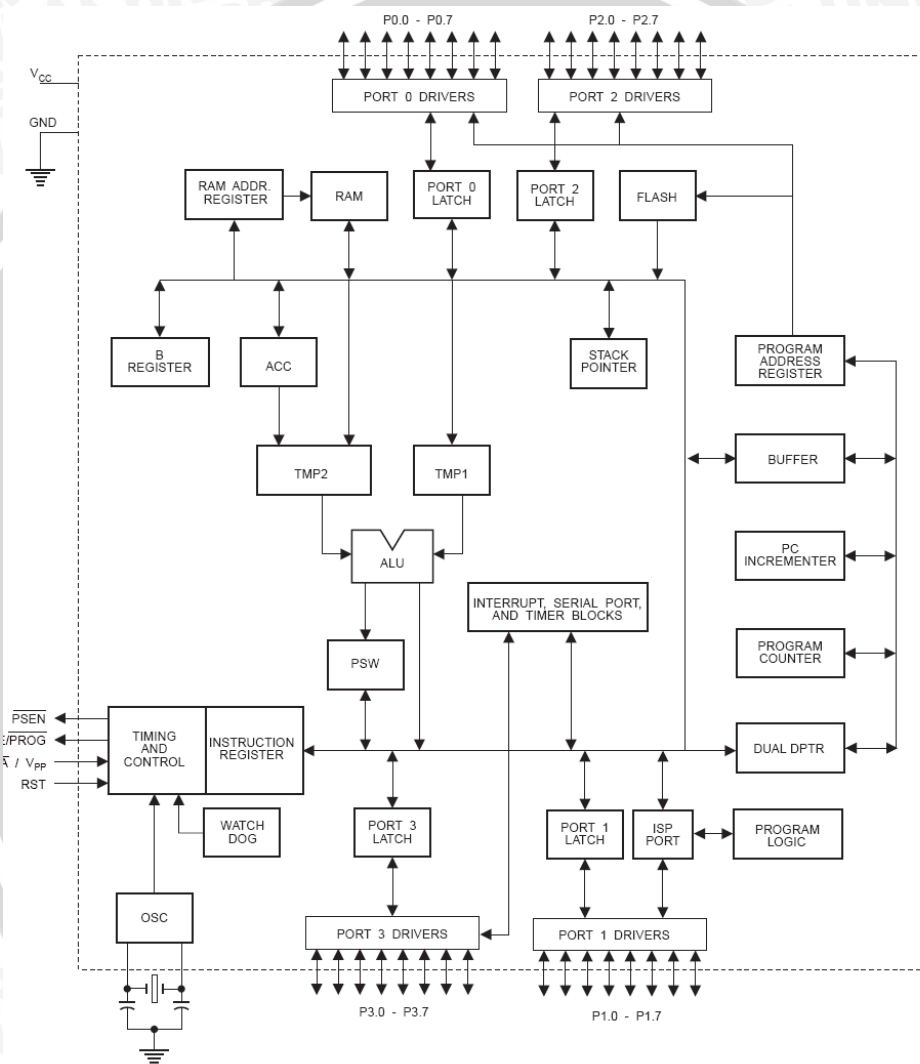
Sumber: Atmel, 2001: 2

Fungsi pin-pin mikrokontroler AT89S52 adalah sebagai berikut:

- ☑ Port 1 (Pin 1..8), merupakan saluran masukan/keluaran dua arah. Sebagai tambahan P1.0 dan P1.1 dapat digunakan sebagai T2 (*Timer 2*) dan T2EX (*Timer 2 external*),
- ☑ Pin 9 RST, merupakan saluran dua masukan untuk mereset mikrokontroler dengan cara memberi masukan logika tinggi,
- ☑ Port 3 (Pin 10..17), merupakan saluran masukan/keluaran dua arah dan mempunyai fungsi khusus. Fungsi khusus meliputi TXD (*Transmit Data*), RXD (*Receive Data*), $\overline{\text{INT}}0$ (*Interrupt 0*), $\overline{\text{INT}}1$ (*Interrupt 1*), T0 (*Timer 0*), T1 (*Timer 1*), $\overline{\text{WR}}$ (*Write*), $\overline{\text{RD}}$ (*Read*),
- ☑ Pin 18 dan 19 (XTAL₁ dan XTAL₂), merupakan saluran untuk mengatur pewaktuan sistem. Untuk pewaktuan dapat menggunakan pewaktuan internal maupun eksternal,
- ☑ Pin 20 V_{SS}, merupakan hubungan ke *ground* dari rangkaian,
- ☑ Port 2 (Pin 21..28), merupakan saluran masukan/keluaran dua arah,
- ☑ Pin 29 $\overline{\text{PSEN}}$ (*Program Store Enable*), merupakan sinyal baca untuk mengaktifkan memori program eksternal,
- ☑ Pin 30 ALE / $\overline{\text{PROG}}$ (*Address Latch Enable*), merupakan pulsa yang berfungsi untuk menahan alamat rendah (A0-A7) dalam port 0, selama proses baca/tulis memori eksternal. Frekuensi ALE adalah $\frac{1}{6}$ kali frekuensi osilator dan dapat digunakan sebagai pewaktu. Pin ini juga berfungsi sebagai saluran program selama dilakukan pemrograman jika menggunakan memori program internal,
- ☑ Pin 31 $\overline{\text{EA}}$ / VPP (*External Access Enable*), untuk mengatur penggunaan memori program eksternal dan internal. Pin ini harus dihubungkan dengan *ground* bila menggunakan memori program eksternal dan dihubungkan dengan VCC bila menggunakan memori program internal,
- ☑ Port 0 (Pin 32..39), merupakan saluran masukan/keluaran *open drain*,
- ☑ Pin 40 VCC, merupakan saluran masukan untuk catu daya positif sebesar 5 volt DC.

2.5.2. Arsitektur Mikrokontroler AT89S52

Mikrokontroler AT89S52 terdiri dari sebuah CPU (*central processing unit*), dua jenis memori yaitu memori data (RAM) dan memori program (ROM), port masukan/keluaran dengan *programmable* pin secara *independent* dan register-register mode, status, internal *timer* dan *counter*, *serial communication* serta logika *random* yang diperlukan oleh berbagai fungsi *peripheral*. Arsitektur mikrokontroler AT89S52 ditunjukkan dalam Gambar 2.10.

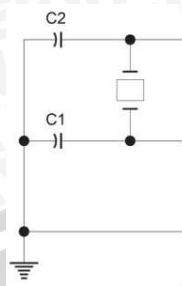


Gambar 2.10 Arsitektur Mikrokontroler AT89S52
Sumber: Atmel, 2001: 3

2.5.3. Pewaktu Mikrokontroler AT89S52

Mikrokontroler AT89S52 memiliki osilator internal (*on chip oscillator*) yang dapat digunakan sebagai sumber *clock* bagi MCU. Untuk menggunakan osilator internal diperlukan sebuah kristal antara pin XTAL 1 dan pin XTAL 2 serta dua buah kapasitor ke *ground*. Komponen kristal dapat digunakan frekuensi dari 0 sampai 33

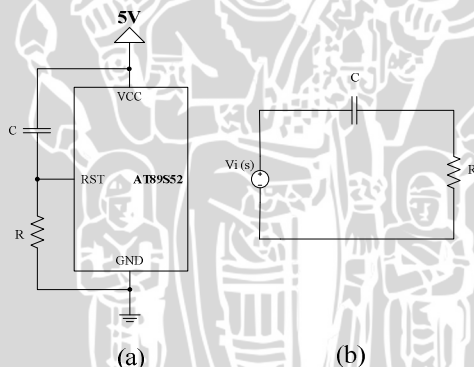
MHz, sedangkan untuk kapasitor dapat bernilai $30 \text{ pF} \pm 10 \text{ pF}$. Rangkaian osilator eksternal ditunjukkan dalam Gambar 2.11.



Gambar 2.11 Osilator Eksternal
Sumber: Atmel, 2001 : 15

2.5.4. Reset

Rangkaian *power on reset* diperlukan untuk mereset mikrokontroler secara otomatis setiap catu daya dinyalakan. Ketika catu daya diaktifkan, rangkaian *power on reset* menahan logika tinggi pin RST dengan jangka waktu yang ditentukan oleh lamanya pengisian muatan C, yang nilai jangka waktunya dihitung dalam Persamaan (2-21). Rangkaian *power on reset* ditunjukkan dalam Gambar 2.12 (a).



Gambar 2.12 (a) Rangkaian *power on reset* (b) Rangkaian ekivalen *power on reset*
Sumber: Atmel, 2001 : 4

Rangkaian ekivalen ditunjukkan dalam Gambar 2.12 (b), didapatkan bahwa:

$$V_o(s) = \frac{R}{R + \frac{1}{Cs}} \cdot V_i(s) = \frac{RCs}{RCs + 1} \cdot V_i(s) \quad (2-21)$$

Keterangan:

V_o = Tegangan keluaran (V)

V_i = Tegangan masukan (V)

R = Tahanan resistor (Ω)

C = Tahanan kapasitor (F)

Dengan tegangan V_i adalah tegangan VCC yaitu 5V, dalam fungsi Laplace adalah $5/s$, sehingga:

$$V_o = \frac{RCs}{RCs+1} \cdot \frac{5}{s} = \frac{RC}{RCs+1} \cdot 5 = 5 \cdot \left(\frac{1}{s + \frac{1}{RC}} \right)$$

$$V_o = 5 \cdot e^{-\left(\frac{t}{RC}\right)} \quad (2-22)$$

$$\frac{5}{V_o(t)} = e^{\frac{t}{RC}} \rightarrow \ln\left(\frac{5}{V_o}\right) = \frac{t}{RC}$$

maka:

$$t = R \cdot C \cdot \left(\ln \frac{5}{V_o} \right), \quad \text{atau} \quad t = R \cdot C \cdot \frac{\left(\log \frac{5}{V_o} \right)}{\log e} \quad (2-23)$$

Dengan nilai V_o adalah tegangan logika nominal yang diijinkan oleh pin RST (Atmel, 2001: 23), dimana:

$$V_o = 0,7 \cdot V_{CC} = 0,7 \times 5 = 3,5 \text{ volt.}$$

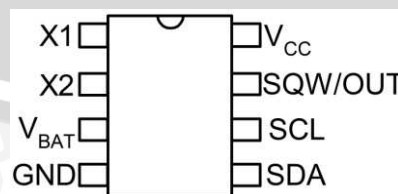
maka:

$$t = R \cdot C \cdot \frac{\left(\log \frac{5}{3,5} \right)}{\log e}, \text{ sehingga}$$

$$t = 0,357 \cdot R \cdot C \quad (2-24)$$

2.6. RTC DS1307

DS1307 merupakan sebuah IC RTC (*Real Time Clock*) yang dapat merekam dan memberikan informasi waktu secara lengkap mulai dari informasi detik, menit, jam, tanggal, bulan, hingga tahun. Konfigurasi pin RTC DS 1307 ditunjukkan dalam Gambar 2.13.



Gambar 2.13 Konfigurasi Pin RTC DS1307
Sumber: Maxim, 2008: 1

Berbagai fitur yang disediakan IC ini antara lain:

- ☑ Perhitungan RTC mulai dari detik, menit, jam, tanggal, bulan, hari dalam seminggu dan tahun,
- ☑ RAM internal sebesar 56 byte,
- ☑ Antarmuka serial I²C,
- ☑ Sinyal keluaran dalam bentuk gelombang kotak terprogram,
- ☑ Konsumsi daya kurang dari 500 nA menggunakan mode baterai cadangan dengan operasional osilator.

Informasi penanggalan dan waktu diperoleh dengan pembacaan byte register yang sesuai. Alamat register RTC ditunjukkan dalam Tabel 2.1. Penanggalan dan waktu di *set* dengan penulisan *byte register* yang sesuai. Isi dari register penanggalan dan waktu adalah dalam format *binary-coded decimal* (BCD).

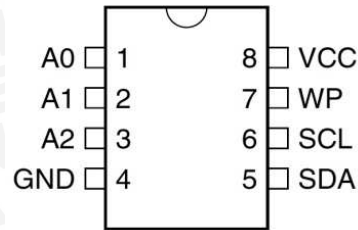
Tabel 2.1 Alamat Register RTC

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	FUNCTION	RANGE
00h	CH	10 Seconds			Seconds			Seconds	00–59	
01h	0	10 Minutes			Minutes			Minutes	00–59	
02h	0	12	10 Hour	10 Hour	Hours			Hours	1–12 +AM/PM 00–23	
		24	PM/AM							
03h	0	0	0	0	0	DAY		Day	01–07	
04h	0	0	10 Date		Date		Date	Date	01–31	
05h	0	0	0	10 Month	Month			Month	01–12	
06h	10 Year			Year			Year	Year	00–99	
07h	OUT	0	0	SQWE	0	0	RS1	RS0	Control	—
08h–3Fh									RAM 56 x 8	00h–FFh

Sumber: Maxim, 2008: 8

2.7. EEPROM AT24C16

EEPROM (*Electrically Erasable and Programmable Read Only Memory*) merupakan memori jenis *nonvolatile* sehingga isinya tidak mudah hilang walaupun catu daya dimatikan. Data dapat dihapus secara elektrik menggunakan tegangan listrik. Penghapusan EEPROM merupakan kata lain dari penulisan data baru yang diinginkan. Jadi dalam memprogram tidak perlu dilakukan penghapusan, tetapi data yang baru ditulis diatas data yang lama. Hal tersebut dilakukan dengan catu daya sebesar 5 volt. Dengan demikian sistem baca atau tulis memori EEPROM dapat diperlakukan seperti baca tulis pada RAM. Konfigurasi pin IC AT24C16 ditunjukkan dalam Gambar 2.14.



Gambar 2.14 Konfigurasi Pin EEPROM AT24C16

Sumber: Atmel, 2001: 1

Berbagai fitur yang disediakan IC EEPROM AT24C16 antara lain:

- Bekerja pada tegangan rendah,
- Memiliki kapasitas 16 kBit / 16384 bit (2048 karakter(byte)×8),
- Transfer data dapat dilakukan dengan dua arah (*bidirectional*),
- Memiliki pin WP (*Write Protection*) untuk melindungi data yang tersimpan dalam memori,
- Memiliki daya tahan penulisan sampai 1 juta kali penulisan,
- Kemampuan menyimpan data sampai 100 tahun.

2.8. LCD

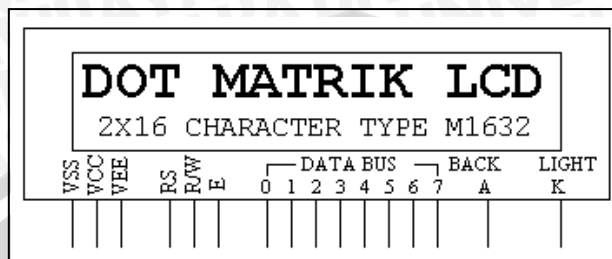
Modul LCD (*Liquid Crystal Display*) adalah salah satu jenis penampil yang digunakan untuk menampilkan angka, karakter atau bahkan angka dan karakter. Modul LCD terdiri atas tumpukan tipis atau sel dari dua lembar kaca dengan pinggirannya yang tertutup rapat. Di antara dua lembar kaca tersebut diberi bahan kristal cair (*liquid crystal*) yang tembus cahaya dimana kaca tersebut akan beremulsi apabila diberi tegangan. Bus data modul LCD terhubung dengan bus data mikrokontroler.

LCD pada alat ini digunakan untuk menampilkan informasi pilihan menu, informasi berat benda dalam satuan gram serta informasi tanggal penimbangan. LCD yang digunakan bertipe M1632. Spesifikasi LCD ini adalah sebagai berikut:

- Menampilkan 16 karakter pada tiap baris TN LCD dengan 5×7 dot matrik,
- Pembangkit karakter ROM untuk 192 jenis karakter,
- Pembangkit karakter RAM untuk 8 jenis karakter,
- Data RAM 80×8 bit,
- Tegangan catu 5 volt dan temperatur operasi 0-50 °C,
- Otomatis *reset* pada saat dihidupkan.

Liquid adalah modul tampilan yang mempunyai konsumsi daya yang relatif rendah dan terdapat sebuah modul *controller* CMOS didalamnya. *Controller* tersebut sebagai pembangkit karakter ROM/RAM dan tampilan data RAM. Semua fungsi

tampilan dikontrol oleh suatu unit mikrokontroler. Masukan yang diperlukan untuk mengendalikan modul ini berupa data bus yang masih termultipleks dengan bus alamat serta 3 bit sinyal kontrol. Pengendalian dot matrik LCD dilakukan secara internal oleh *controller* yang sudah ada pada modul LCD. Konfigurasi pin LCD M1632 ditunjukkan dalam Gambar 2.15.



Gambar 2.15 Konfigurasi Pin LCD M1632

Sumber: www.delta-electronic.com

Fungsi masing-masing pin modul LCD M1632 dijelaskan pada Tabel 2.2.

Tabel 2.2 Fungsi Pin-pin Modul LCD

Nama Pin	Fungsi
D0-D7	Merupakan saluran data, berisi perintah dan data yang akan ditampilkan pada modul LCD.
E	Sinyal operasi awal. Sinyal ini akan mengaktifkan data tulis atau baca.
R/ W	Sinyal seleksi tulis dan baca: 0 = tulis dan 1 = baca
RS	Sinyal pemilih register internal: 0 = instruksi register (tulis), 1 = data register (tulis dan baca).
VEE	Untuk mengendalikan kecerahan modul LCD dengan mengubah-ubah nilai resistor variabel yang diumpangkan.
VCC	Tegangan sumber +5V
VSS	Terminal <i>ground</i>

Sumber: www.delta-electronic.com

Instruksi operasi pada modul LCD merupakan kombinasi bilangan biner yang melalui pin saluran data 8 bit (DB0 - DB7), pin seleksi register (RS) dan pin seleksi instruksi (R/W). Instruksi operasi pada modul LCD dijelaskan dalam Tabel 2.3.

Tabel 2.3 Instruksi pada Modul LCD

No	INSTRUKSI	RS	RW	D7	D6	D5	D4	D3	D2	D1	D0
1	<i>Display Clear</i>	0	0	0	0	0	0	0	0	0	1
2	<i>Cursor Home</i>	0	0	0	0	0	0	0	0	1	S
3	<i>Entry Mode Set</i>	0	0	0	0	0	0	0	1	I/D	S
4	<i>Display ON/OFF</i>	0	0	0	0	0	0	1	D	C	B
5	<i>Cursor Display Shift</i>	0	0	0	0	0	1	S/C	R/L	*	*
6	<i>Function Set</i>	0	0	0	0	1	D/L	1	*	*	*
7	<i>CGRAM Address Set</i>	0	0	0	1	A_{CG}					
8	<i>DDRAM Address Set</i>	0	0	1	A_{DD}						
9	<i>BF/ Address Read</i>	0	1	BF	AC						
10	<i>Data Write To CGRAM</i>	0	0	Write Data							
11	<i>Data Read From CGRAM</i>	1	1	Read Data							

Sumber: www.delta-electronic.com

Fungsi masing-masing instruksi adalah sebagai berikut:

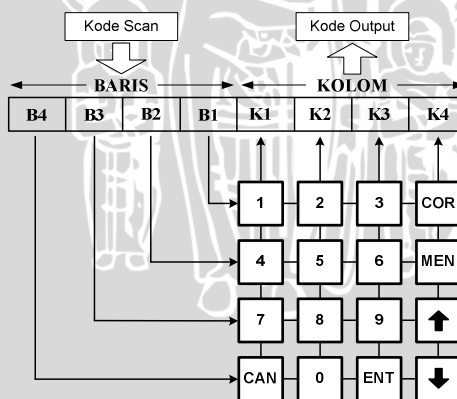
- 1) *Display Clear*: membersihkan tampilan yang ada pada modul LCD dan mengembalikan kursor kembali ke posisi semula.
- 2) *Cursor home*: hanya membersihkan semua tampilan dan kursor kembali semula.
- 3) *Entry mode set*: layar beraksi sebagai tampilan tulis.
 $S = 1 / 0$: menggeser layar
 $I / O = 1$: kursor bergerak ke kanan dan layar bergerak ke kiri.
 $I / O = 0$: kursor bergerak ke kiri dan layar bergerak ke kanan.
- 4) *Display on/off control*
 $D = 1$: layar *on*. $C = 1$: kursor *on*. $B = 1$: kursor berkedip-kedip.
 $D = 0$: layar *off*. $C = 0$: kursor *off*. $B = 0$: kursor tidak berkedip-kedip.
- 5) *Cursor display shift*
 $S/C = 1$: modul LCD diidentifikasi sebagai layar.
 $S/C = 0$: modul LCD diidentifikasi sebagai kursor.
 $R/L = 1$: menggeser satu spasi ke kanan.
 $R/L = 0$: menggeser satu spasi ke kiri.
- 6) *Function set*
 $DL = 1$: panjang data modul LCD pada 8 bit (DB7 - DB0).
 $DL = 0$: panjang data modul LCD pada 4 bit (DB7 - DB0).
 Bit *upper* ditransfer terlebih dahulu kemudian diikuti dengan 4 bit *lower*.
 $N = 1/0$: modul LCD menggunakan 2 atau 1 baris karakter.
 $P = 1/0$: modul LCD menggunakan 5×10 atau 5×7 dot matrik.
- 7) *CG RAM address set* : menulis alamat RAM ke karakter.
- 8) *DD RAM address set* : menulis alamat RAM ke tampilan.

- 9) *BF/address set* : $BF = 1/0$, modul LCD dalam keadaan sibuk atau tidak sibuk.
- 10) *Data write to CG RAM or DD RAM*: menulis byte ke alamat terakhir RAM yang dipilih.
- 11) *Data read from CG RAM or DD RAM*: membaca byte dari alamat terakhir RAM yang dipilih.

Display Data RAM (DD RAM) pada modul LCD masing-masing *line* mempunyai *range* alamat tersendiri. Alamat itu diekspresikan dengan bilangan heksadesimal. Untuk *line* 1 alamat berkisar antara 00H - 0FH sedangkan untuk *line* 2 berkisar antara 40H - 4FH.

2.9. Keypad

Keypad memiliki bentuk sederhana menyerupai *keyboard* kecil yang difungsikan untuk memberikan masukan data melalui tombol-tombol yang terdapat dalam papan keypad tersebut. Keypad merupakan penyederhanaan dari tombol-tombol *push button* yang disusun secara matrik. Keypad dapat dibedakan berdasarkan jumlah baris dan kolomnya, salah satunya adalah keypad matrik 4×4. Keypad matrik 4×4 terdiri dari 4 lajur kolom dan 4 lajur baris, dimana tiap baris dan kolom tersebut dihubungkan dengan tombol *push button* dengan jumlah keseluruhan sebanyak 16 buah. Bentuk dasar keypad 4×4 ditunjukkan dalam Gambar 2.16.



Gambar 2.16 Keypad 4×4
Sumber: www.grayhill.com

2.10. Printer LX-800

Printer yang digunakan dalam penelitian ini adalah tipe Epson LX-800 yang merupakan printer dot matrik, yakni sebuah printer yang menggunakan jarum dan pita sebagai alat pencetak atau hasil gambar dan tulisan (seperti mesin ketik). Printer Epson LX-800 merupakan contoh printer yang mempunyai 9 pin, dengan kualitas cetakan

tergantung pada jumlah jarum (pin) pada kepala cetaknya (*head print*). Printer Epson LX-800 ditunjukkan dalam Gambar 2.17.



Gambar 2.17 Printer Epson LX-800
Sumber : <http://www.rentit.biz/specs/pdf/lx-800.pdf>

- Beberapa keunggulan dari Printer Epson LX-800 adalah sebagai berikut:
 - ☑ Kecepatan *processing input* data 64kb
 - ☑ Mampu mencetak 337 karakter per-detik (12-CPI)
 - ☑ Efisiensi optimum *print-out* yaitu 5 part forms (1 original + 4 kopi).
 - ☑ Komunikasi data menggunakan USB, *interface* serial port dan paralel port.
 - ☑ Tidak memerlukan sebuah *driver*.
 - ☑ memiliki kecepatan cetak yang relatif tinggi.
- Spesifikasi produk printer Epson LX-800 adalah sebagai berikut:
 - ☑ Model: Epson LX-800
 - ☑ RAM *memory installed size*: 64 kb
 - ☑ Printer teknologi: Dot-matrix
 - ☑ Media *size maximum*: 10 min × 22 in, Roll (8.5 in)
 - ☑ Resolusi printer horizontal maksimum: 240 Dot per inci
 - ☑ Resolusi printer vertikal maksimum: 144 Dot per Inci
 - ☑ Maximum *print speed (Black and White)*: 337
 - ☑ Minimum *system requirements*: Microsoft Windows NT 4.0, Microsoft Windows 95/98, Microsoft Windows 3.1, Microsoft Windows 2000, Microsoft Windows NT 3.5
 - ☑ Printer *interface*: Paralel, Serial, USB.

2.10.1. Sistem Komunikasi Paralel

Pengiriman data menggunakan sistem transmisi elektronik, biasanya disebut dengan istilah komunikasi data (*data communication*). Komunikasi data adalah terjadinya pertukaran data antara komputer dengan terminal. Komunikasi ini juga dapat

dilakukan dengan jarak jauh antara terminal dengan komputer. Terminal-terminal yang sering dipakai antara lain adalah *disc drive*, pencetak (*printer*), *plotter*, layar tampilan, dan papan ketik. Pada sistem alat ini peranan komputer digantikan oleh mikrokontroler yang dapat mengendalikan kerja dari seluruh terminal dan terminal yang digunakan berupa printer yang menggunakan sistem komunikasi data paralel.

Port paralel adalah port data untuk mentransmisi 8 bit data dalam sekali detak, artinya data dalam 8 bit bisa langsung dikeluarkan secara paralel melalui port ini, oleh karena itu sering disebut *parallel port*. Ada 3 macam standar jenis port paralel pada komputer, yaitu SPP (*Standard Paralel Port*), EPP (*Enhanced Parallel Port*), dan ECP (*Extended Capability Port*). Sedangkan standar port paralel yang baru adalah IEEE 1284 yang dikeluarkan tahun 1994. Standar ini mendefinisikan 5 mode operasi sebagai berikut, yaitu Mode kompatibilitas, Mode *nibble*, Mode byte, Mode EPP (*enhanced parallel port*), dan Mode ECP (*extended capability port*).

Tujuan dari standar yang baru tersebut adalah untuk mendesain pemasangan *printer-printer* tipe baru beserta *drivernya* agar sesuai dengan peralatan yang lainnya. Seperti halnya standar *paralel port* sebelumnya (SPP), menggunakan perangkat standar yang tersedia pada port paralel asli. Sedangkan EPP dan ECP masih membutuhkan tambahan *hardware* untuk melakukan pekerjaan yang lebih cepat.

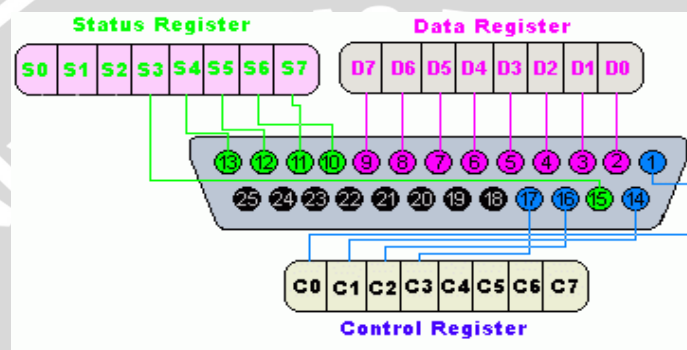
Port paralel *Extend* dan *Enhanced* menggunakan *hardware* tambahan untuk membangkitkan dan mengatur *handshaking*. Untuk mengeluarkan satu byte ke *printer* menggunakan mode kompatibilitas, *software* harus:

- 1) Menulis byte ke data *port*.
- 2) Mengecek untuk melihat apakah printer sibuk, bila sibuk printer tidak akan menerima data, sehingga data yang telah ditulis akan hilang
- 3) Membuat *strobe* (pin 1) ke kondisi *low*. Ini memberitahukan printer bahwa data yang benar telah berada pada *line* data.
- 4) Membuat *strobe* dalam logika *high* kembali setelah menunggu sekitar 5 μ s setelah membuat *strobe* ke kondisi *low*.

Batas kecepatan data ditentukan oleh aktifitas sinyal *strobe* dan *busy*. Pada mode EPP dan ECP penambahan perangkat kerasnya memungkinkan port untuk melakukan aktifitas *handshaking* dengan kecepatan tinggi sekitar 1-2 megabytes per detik. Mode ECP juga mempunyai keuntungan dengan pemakaian saluran DMA dan buffer FIFO.

Tabel 2.4 merupakan gambar dan tabel nama pin dari konektor DB25 dan centronics dengan jumlah konektor 34 pin. DB25 adalah konektor yang paling umum

digunakan pada komputer sebagai port paralel, sedangkan DB34 konektor centronics umum ditemukan pada *printer*. IEEE 1284 adalah standar yang menentukan 3 konektor berbeda yang dapat digunakan dengan port paralel, yaitu 1284 tipe A adalah konektor DB25 yang dapat ditemukan di hampir semua komputer, 1284 tipe B adalah konektor centronics 34 pin yang umum ditemukan di *printer*, IEEE 1284 tipe C adalah konektor 36 pin seperti centronics, tetapi ukurannya lebih kecil dan lebih memuaskan. Konektor ini diklaim untuk mempunyai pintu klip lebih baik, kemampuan elektrik lebih baik dan lebih mudah untuk memasang. Jenis C konektor direkomendasikan untuk disain baru. Konfigurasi pin port paralel ditunjukkan dalam Gambar 2.18.



Gambar 2.18 Konfigurasi Pin Port Paralel

Sumber: http://www.zytrax.com/tech/layer_1/cables/tech_rs232.232.htm

Tabel 2.4. Nama Pin dari Konektor *Parallel Port* DB25 dan Centronics

DB 25	Pin Centronics	SPP Signal	Arah In/Out	Register	Hardware di invert
1	1	<i>nStrobe</i>	In/Out	Control	Ya
2	2	Data Bit 0 (LSB)	Out	Data	
3	3	Data Bit 1	Out	Data	
4	4	Data Bit 2	Out	Data	
5	5	Data Bit 3	Out	Data	
6	6	Data Bit 4	Out	Data	
7	7	Data Bit 5	Out	Data	
8	8	Data Bit 6	Out	Data	
9	9	Data Bit7 (MSB)	Out	Data	
10	10	<i>nAcknowledge</i>	In	Status	
11	11	<i>Busy</i>	In	Status	Ya
12	12	<i>Paper End/Out</i>	In	Status	
13	13	<i>Select</i>	In	Status	
14	14	<i>nAuto-Linefeed</i>	In/Out	Control	Ya
15	32	<i>nError / nFault</i>	In	Status	
16	31	<i>nInitialize Printer</i>	In/Out	Control	
17	36	<i>nSelect Input</i>	In/Out	Control	Ya
18-25	19-30	<i>Ground</i>	Gnd		

Sumber: Budiharto, Widodo S.Si. M.Kom; Firmansyah, Sigit. 2005. *Elektronika Digital dan Mikroprosesor*. Andi Offset : Yogyakarta.

BAB III

METODOLOGI PENELITIAN

Kajian yang dilakukan dalam skripsi ini adalah perancangan dan pembuatan alat penimbang perhiasan emas dilengkapi dengan pencetak sertifikat perhiasan. Metode yang digunakan untuk merealisasikan alat yang akan dibuat secara umum adalah sebagai berikut:

- 1) Studi Literatur
- 2) Penentuan Spesifikasi Alat
- 3) Perancangan Alat
- 4) Pembuatan Alat
- 5) Pengujian Alat dan Analisis

3.1 Studi Literatur

Studi literatur dilakukan untuk mendapatkan pengetahuan dasar tentang segala sesuatu yang mendukung perancangan serta pembuatan alat ini. Pembuatan alat ini menggunakan referensi dari buku-buku maupun artikel dari internet untuk mengetahui karakteristik komponen, prinsip kerja serta teori yang menunjang.

3.2 Penentuan Spesifikasi Alat

Setelah mempelajari berbagai literatur yang menunjang kemudian dari permasalahan yang ada ditentukan spesifikasi alat yang akan dirancang. Penentuan spesifikasi alat dilakukan untuk memudahkan melakukan perancangan dan pembuatan alat serta memberikan informasi tentang kemampuan alat.

3.3 Perencanaan Alat

Berdasarkan studi literatur, tahap selanjutnya adalah perencanaan alat. Dalam perencanaan ini langkah-langkah yang dilakukan meliputi penyusunan diagram blok sistem untuk mempermudah pemahaman tentang alur kerja dari alat yang akan dibuat, pembuatan skema rangkaian dan perhitungan besaran-besaran listriknya.

Selanjutnya dilakukan perhitungan dan perencanaan tiap-tiap blok rangkaian dengan mengacu pada data dari buku data komponen elektroniknya. Dari data tersebut dilakukan analisis dan perhitungan untuk mencapai hasil optimal dari komponen yang digunakan, yang akan disesuaikan dengan komponen yang ada dipasaran.

3.4 Pembuatan Alat

Pembuatan alat dilakukan berdasarkan perencanaan dari masing-masing blok. Tahap pembuatan alat ini diawali dengan merancang tata letak komponen melalui perangkat lunak *Protel Advanced Schematic 3.20 EDA/Client* dan *Protel Advanced PCB Design 2.5*, pembuatan papan rangkaian tercetak, perakitan komponen, serta pembuatan *software*.

3.5 Pengujian Alat

Untuk mengetahui tingkat keberhasilan sistem dan kesesuaian dengan perencanaan maka dilakukan pengujian rangkaian. Pengujian dilakukan pada masing-masing blok secara keseluruhan. Pengujian yang dilakukan yaitu:

- Pengujian antarmuka keypad,
- Pengujian antarmuka modul LCD,
- Pengujian rangkaian RTC,
- Pengujian rangkaian EEPROM,
- Pengujian rangkaian ADC,
- Pengujian rangkaian pengkondisi sinyal,
- Pengujian rangkaian port paralel,
- Pengujian sistem secara keseluruhan.

BAB IV

PERENCANAAN DAN PEMBUATAN

Pada bab ini membahas mengenai spesifikasi alat, perencanaan sistem, prinsip kerja sistem, perancangan perangkat keras (*hardware*) dan perancangan perangkat lunak (*software*). Bagian perancangan perangkat keras terbagi menjadi beberapa bagian yaitu bagian sensor *load cell*, bagian rangkaian pengkondisi sinyal, bagian rangkaian ADC, bagian mikrokontroler AT89S52, bagian rangkaian RTC, bagian rangkaian EEPROM, bagian konfigurasi pin SDA dan SCL, bagian antarmuka modul LCD, bagian antarmuka keypad, dan bagian port paralel.

Masing-masing bagian tersebut disusun dengan pemilihan beberapa jenis komponen dengan fungsi sesuai dengan perancangan, sehingga akan dihasilkan satu bentuk bagian dengan fungsi sesuai dengan perancangan yang dilakukan diawal. Bagian perencanaan perangkat lunak membahas mengenai perancangan pembuatan diagram alir program.

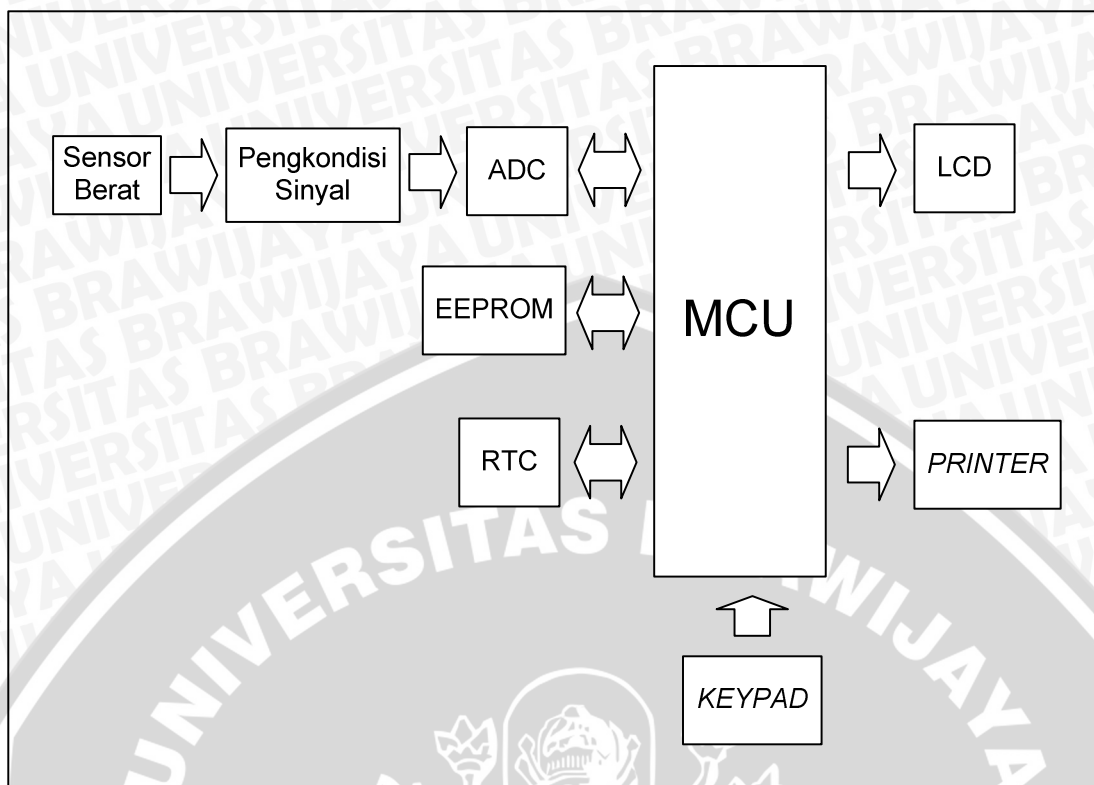
4.1 Spesifikasi Alat

Spesifikasi alat ditetapkan terlebih dahulu sebagai acuan dalam perancangan. Spesifikasi alat yang direncanakan adalah sebagai berikut:

- 1) *Range* pengukuran berat yang ditimbang 0,5 - 15 gram,
- 2) Resolusi pengukuran berat sebesar 0,1 gram,
- 3) Menggunakan LCD 16×2 sebagai tampilan,
- 4) Menggunakan keypad 4×4 sebagai media untuk memilih menu,
- 5) Menggunakan printer Epson LX-800 sebagai media pencetak sertifikat,
- 6) Mikrokontroler AT89S52 sebagai pengolah data dan pusat kontrol,
- 7) Sistem menggunakan catu daya DC +5 volt dan 15 volt.

4.2 Perencanaan Sistem

Diagram blok sistem merupakan salah satu bagian terpenting dalam perancangan dan pembuatan alat ini, karena dari diagram blok dapat diketahui prinsip kerja keseluruhan rangkaian. Tujuan lain diagram blok ini adalah memudahkan proses perancangan dan pembuatan pada masing-masing bagian, sehingga akan terbentuk suatu sistem yang sesuai dengan perancangan sebelumnya. Diagram blok sistem ditunjukkan dalam Gambar 4.1.



Gambar 4.1 Diagram Blok Keseluruhan Sistem

Berdasarkan diagram blok pada Gambar 4.1, dapat dijelaskan secara umum mengenai bagian-bagian yang menyusun keseluruhan sistem dari alat ini antara lain:

1) Sensor Berat

Sensor berat berfungsi untuk mendeteksi berat suatu benda.

2) Pengkondisi Sinyal

Pengkondisi sinyal digunakan untuk menguatkan sinyal analog yang berasal dari keluaran sensor berat.

3) ADC (*Analog to Digital Converter*)

ADC berfungsi untuk mengubah sinyal analog menjadi sinyal digital yang akan diumpankan ke mikrokontroler.

4) Mikrokontroler

Berfungsi sebagai pusat pengontrol semua rangkaian.

5) RTC (*Real Time Clock*)

RTC berfungsi untuk memberikan informasi waktu (detik, menit, jam, tanggal, bulan, dan tahun) secara kontinu ke mikrokontroler.

6) EEPROM (*Electrically Erasable and Programmable ROM*)

EEPROM digunakan untuk menyimpan data transaksi.

7) LCD (*Liquid Crystal Display*)

LCD merupakan tampilan yang digunakan untuk menampilkan hasil penimbangan dengan satuan gram dan dalam bentuk angka nominal, serta menampilkan data transaksi yang telah disimpan.

8) Keypad

Keypad berfungsi sebagai media untuk memilih menu yang akan ditampilkan ke *display* (LCD), *entry* data nama pembeli dan *entry* nilai tukar perhiasan emas.

9) Printer

Printer berfungsi sebagai media untuk mencetak data hasil proses penimbangan sekaligus hasil cetakan tersebut difungsikan sebagai sertifikat perhiasan.

4.3 Prinsip Kerja Sistem

Alat ini menggunakan sensor *load cell* untuk mendeteksi berat benda yang akan ditimbang. Prinsip kerja alat ini jika ada penambahan berat, sensor akan mendeteksi berat benda. Kemudian tegangan keluaran dari sensor tersebut dikuatkan oleh rangkaian pengkondisi sinyal agar dapat diolah oleh rangkaian ADC. Rangkaian ADC akan mengubah sinyal analog menjadi sinyal digital. Kemudian hasil konversi dari rangkaian ADC akan diproses oleh rangkaian mikrokontroler untuk ditampilkan ke *display* LCD sebagai data berat dalam satuan gram. Selanjutnya setelah proses penimbangan dan *entry* data selesai, data transaksi disimpan pada EEPROM.

Alat ini juga menggunakan keypad yang berfungsi sebagai media memilih menu, *entry* data nama pembeli dan *entry* nilai tukar perhiasan emas per gram serta dilengkapi dengan *printer* untuk mencetak data hasil penimbangan dan sekaligus hasil cetakan tersebut digunakan sebagai sertifikat perhiasan.

4.4 Perancangan Perangkat Keras

Perancangan dari masing-masing bagian dapat diuraikan sebagai berikut:

1) Perancangan bagian perangkat keras (*hardware*) meliputi:

- Bagian sensor *load cell*
- Bagian rangkaian pengkondisi sinyal,
- Bagian rangkaian ADC,
- Bagian rangkaian mikrokontroler,

- Bagian rangkaian RTC,
- Bagian rangkaian EEPROM,
- Bagian konfigurasi pin SDA dan SCL,
- Bagian antarmuka modul LCD,
- Bagian antarmuka keypad,
- Bagian antarmuka port paralel.

2) Perancangan bagian perangkat lunak (*software*).

4.4.1. Perancangan Sensor *Load Cell*

Pada perancangan ini digunakan sebuah *load cell* dengan kapasitas 200 gram.

Load cell ini memiliki spesifikasi sebagai berikut:

- *Rated capacity-R.C. (E_{max})* : 200, 300, 600 gram
- *Output sensitivity* : 0.8 mV/V
- *Rated output tolerance* : $0.1 \pm$ mV/V
- *Zero balance* : $0.045 \pm$ mV/V
- *Zero Return, 2 minutes* : $0.0100, 0.0033 \pm$ % of applied load
- *Zero Return, 30 minutes* : $0.017 \pm$ % of applied load
- *Temperature effect on zero* : $0.0040, 0.004 \pm$ % of rated output/ $^{\circ}$ C
- *Temperature effect on output* : $0.0020, 0.001 \pm$ % of load/ $^{\circ}$ C
- *Temp. range, compensated* : +5 to +40 $^{\circ}$ C
- *Temp. range, safe* : -3 to +70 $^{\circ}$ C
- *Eccentric loading error* : $0.0033 \pm$ % of rated load/cm
- *Maximum safe central overload* : 150 % of R.C.
- *Ultimate central overload* : 250 % of R.C.
- *Excitation, recommended* : 10 Vdc or Vac rms
- *Excitation maximum* : 15 Vdc or Vac rms
- *Input impedance* : 415 ± 20 ohms
- *Output impedance* : 350 ± 3 ohms
- *Insulation resistance* : > 2000 Mega-ohms
- *Recommended torque* : 2.0 N*m

Pada *datasheet load cell* tersebut ditunjukkan bahwa *load cell* memiliki *output sensitivity* sebesar 0,8 mV/V dan *excitation maximum* sebesar 15 Vdc. Jadi pada berat 200 gram, tegangan keluaran (V_{out}) *load cell* sebesar 12 mV (diperoleh dengan

mengalikan *output sensitivity* dengan *excitation maximum*). Sedangkan untuk mencari nilai tegangan keluaran (V_{out}) *load cell* pada berat 15 gram dilakukan perhitungan dengan Persamaan (4-1) sebagai berikut:

$$\frac{m_1}{V_1} = \frac{m_2}{V_2} \quad (4-1)$$

Keterangan:

m_1 : nilai berat 1 (gram)

m_2 : nilai berat 2 (gram)

v_1 : nilai tegangan 1 (mV)

v_2 : nilai tegangan 2 (mV)

$$\frac{m_1}{V_1} = \frac{m_2}{V_2}$$

$$\frac{200 \text{ gram}}{12 \text{ mV}} = \frac{15 \text{ gram}}{V_2}$$

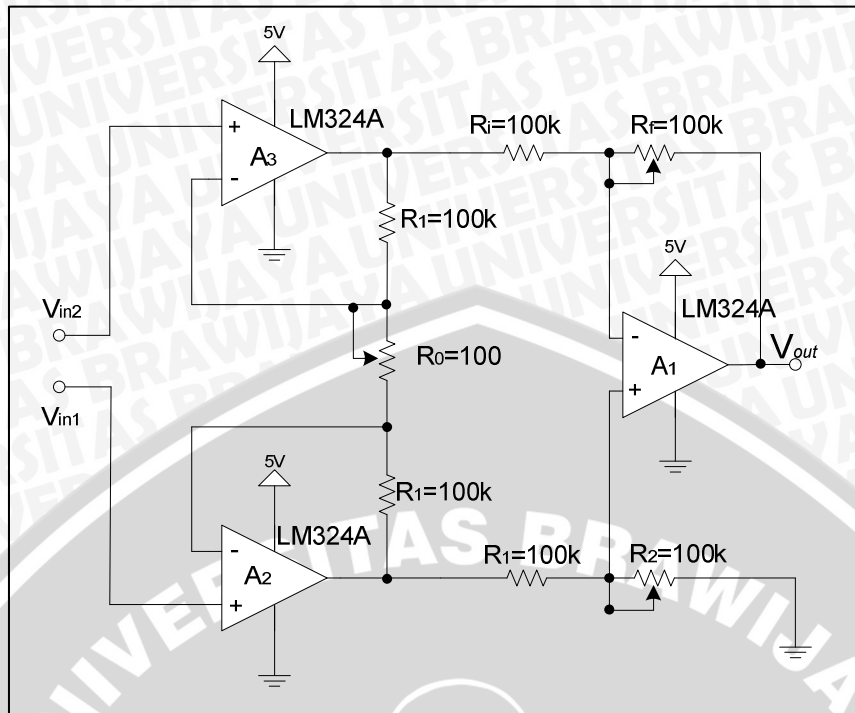
$$V_2 = \frac{15 \text{ gram}}{200 \text{ mV}} \times 12 \text{ mV}$$

$$V_2 = 0.9 \text{ mV}$$

Jadi pada berat 15 gram, tegangan keluaran (V_{out}) *load cell* sebesar 0,9 mV. Tegangan keluaran ini sangat kecil, sehingga harus dikuatkan terlebih dahulu oleh rangkaian pengkondisi sinyal.

4.4.2. Perancangan Rangkaian Pengkondisi Sinyal

Dalam perancangan ini rangkaian pengkondisi sinyal terdiri dari rangkaian penguat instrumentasi (*instrumentation amplifier*). Rangkaian penguat instrumentasi disusun dari tiga buah op-amp dan beberapa tahanan. Rangkaian penguat instrumentasi ini digunakan untuk menguatkan sinyal analog yang berasal dari keluaran sensor berat. Rangkaian pengkondisi sinyal ditunjukkan dalam Gambar 4.2.



Gambar 4.2 Rangkaian Pengkondisi Sinyal

Rangkaian penguat instrumentasi dalam Gambar 4.2 dibentuk oleh rangkaian penguat diferensial dan rangkaian penguat penyangga. Penguatan (*gain*) penguat instrumentasi dapat diatur melalui tahanan R_0 . Dalam perancangan ini nilai tahanan $R_1 = R_2 = R_i = R_f$ sebesar $100 \text{ k}\Omega$, dipilih nilai tahanan sebesar $100 \text{ k}\Omega$ untuk memudahkan dalam perancangan. Sedangkan untuk mengetahui nilai tahanan R_0 dilakukan perhitungan dengan menggunakan Persamaan 2-17, yaitu sebagai berikut:

$$V_{\text{out}} = (V_1 - V_2) \left(1 + \frac{2R_1}{R_0}\right) \left(\frac{R_f}{R_i}\right)$$

Untuk analisis rangkaian penguat instrumentasi di atas, tegangan keluaran yang diinginkan saat tidak ada beban atau berat 0 gram adalah 0 V dan untuk berat maksimal 15 gram tegangan keluaran yang diinginkan adalah 2,5 V.

- Analisis rangkaian saat berat 0 gram

Diketahui keluaran sensor *load cell* pada saat tidak ada beban atau berat 0 gram masing-masing adalah $V_{\text{in1}} = 2,5 \text{ V}$ dan $V_{\text{in2}} = 2,5 \text{ V}$. Maka untuk mencari tegangan keluaran atau V_{out} adalah sebagai berikut:

$$V_{\text{out}} = (V_{\text{in1}} - V_{\text{in2}}) \left(1 + \frac{2R_1}{R_0}\right) \left(\frac{R_f}{R_i}\right)$$

$$V_{\text{out}} = (2,5 - 2,5) \left(1 + \frac{2 \times 100.000}{R_0}\right) \left(\frac{100.000}{100.000}\right)$$

$$V_{\text{out}} = (0) \left(1 + \frac{2 \times 100.000}{R_0}\right)$$

$$V_{\text{out}} = 0 \text{ volt}$$

- Analisis rangkaian saat berat 15 gram

Tegangan keluaran *load cell* pada berat maksimum 15 gram sebesar 0,9 mV. Sedangkan dalam perancangan ini diinginkan tegangan masukan ADC sebesar 2,5 V. Jadi untuk mencari besarnya penguatan pada rangkaian penguat instrumentasi di atas dilakukan dengan perhitungan sebagai berikut:

$$V_{\text{out}} = (V_{\text{in1}} - V_{\text{in2}}) \left(1 + \frac{2R_1}{R_0}\right) \left(\frac{R_f}{R_i}\right)$$

$$2,5 = (0,0009) \left(1 + \frac{2 \times 100.000}{R_0}\right) \left(\frac{100.000}{100.000}\right)$$

$$\left(1 + \frac{2 \times 100.000}{R_0}\right) = 2777,78$$

$$\frac{2 \times 100.000}{R_0} = 2776,78$$

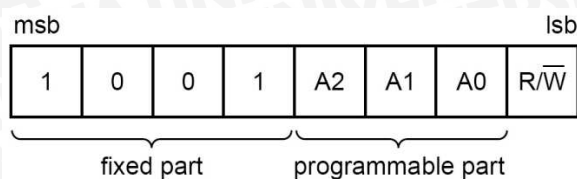
$$R_0 = \frac{2 \times 100.000}{2776,78}$$

$$R_0 = 71,999 \approx 72 \Omega$$

Dari perhitungan di atas diperoleh R_0 sebesar 72Ω . Dalam perancangan ini, karena nilai tahanan 72Ω tidak ada di pasaran, maka digunakan resistor variabel 100Ω .

4.4.3. Perancangan Rangkaian ADC

Mikrokontroler hanya dapat memproses data digital maka sinyal yang masuk ke dalam mikrokontroler harus sinyal digital. Sinyal dari sensor berat yang telah dikuatkan harus dikonversikan terlebih dahulu menjadi sinyal digital dengan rangkaian ADC. Dalam perancangan ini digunakan ADC PCF8591 yang merupakan ADC 8 bit, dengan empat masukan analog yang dimultiplex menjadi data 8 bit digital. ADC PCF8591 dengan sistem bus I^2C akan aktif jika mengirimkan alamat yang benar ke piranti ini. Pengalamatan dilakukan dengan diawali dengan nomor *group* (1001), nomor *chip* (A2, A1, A0) dan mode tulis atau baca. Nomor *group* adalah nomor yang diberikan oleh philips sebagai pencipta I^2C dan nomor *chip* adalah nomor yang diberikan pada masing-masing *chip* lewat pin A0, A1 dan A2 dari masing-masing IC. Pengalamatan IC ADC PCF8591 ditunjukkan dalam Gambar 4.3.



Gambar 4.3 Pengalamatan IC ADC PCF8591

Sumber: Philips, 2003: 5

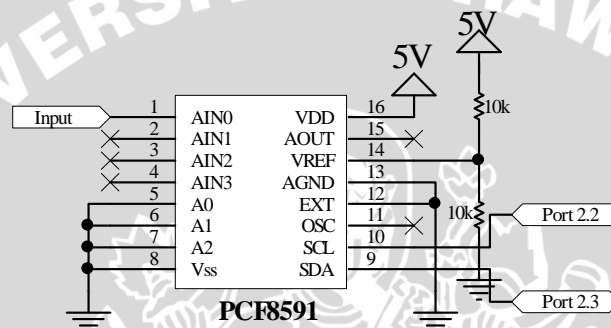
Pada perancangan ini, pin A0, A1 dan A2 dihubungkan ke *ground* karena pin-pin tersebut tidak digunakan untuk memprogram pengalaman *hardware*. Sedangkan pin V_{REF} dihubungkan ke rangkaian pembagi tegangan untuk mendapatkan tegangan referensi 2,5V. Tegangan referensi (V_{ref}) diperoleh dengan perhitungan Persamaan (4-2) sebagai berikut:

$$V_{ref} = \frac{R_1}{R_1 + R_2} \times VCC \quad (4-2)$$

$$V_{ref} = \frac{10.000}{10.000 + 10.000} \times 5V$$

$$V_{ref} = 2,5 V$$

Rangkaian ADC PCF8591 ditunjukkan dalam Gambar 4.4.



Gambar 4.4 Rangkaian ADC PCF8591

Kenaikan tiap langkah untuk kenaikan keluaran biner (resolusi tegangan ADC) diperoleh dengan Persamaan (2-20), yaitu:

$$V_{step} = \frac{V_{ref}}{2^n}$$

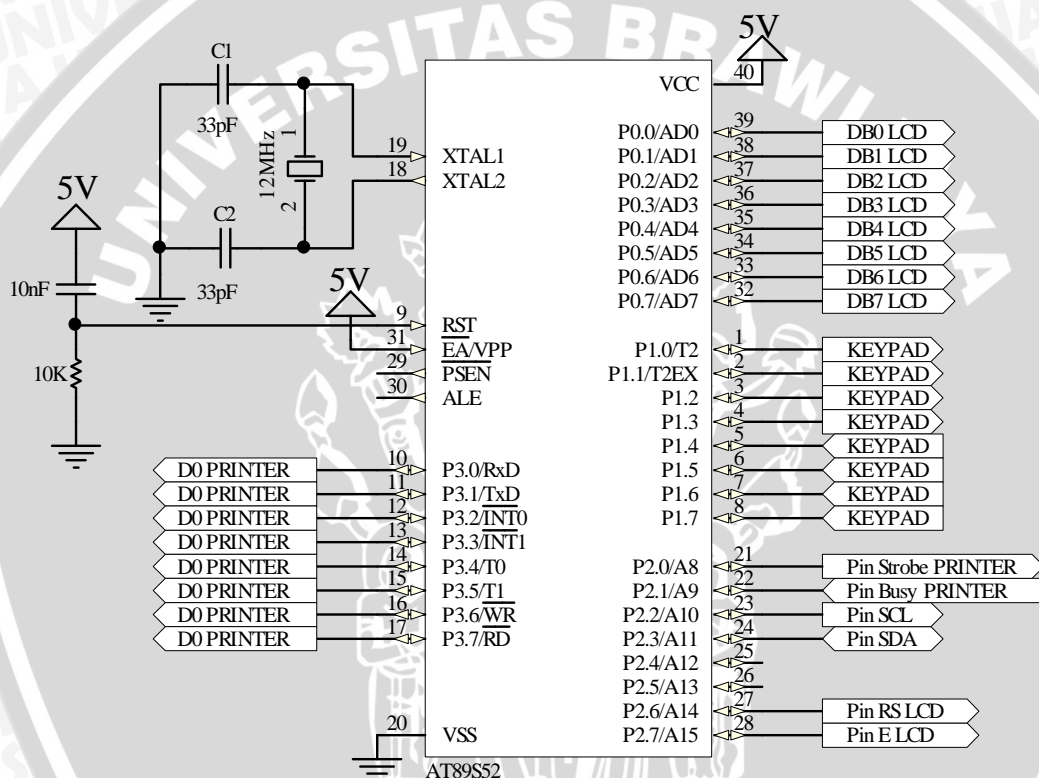
$$V_{step} = \frac{2,5}{256}$$

$$V_{step} = 0.0097656V \approx 9,77mV$$

Pada perancangan rangkaian pengondisi sinyal sensor berat diinginkan resolusi sebesar 0,1 gram untuk pengukuran berat pada *range* 0,5 gram sampai 15 gram, sehingga pengukuran berat ini mempunyai 145 step (diperoleh dari perhitungan $(15 \text{ gram} - 0,5 \text{ gram}) / 0,1 \text{ gram}$). Tegangan keluaran minimum menurut perhitungan dari rangkaian pengkondisi sinyal sensor berat sebesar 0 V pada saat berat 0 gram (tidak ada beban), sehingga keluaran ADC PCF8591 adalah $0 \text{ mV} / 9,77mV = 0$ (desimal). Sedangkan tegangan keluaran maksimum menurut perhitungan dari rangkaian pengkondisi sinyal sensor berat sebesar 2,5 V pada saat berat 15 gram, sehingga keluaran ADC PCF8591 adalah $2500mV / 9,77mV = 255$ (desimal).

4.4.4. Mikrokontroler AT89S52

Mikrokontroler yang digunakan pada rangkaian ini adalah mikrokontroler tipe AT89S52 yang merupakan keluarga dari MCS-51. Komponen ini mempunyai 8 kbyte Flash EPROM (*Erasable and Programmable Read Only Memory*), 32 pin masukan keluaran dimana tiap pin tersebut dapat diprogram secara paralel dan tersendiri. Pemilihan mikrokontroler jenis ini karena mudah diperoleh dipasaran dan mempunyai memory internal. Sebagai tempat pengolahan data dan pengontrolan alat, pin-pin AT89S52 dihubungkan pada rangkaian pendukung membentuk suatu minimum sistem ditunjukkan dalam Gambar 4.5.



Gambar 4.5 Rangkaian Mikrokontroler AT89S52

Mikrokontroler AT89S52 mempunyai 23 jalur I/O yang dapat diprogram menjadi masukan atau keluaran. 23 jalur I/O ini dikelompokkan menjadi 4 kelompok, yaitu port 0, 1, 2, dan 3. Pada perancangan ini, pin-pin yang digunakan adalah sebagai berikut:

- 1) Port 0
- 2) P0.0 - P0.7 digunakan sebagai keluaran mengirimkan data ke alamat bus data LCD DB0 - DB7.
- 3) Port 1
P1.0 - P1.7 digunakan sebagai saluran masukan data dari keypad.

4) Port 2

P2.0 digunakan sebagai pengirim perintah *strobe* pada port paralel printer,

P2.1 digunakan sebagai penerima sinyal *busy* dari port paralel printer,

P2.2 digunakan sebagai pengirim *clock* SCL,

P2.3 digunakan sebagai saluran masukan dan keluaran data secara serial SDA,

P2.6 digunakan sebagai pengirim perintah RS (*Register Selection*) pada LCD,

P2.7 digunakan sebagai pengirim perintah E (*Enable*) pada LCD.

5) Port 3

P3.0 - P3.7 digunakan sebagai keluaran mengirimkan data ke alamat *bus* data *printer* D0 - D7.

6) XTAL1 dan XTAL2

Digunakan sebagai masukan dari rangkaian osilator kristal. Rangkaian osilator kristal terdiri dari kristal osilator 12 MHz, kapasitor C₁ dan C₂ yang digunakan masing-masing bernilai 33 pF, akan membangkitkan pulsa *clock* yang menjadi penggerak bagi seluruh operasi internal mikrokontroler.

7) VCC

VCC dihubungkan dengan tegangan sebesar +5V.

8) GND

GND dihubungkan ke *ground* catu daya.

9) *Reset*

Digunakan untuk *mereset* program kontrol mikrokontroler, maka pin *reset* diberi logika tinggi selama sekurangnya dua siklus mesin (24 periode osilator). Untuk membangkitkan sinyal *reset*, kapasitor dihubungkan dengan VCC dan sebuah resistor yang dihubungkan ke *ground*.

10) EA

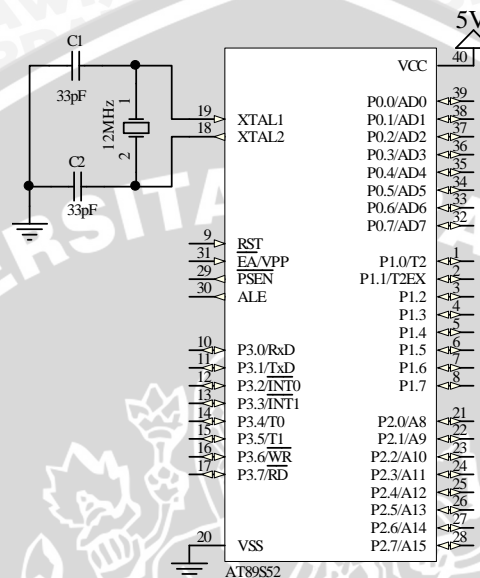
Pin EA dihubungkan dengan VCC yang berarti pin ini akan berkeadaan *high* sehingga akan berfungsi untuk menjalankan program yang ada pada memori internal.

Perancangan Rangkaian Osilator

Osilator internal mikrokontroler AT89S52 dapat dibangkitkan dengan menggunakan kristal sebagai pembangkit pulsa dimana besarnya kristal yang diizinkan

sebesar 0 - 33 MHz. Dalam perancangan digunakan kristal sebesar 12 MHz, sehingga kecepatan pelaksanaan instruksi per siklus sebesar $1 \mu\text{s}$ ($(1/12\text{MHz}) \times 12$ siklus periode).

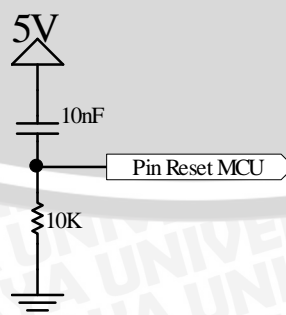
Rangkaian osilator ditunjukkan dalam Gambar 4.6. Kristal dihubungkan ke pin 18 (X₂) dan 19 (X₁) pada mikrokontroler AT89S52 dengan menambahkan C₁ dan C₂ sebesar 33 pF. Besarnya kapasitansi C₁ dan C₂ disesuaikan dengan spesifikasi dalam lembar *datasheet* AT89S52 yaitu $30 \text{ pF} \pm 10 \text{ pF}$. (Atmel, 2001: 15).



Gambar 4.6 Rangkaian Osilator

Perancangan Rangkaian Reset

Dalam rangkaian mikrokontroler diperlukan rangkaian *reset*. Untuk mereset mikrokontroler AT89S52, pin RST harus diberi logika tinggi selama sekurangnya dua siklus mesin (24 periode osilator). Untuk membangkitkan sinyal *reset*, sebuah kapasitor dihubungkan dengan VCC dan sebuah resistor dihubungkan ke *ground*. Rangkaian *reset* ditunjukkan dalam Gambar 4.7.



Gambar 4.7 Rangkaian Reset

Kristal yang digunakan mempunyai frekuensi sebesar 12 MHz, maka satu periode membutuhkan waktu sebesar:

$$T = \frac{1}{f_{XTAL}} = \frac{1}{12 \times 10^6} = 0,0833 \times 10^{-6} s$$

Sehingga waktu minimal logika tinggi yang dibutuhkan untuk mereset mikrokontroler adalah:

$$\begin{aligned} t_{reset(min)} &= T \times \text{periode yang dibutuhkan} \\ &= 0,0833 \times 10^{-6} \cdot 24 \\ &= 1,99 \mu s \approx 2 \mu s \end{aligned}$$

Berdasarkan perhitungan di atas maka mikrokontroler membutuhkan waktu minimal 2 μs untuk mereset. Waktu minimal ini dijadikan pedoman untuk menentukan nilai R dan C. Nilai C telah ditetapkan sebesar 10 nF, maka berdasarkan Persamaan 2-24, nilai R adalah sebagai berikut:

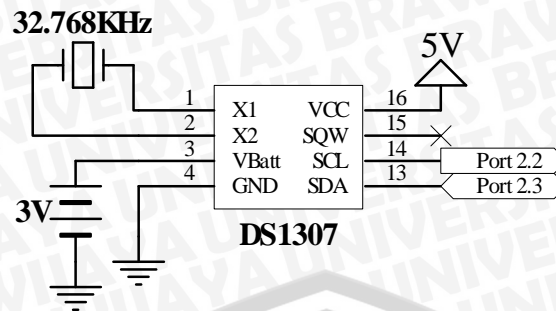
$$\begin{aligned} t &= 0,357 \cdot R \cdot C \\ 2 \times 10^{-6} &= 0,357 \cdot R \cdot 10 \times 10^{-9} \\ R &= \frac{2 \times 10^{-6}}{0,357 \cdot 10 \times 10^{-9}} \\ R &= 560,224 \Omega \end{aligned}$$

Jadi dengan nilai komponen $C = 10$ nF nilai resistor yang dapat memenuhi syarat untuk mereset mikrokontroler harus di atas 560,224 Ω . Pada perancangan ini dipilih nilai resistor 10 k Ω , sehingga waktu logika tinggi untuk mereset lebih besar dari 2 μs yaitu :

$$\begin{aligned} t &= 0,357 \cdot R \cdot C \\ &= 0,357 \cdot 10 \times 10^3 \Omega \cdot 10 \times 10^{-9} F \\ &= 35,7 \mu s \end{aligned}$$

4.4.5. Perancangan Rangkaian RTC

Dalam perancangan ini RTC (*Real Time Clock*) digunakan untuk memberikan informasi waktu. RTC yang digunakan adalah RTC DS1307. IC RTC ini memiliki dua buah catu daya. Pin VCC dihubungkan ke catu daya utama 5 V dan pin V_{BAT} dihubungkan dengan baterai 3 V. V_{BAT} berfungsi untuk memelihara kerja dari IC DS1307 jika catu daya utama tidak mampu untuk mencatu atau alat dimatikan. Pin X1 dan X2 dihubungkan dengan kristal 32,768 kHz untuk memberikan frekuensi yang sesuai. Sedangkan pin SDA terhubung dengan port 2.3 mikrokontroler dan pin SCL terhubung dengan pin 2.2 mikrokontroler. Rangkaian RTC DS1307 ditunjukkan dalam Gambar 4.8.

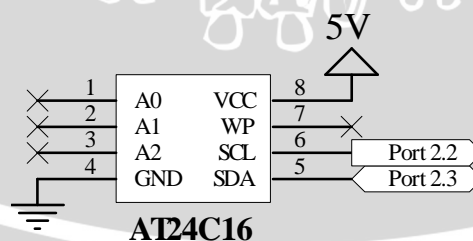


Gambar 4.8 Rangkaian RTC DS1307

4.4.6. Perancangan Rangkaian EEPROM

EEPROM digunakan untuk menyimpan data hasil penimbangan berupa data jam, data tanggal, data berat, nama pembeli, nilai tukar perhiasan, harga perhiasan, ongkos atau biaya pembuatan dan total harga perhiasan. Dalam perancangan ini digunakan IC EEPROM AT24C16. Pemilihan penggunaan EEPROM ini didasarkan pada kapasitas memori yang dimilikinya yaitu sebesar 16 kbit atau 2 kbyte.

IC EEPROM AT24C16 menggunakan protokol I²C, sehingga hanya memerlukan dua buah jalur saja dalam berkomunikasi dengan mikrokontroler yaitu melalui pin SDA dan SCL. Pin SDA dihubungkan ke port 2.3 mikrokontroler dan pin SCL dihubungkan ke port 2.2 mikrokontroler. Pin A0, A1 dan A2 merupakan pin yang digunakan untuk fasilitas penomoran *chip*, hal ini diperlukan apabila dalam satu rangkaian digunakan lebih dari satu IC EEPROM yang sejenis. Karena dalam perancangan ini hanya menggunakan satu buah IC EEPROM, maka pin A0, A1, A2 tidak digunakan. Pin WP (*Write Protect*) berfungsi untuk melindungi isi yang disimpan di dalam IC EEPROM. Rangkaian EEPROM AT24C16 ditunjukkan dalam Gambar 4.9.



Gambar 4.9 Rangkaian EEPROM AT24C16

Penyimpanan *database* hasil penimbangan ke dalam EEPROM memerlukan kapasitas 2048 byte dari alamat 000h hingga 7f8h. Dalam perancangan ini, kebutuhan memori untuk setiap transaksi atau penimbangan yang akan disimpan pada EEPROM AT24C16 ditunjukkan dalam Tabel 4.1.

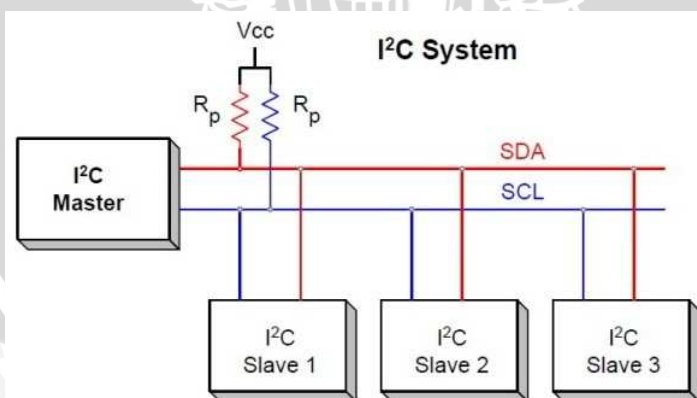
Tabel 4.1 Kebutuhan Memori untuk Setiap Transaksi

No.	Penggunaan Memori	Byte
1	Jam	2
2	Tanggal	1
3	Bulan	1
4	Tahun	1
5	Nama Pembeli	16
6	Nilai Tukar Perhiasan	6
7	Berat	3
8	Harga Perhiasan	6
9	Ongkos	6
10	Total Harga Perhiasan	6
Jumlah byte		48

Kebutuhan memori untuk setiap penimbangan atau setiap transaksi adalah 48 byte. Namun untuk cadangan, agar tidak terjadi penumpukan, maka akan disediakan ruang 20 byte untuk setiap transaksi. Sistem ini dapat digunakan untuk menyimpan database 30 transaksi, sehingga alamat yang digunakan adalah 2040 alamat. Yaitu mulai alamat 000h hingga 7f8h.

4.4.7. Perancangan Konfigurasi Pin SDA dan SCL

Dalam perancangan ini IC ADC, RTC dan EEPROM sama-sama menggunakan protokol I²C dalam berkomunikasi dengan mikrokontroler. Sehingga masing-masing pin SDA dan SCL dari IC-IC tersebut digabung menjadi satu. Konfigurasi pin SDA dan SCL ditunjukkan dalam Gambar 4.10.



Gambar 4.10 Konfigurasi pin SDA dan SCL

Pin SDA dan SCL adalah *open drain*, maka diperlukan resistor *pull-up* sebagai penyedia level logika tinggi. Berdasarkan *SMBus and I2C Bus Design*, ada tiga hal yang harus dipertimbangkan saat menghitung nilai resistor *pull-up* untuk bus I²C, yaitu V_{ih} , V_{il} , *rise time* dan *fall time*. Nilai V_{ih} dan V_{il} digunakan untuk menentukan ruang solusi,

sedangkan *rise time* dan *fall time* digunakan untuk menentukan nilai spesifik resistor *pull-up* yang ada dalam ruang solusi. Langkah pertama yaitu menentukan nilai minimum resistor *pull-up* dengan persamaan sebagai berikut:

$$R_{P_{\min}} = \frac{V_{CC_{\max}} - V_{OL_{\min}}}{I_{OL_{\max}}} \quad (4-3)$$

Keterangan:

$R_{P_{\min}}$: nilai resistor *pull-up* minimum

$V_{CC_{\max}}$: nilai VCC maksimum dimana resistor *pull-up* akan dihubungkan

$V_{OL_{\min}}$: nilai $V_{OL_{\min}}$ mikrokontroler minimum

$I_{OL_{\max}}$: nilai $I_{OL_{\max}}$ maksimum saat nilai $V_{OL_{\min}}$ mikrokontroler maksimum

Berdasarkan *datasheet* diketahui bahwa $V_{OL_{\min}}$ bernilai 0 V, dan nilai $I_{OL_{\max}} = 3$ mA. Maka akan didapatkan nilai resistor *pull-up* minimum sebesar :

$$R_{P_{\min}} = \frac{V_{CC_{\max}} - V_{OL_{\min}}}{I_{OL_{\max}}}$$

$$R_{P_{\min}} = \frac{5,5V - 0V}{3 \times 10^{-3}A} = 1,83 \text{ k}\Omega$$

Langkah kedua yaitu menentukan nilai resistor *pull-up* maksimum dengan persamaan sebagai berikut:

$$R_{P_{\max}} = \frac{(V_{CC_{\min}} - (V_{ih_{\min}} + NM_{\min}))}{I_{ih_{\max}}} \quad (4-4)$$

Keterangan:

$R_{P_{\max}}$: nilai resistor *pull up* maksimum

$V_{CC_{\min}}$: nilai VCC minimum dimana resistor *pull up* akan dihubungkan

$V_{ih_{\min}}$: nilai $V_{ih_{\min}}$ dari setiap agen bus (dipilih yang paling besar) saat $V_{CC_{\min}}$

NM_{\min} : *noise margin*

$I_{ih_{\max}}$: nilai $I_{ih_{\max}}$ maksimum

Berdasarkan *datasheet* diketahui bahwa $V_{ih_{\min}}$ bernilai 0,7 $V_{CC_{\min}}$ atau bernilai 3,15V, nilai NM_{\min} yaitu 0,2, dan nilai $I_{ih_{\max}} = 1 \mu A$. Maka akan didapatkan nilai maksimum resistor *pull up* sebesar :

$$R_{P_{\max}} = \frac{(V_{CC_{\min}} - (V_{ih_{\min}} + NM_{\min}))}{I_{ih_{\max}}} = \frac{(4,5 - (3,15 + 0,2))}{1 \mu A}$$

$$R_{P_{\max}} = \frac{(1,15)}{1 \mu A} = 1,15 \text{ M}\Omega$$

Langkah terakhir yaitu menentukan nilai spesifik resistor *pull-up* yang ada dalam ruang solusi antara nilai maksimum dan nilai minimum. Persamaan untuk menentukan nilai spesifik resistor *pull up* ini adalah:

$$R_p = -t_r / (C \times \ln [(V_{ih_{min}} + NM_{min} - VCC_{min}) / (V_o - VCC_{min})]) \quad (4-5)$$

Keterangan:

t_r : nilai *rise time* maksimum setelah dikurangi margin (50 – 100ns)

C : nilai total kapasitansi bus

$V_{ih_{min}}$: nilai $V_{ih_{min}}$ dari setiap agen bus (dipilih yang paling besar) saat VCC_{max}

V_o : tegangan minimum saat logika 0

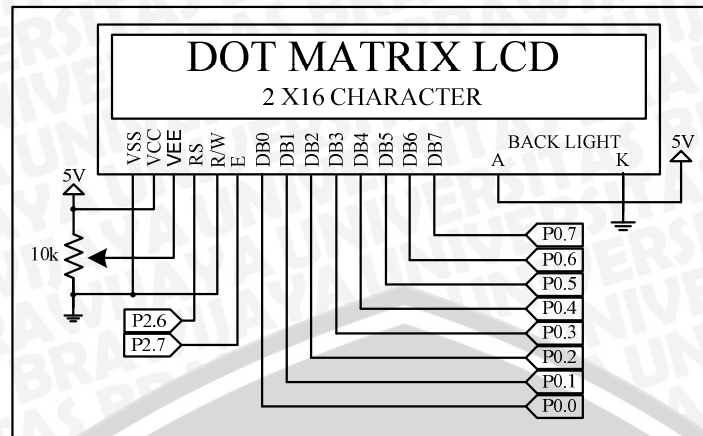
Berdasarkan *datasheet* diketahui bahwa nilai $t_{r_{max}}$ adalah 1000 ns, nilai C ($C_{i(MK)} + C_{i(RTC)} + C_{i(EEPROM)} + C_{i(ADC)}$) adalah 10pF+ 10pF+ 10pF +5pF, $V_{ih_{min}}$ bernilai 0,7 $VCC_{max} = 3,85$ V, VCC_{min} bernilai 4,5 V dan V_o bernilai 0 V. Maka akan didapatkan nilai spesifik resistor *pull-up* sebesar :

$$\begin{aligned} R_p &= -t_{r_{max}} / (C \cdot \ln [(V_{ih_{min}} + NM_{min} - VCC_{min}) / (V_o - VCC_{min})]) \\ &= -900 \times 10^{-9} / (35 \times 10^{-12} \cdot \ln [(3,85 + 0,2 - 4,5) / (0 - 4,5)]) \\ &= -900 \times 10^{-9} / (35 \times 10^{-12} \cdot \ln [0,1]) \\ &= -900 \times 10^{-9} / - 80,59 \times 10^{-12} \\ &= 11,17 \text{ k}\Omega \end{aligned}$$

Dalam perancangan ini digunakan resistor *pull-up* bernilai 10 k Ω karena lebih kecil dari 1,15 M Ω dan lebih besar dari 1,83 k Ω .

4.4.8. Perancangan Antarmuka Modul LCD

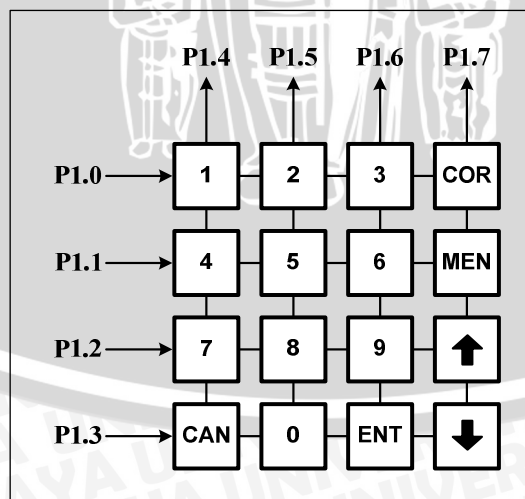
Penggunaan LCD dalam perancangan ini untuk menampilkan data berat, data harga serta informasi waktu. LCD yang digunakan adalah LCD dengan tipe M1632 (16x2). Bus data LCD (DB0-DB7) terhubung dengan port 0 mikrokontroler (P0.0-P0.7). LCD dioperasikan untuk menerima data, maka pin R/W dihubungkan dengan *ground*. Sedangkan pin RS dihubungkan dengan port 2.6 mikrokontroler dan pin E LCD dihubungkan ke port 2.7 mikrokontroler. Pengaturan tingkat kecerahan LCD dilakukan dengan mengubah resistor variabel 10 k Ω pada pin VEE LCD, sedangkan untuk mencatu lampu latar LCD, pin 15 (anoda) LCD dihubungkan ke catu daya 5V dan pin 16 (katoda) LCD dihubungkan ke *ground*. Antarmuka modul LCD ditunjukkan dalam Gambar 4.11.



Gambar 4.11 Antarmuka Modul LCD

4.4.9. Perancangan Antarmuka Keypad

Dalam perancangan ini keypad berfungsi sebagai media untuk memasukkan harga dan memilih menu yang akan ditampilkan ke LCD. Keypad yang digunakan adalah keypad matrik 4×4, keypad matrik ini bekerja menggunakan metode *scanning* pada lajur kolom dan lajur baris. Jika terdeteksi adanya persambungan antara baris dan kolom yang valid, maka mikrokontroler akan mengkodekan baris dan kolom yang aktif menjadi data *biner*. Apabila ada yang sesuai, maka mikrokontroler akan melakukan instruksi sesuai dengan data yang dimasukkan dari keypad. Untuk lajur baris B₀ - B₃ keypad dihubungkan ke port P1.0 - P1.3 mikrokontroler dan lajur kolom K₀ - K₃ keypad dihubungkan ke port P1.4 - P1.7 mikrokontroler. Antarmuka keypad ditunjukkan dalam Gambar 4.12.

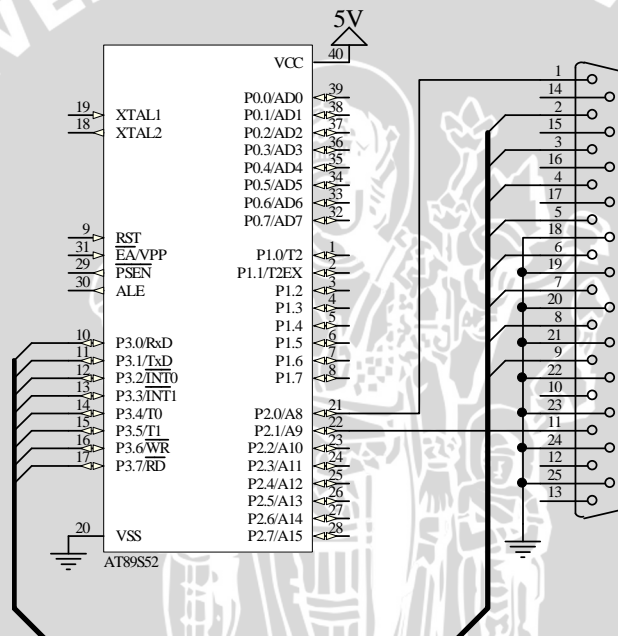


Gambar 4.12 Antarmuka Keypad

4.4.10. Perancangan Antarmuka Port Paralel

Alat ini menggunakan *printer* dot matrik yaitu Epson LX-800, dimana dalam proses penulisan pada kertas menggunakan *head* yang menumbuk pita tinta seperti halnya mesin ketik biasa, pita tersebut ditumbuk oleh jarum-jarum yang ada pada *head printer*. sehingga hasil tumbukan jarum ke pita inilah yang membekas pada kertas dan menghasilkan sebuah cetakan karakter.

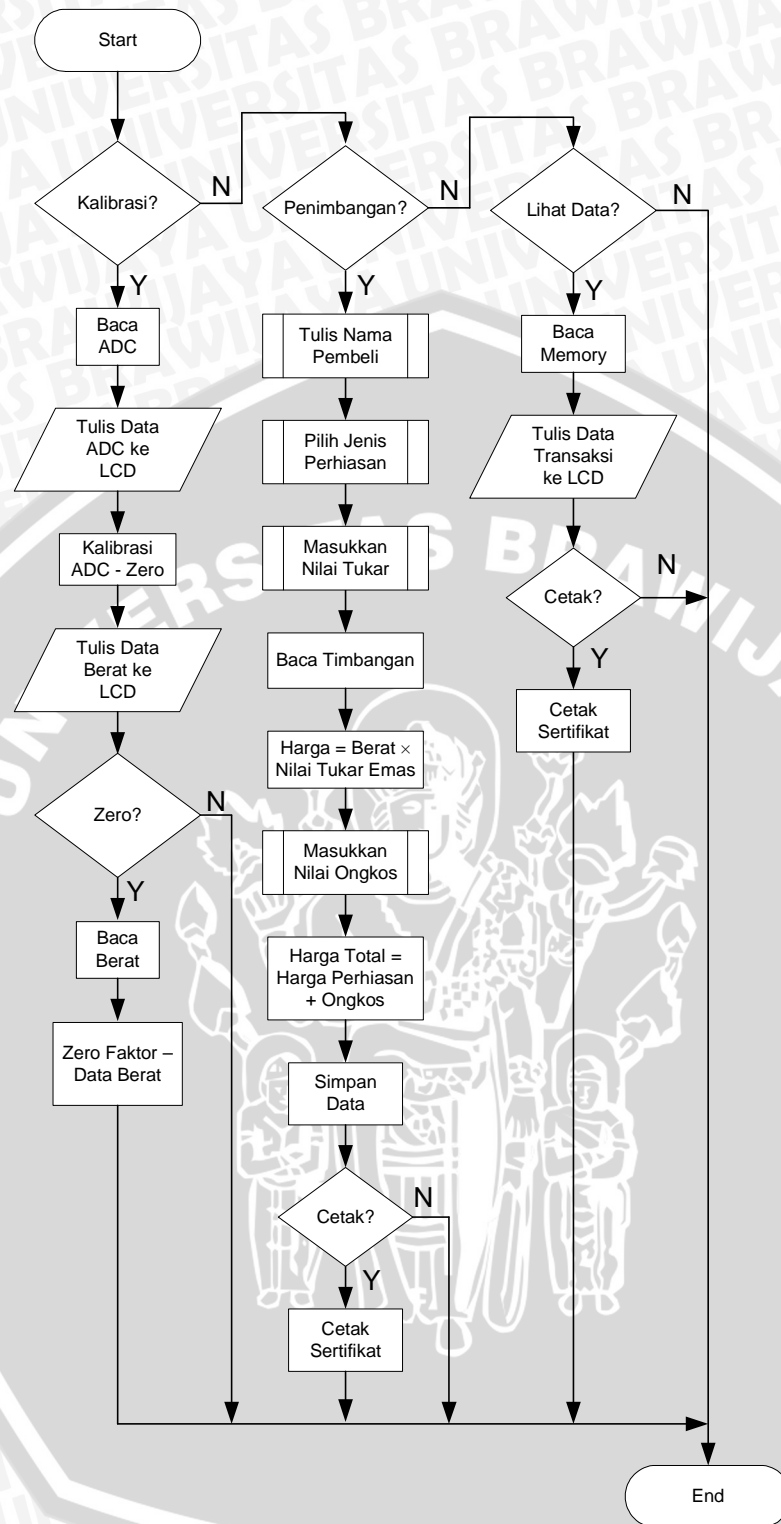
Port paralel dalam perancangan ini berfungsi sebagai antarmuka mikrokontroler dengan *printer*. Pin D0-D9 dari *printer* dihubungkan ke port P3.0-P3.9 mikrokontroler. Pin *strobe* (pin 1) dihubungkan ke port 2.1 mikrokontroler dan pin *busy* (pin 11) dihubungkan ke port 2.1 mikrokontroler. Antarmuka port paralel ditunjukkan dalam Gambar 4.13.



Gambar 4.13 Antarmuka Port Paralel

4.5 Perancangan Perangkat Lunak (*software*)

Perangkat lunak yang dirancang dengan menggunakan bahasa *assembly* mikrokontroler MCS-51. Untuk memberikan gambaran umum jalannya program dan memudahkan pembuatan perangkat lunak, maka dibuat diagram alir yang menunjukkan jalannya program. Diagram alir program utama ditunjukkan dalam Gambar 4.14.



Gambar 4.14 Diagram Alir Program Utama

Ketika catu daya diaktifkan, mikrokontroler akan melakukan *scan* keypad. *Scan* keypad ini berfungsi untuk memilih menu yang akan dijalankan. Menu yang ada dalam *software* mikrokontroler ini ada 3 macam, yaitu menu untuk kalibrasi, menu untuk penimbangan dan menu untuk melihat data.

Menu untuk kalibrasi berat perhiasan emas. Pada menu ini akan aktif jika ada penekanan tombol MEN pada keypad sebanyak 2 kali yang selanjutnya diikuti dengan penekanan tombol ENT. Kemudian ADC akan membaca data dari sensor. Selanjutnya MCU akan mengolah data dari ADC dan hasilnya ditampilkan ke LCD. Pada MCU, nilai ADC akan dikurangi dengan nilai zero dan hasilnya ditampilkan ke LCD. Selanjutnya apakah akan dilakukan zero (*set* data pada saat tidak ada beban LCD menunjukkan data 0,0 gram). Jika dilakukan zero maka akan dilakukan pembacaan ADC kembali. Pembacaan ADC yang kedua ini sebagai data berat sedangkan pembacaan ADC yang pertama sebagai zero faktor. Kemudian MCU akan melakukan pengurangan zero faktor dengan data berat. Sedangkan jika tidak dilakukan zero maka proses kalibrasi akan kembali ke awal atau berakhir.

Menu untuk penimbangan/pengukuran berat perhiasan emas. Pada menu ini akan aktif jika ada penekanan tombol ENT pada keypad. Kemudian *entry* data nama pembeli/konsumen, pemilihan jenis perhiasan emas yang akan ditimbang, *entry* nilai tukar perhiasan emas per gram dan meletakkan perhiasan emas yang akan ditimbang. Selanjutnya MCU akan mengalikan data berat penimbangan perhiasan emas dengan nilai tukar perhiasan emas. Kemudian *entry* biaya/ongkos pembuatan perhiasan. Setelah proses pengisian data selesai dilakukan, MCU akan menjumlahkan nilai harga perhiasan emas dengan biaya/ongkos pembuatan perhiasan. Hasil penjumlahan tersebut merupakan harga total perhiasan emas yang harus dibayar oleh pembeli/konsumen. Kemudian data hasil penimbangan tersebut disimpan ke dalam EEPROM. Selanjutnya apakah akan dilakukan pencetakan sertifikat perhiasan, jika dilakukan pencetakan, maka printer akan mencetak sertifikat. Sedangkan jika tidak dilakukan pencetakan sertifikat, maka proses akan berakhir atau kembali ke proses awal.

Menu untuk melihat data hasil penimbangan/pengukuran berat perhiasan emas. Pada menu ini akan aktif jika ada penekanan tombol COR sebanyak 2 kali dilanjutkan dengan penekanan tombol ENT pada keypad dan untuk mengakhiri dilakukan penekanan tombol CAN. Kemudian MCU akan membaca memori. Hasil pembacaan tersebut ditampilkan ke LCD. Selanjutnya apakah akan dilakukan pencetakan sertifikat perhiasan, jika dilakukan pencetakan, maka printer akan mencetak sertifikat. Sedangkan jika tidak dilakukan pencetakan sertifikat, maka proses akan berakhir atau kembali ke proses awalmenampilkan mode yang dipilih dan beban yang akan dinyalakan atau dimatikan ke LCD.

BAB V

PENGUJIAN DAN PEMBAHASAN

Bab ini akan membahas pengujian dan analisis alat yang telah dirancang dan direalisasikan. Pengujian dilakukan pada tiap-tiap blok dengan tujuan untuk mengamati apakah tiap blok rangkaian sudah sesuai dengan yang diharapkan baru kemudian dilanjutkan dengan pengujian keseluruhan sistem.

Pengujian yang dilakukan adalah:

- Pengujian antarmuka keypad,
- Pengujian antarmuka modul LCD,
- Pengujian rangkaian RTC,
- Pengujian rangkaian EEPROM,
- Pengujian rangkaian ADC,
- Pengujian rangkaian pengkondisi sinyal,
- Pengujian rangkaian port paralel,
- Pengujian sistem secara keseluruhan,

Perhitungan persentase kesalahan serta kesalahan rata-rata menggunakan persamaan sebagai berikut:

$$\% \text{Kesalahan (error)} = \left| \frac{\text{Perhitungan} - \text{Pengukuran}}{\text{Perhitungan}} \right| \times 100\% \quad (5-1)$$

$$\% \text{Kesalahan (error) rata-rata} = \frac{\text{Jumlah kesalahan (\%)}}{\text{banyaknya pengujian yang dilakukan}} \quad (5-2)$$

5.1 Pengujian Antarmuka Keypad

5.1.1. Tujuan Pengujian

Pengujian ini bertujuan untuk menganalisis bahwa keypad dapat digunakan sebagai media memilih menu dan *entry* harga.

5.1.2. Peralatan Pengujian

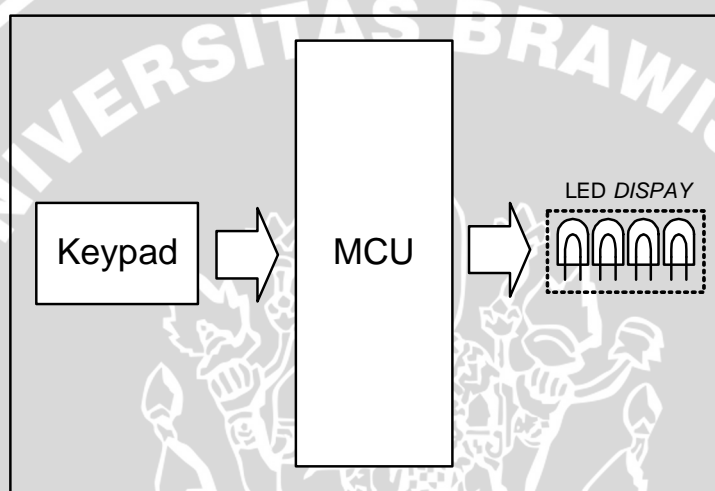
Peralatan pengujian yang diperlukan adalah sebagai berikut:

- Catu daya 5 volt,
- Minimum sistem mikrokontroler AT89S52,
- LED,
- Keypad 4×4.

5.1.3. Prosedur Pengujian

Prosedur pengujian yang dilakukan adalah sebagai berikut:

- Menyusun rangkaian pengujian antarmuka keypad sesuai dengan diagram blok pengujian. Diagram blok pengujian antarmuka keypad ditunjukkan dalam Gambar 5.1,
- Memberi catu daya pada rangkaian pengujian,
- Menekan tombol keypad satu per satu,
- Mengamati keluaran dari setiap penekanan tombol keypad melalui tampilan LED peraga.



Gambar 5.1 Diagram Blok Pengujian Antarmuka Keypad

Pengujian antarmuka keypad dilakukan dengan memberi logika pada lajur baris dan mengecek pada lajur kolom dengan tampilan LED peraga. Pada pengujian ini dilakukan penamaan untuk lajur baris diberikan nama B1, B2, B3, B4 sedangkan lajur kolom diberi nama K1, K2, K3, K4. Hal ini dilakukan untuk memudahkan dalam pengujian antarmuka keypad. Jika tombol '1' pada keypad ditekan, maka pada baris B1 dan kolom K1 berlogika 0 yang artinya lajur baris 1 diset logika 0 dan pada lajur kolom setelah discan menunjukkan logika 0 pada kolom 1 dengan nyala LED K1.

5.1.4. Hasil Pengujian

Hasil pengujian yang diperoleh adalah semua data penekanan tombol keypad dapat ditampilkan dengan benar melalui nyala LED peraga. Berdasarkan hasil pengujian dapat disimpulkan bahwa keypad dapat bekerja atau berfungsi dengan baik sesuai dengan perancangan. Hasil pengujian antarmuka keypad ditunjukkan dalam Tabel 5.1.

Tabel 5.1. Hasil Pengujian Antarmuka Keypad

Cek				Set				Routine	Keterangan
K4	K3	K2	K1	B4	B3	B2	B1		
1	1	1	1	1	1	1	0	Set Baris 1	
1	1	1	0	1	1	1	0	Cek Kolom	1
1	1	0	1	1	1	1	0	Cek Kolom	2
1	0	1	1	1	1	1	0	Cek Kolom	3
0	1	1	1	1	1	1	0	Cek Kolom	COR/13
1	1	1	1	1	1	0	1	Set Baris 2	
1	1	1	0	1	1	0	1	Cek Kolom	4
1	1	0	1	1	1	0	1	Cek Kolom	5
1	0	1	1	1	1	0	1	Cek Kolom	6
0	1	1	1	1	1	0	1	Cek Kolom	MEN/14
1	1	1	1	1	0	1	1	Set Baris 3	
1	1	1	0	1	0	1	1	Cek Kolom	7
1	1	0	1	1	0	1	1	Cek Kolom	8
1	0	1	1	1	0	1	1	Cek Kolom	9
0	1	1	1	1	0	1	1	Cek Kolom	↑/15
1	1	1	1	0	1	1	1	Set Baris 4	
1	1	1	0	0	1	1	1	Cek Kolom	CAN/11
1	1	0	1	0	1	1	1	Cek Kolom	0
1	0	1	1	0	1	1	1	Cek Kolom	ENT/12
0	1	1	1	0	1	1	1	Cek Kolom	↓/16

5.2 Pengujian Antarmuka Modul LCD

5.2.1. Tujuan Pengujian

Pengujian ini bertujuan untuk menganalisis *port 0* mikrokontroler dapat mengirim data ke LCD sesuai dengan program yang dibuat dan untuk mengetahui keberhasilan LCD dalam menampilkan data karakter.

5.2.2. Peralatan Pengujian

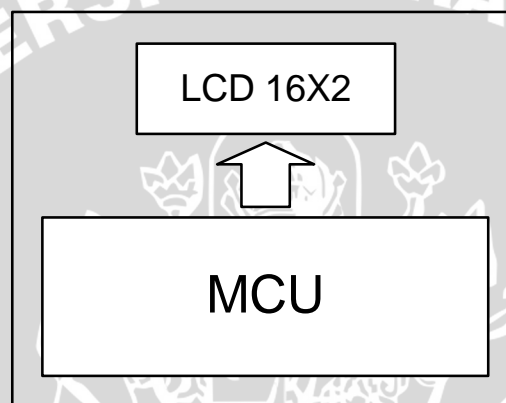
Peralatan pengujian yang diperlukan adalah sebagai berikut:

- Catu daya +5 volt,
- Keypad 4×4,
- Minimum sistem mikrokontroler AT89S52,
- Empat buah LED

5.2.3. Prosedur Pengujian

Prosedur pengujian yang dilakukan adalah sebagai berikut:

- Menyusun rangkaian pengujian sesuai dengan diagram blok pengujian. Diagram blok pengujian antarmuka modul LCD ditunjukkan dalam Gambar 5.2,
- Memberikan catu daya pada rangkaian pengujian,
- Membuat program untuk menampilkan data karakter huruf “DWI JOKOWALOYO” pada baris pertama dan karakter angka “0810632005” pada baris kedua pada LCD,
- Mengamati data karakter yang ditampilkan pada LCD.



Gambar 5.2 Diagram Blok Pengujian Antarmuka Modul LCD

5.2.4. Hasil Pengujian

Pada layar LCD tampil huruf “DWI JOKOWALOYO” pada baris pertama dan tampil angka “0810632005” pada baris kedua. Dari hasil pengujian dapat disimpulkan bahwa LCD dapat menampilkan data karakter dengan baik. Hasil pengujian antarmuka modul LCD ditunjukkan dalam Gambar 5.3.



Gambar 5.3 Hasil Pengujian Antarmuka Modul LCD

5.3 Pengujian Rangkaian RTC

5.3.1. Tujuan Pengujian

Pengujian ini bertujuan untuk menganalisis keberhasilan RTC dalam membaca dan mengirim data pada mikrokontroler.

5.3.2. Peralatan Pengujian

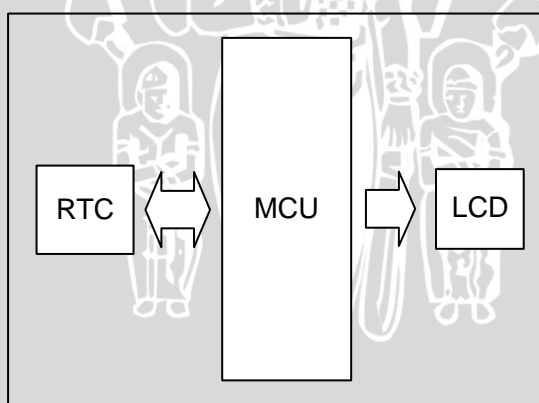
Peralatan pengujian yang diperlukan adalah sebagai berikut:

- Catu daya +5 volt,
- Minimum sistem mikrokontroler AT89S52,
- RTC DS1307,
- LCD 16×2,
- Jam Digital.

5.3.3. Prosedur Pengujian

Langkah-langkah dalam pengujian rangkaian ini adalah:

- Menyusun rangkaian pengujian sesuai dengan diagram blok pengujian rangkaian RTC. Diagram blok pengujian rangkaian RTC ditunjukkan dalam Gambar 5.4,
- Memberikan catu daya pada rangkaian pengujian,
- Membuat program pengujian pada mikrokontroler,
- Menjalankan *software* uji coba,
- Mengamati informasi waktu yang ditampilkan pada LCD dan membandingkan dengan informasi waktu pada jam tangan digital.



Gambar 5.4 Diagram Blok Pengujian Rangkaian RTC

5.3.4. Hasil Pengujian

Informasi waktu yang ditampilkan pada *display* LCD dibandingkan dengan informasi waktu jam tangan digital. Dari hasil pengujian menunjukkan informasi waktu dari RTC yang ditampilkan pada LCD sesuai dengan informasi waktu pada jam tangan digital, sehingga dapat disimpulkan bahwa rangkaian RTC dapat bekerja sesuai dengan perancangan. Hasil pengujian rangkaian RTC ditunjukkan dalam Gambar 5.5. Tabel hasil pengujian rangkaian RTC ditunjukkan dalam Tabel 5.2.

Tabel 5.2 Hasil Pengujian Rangkaian RTC

No.	Jam tangan digital	Jam pada LCD
1	11:37:39	11:37:39
2	11:38:37	11:38:37
3	11:39:09	11:39:09
4	11:40:12	11:40:12
5	11:51:23	11:51:23



Gambar 5.5 Hasil Pengujian Rangkaian RTC

5.4 Pengujian Rangkaian EEPROM

5.4.1 Tujuan Pengujian

Pengujian ini dilakukan untuk menganalisis memori agar dapat digunakan untuk menyimpan *database* hasil penimbangan melalui proses tulis dan baca pada alamat memori tertentu.

5.4.2 Peralatan Pengujian

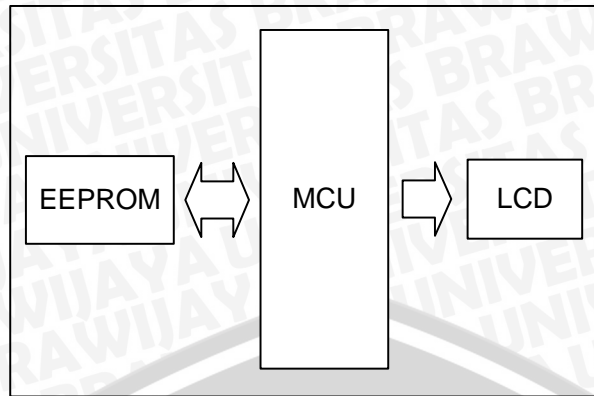
Peralatan pengujian yang diperlukan adalah sebagai berikut:

- Catu daya +5 volt,
- EEPROM AT24C16,
- Minimum sistem mikrokontroler AT89S52,
- LCD 16x2.

5.4.3 Prosedur Pengujian

Langkah-langkah dalam pengujian rangkaian ini adalah:

- Menyusun rangkaian pengujian sesuai dengan diagram blok pengujian rangkaian EEPROM. Diagram blok pengujian rangkaian EEPROM ditunjukkan dalam Gambar 5.6,
- Memberi catu daya pada rangkaian pengujian,
- Membuat program pengujian rangkaian EEPROM,
- Menjalankan program pengujian,
- Mengamati data yang ditampilkan pada LCD.



Gambar 5.6 Diagram Blok Pengujian Rangkaian EEPROM

5.4.4 Hasil Pengujian

Data hasil pengujian rangkaian EEPROM ditunjukkan dalam Tabel 5.3

Tabel 5.3 Data Hasil Pengujian rangkaian EEPROM

No	Tulis EEPROM		Baca EEPROM	
	Alamat	Data	Alamat	Data
1	00	a	00	a
2	01	d	01	d
3	02	m	02	m
4	03	b	03	b
5	04	s	04	s

Dalam pengujian ini, data yang dibaca dari memori diamati melalui tampilan LCD. Setiap data yang dituliskan pada alamat memori tertentu dapat dibaca kembali dengan benar sesuai dengan data yang telah dituliskan pada alamat memori tersebut. Berdasarkan hasil pengujian, memori dapat digunakan untuk menyimpan data pada alamat tertentu dan data dapat dibaca kembali, sehingga memori dapat digunakan untuk menyimpan data transaksi.

5.5 Pengujian Rangkaian ADC

5.5.1 Tujuan Pengujian

Pengujian rangkaian ADC bertujuan untuk menganalisis level tegangan keluaran ADC, menguji kepresisian serta kelinieran ADC dalam mengkonversi tegangan analog ke dalam nilai digital 8 bit yang ekuivalen.

5.5.2 Peralatan Pengujian

Peralatan pengujian yang diperlukan adalah sebagai berikut:

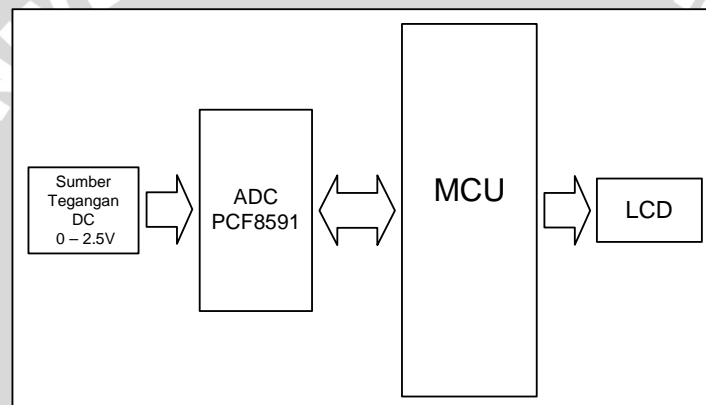
- Catu daya +5 volt,
- Sumber tegangan DC variabel 0 – 2,5 volt,
- ADC PCF8591,
- Minimum sistem mikrokontroler AT89S52,

- LCD 16×2,
- Multimeter Digital.

5.5.3 Prosedur Pengujian

Langkah-langkah dalam pengujian rangkaian ini adalah:

- Menyusun rangkaian pengujian sesuai dengan diagram blok pengujian ADC. Diagram blok pengujian ADC ditunjukkan dalam Gambar 5.7,
- Memberi catu daya pada rangkaian pengujian,
- Memberi masukan tegangan analog secara bergantian dari tegangan 0 – 2,5 volt pada pin AIN0 ADC,
- Mengukur setiap tegangan analog yang berbeda yang akan dikonversi ADC,
- Mencatat setiap keluaran ADC dari tampilan LCD.



Gambar 5.7 Diagram Blok Pengujian Rangkaian ADC

5.5.4 Hasil Pengujian

Data hasil pengujian rangkaian ADC ditunjukkan dalam Tabel 5.4.

Tabel 5.4 Hasil Pengujian Rangkaian ADC

Vin (volt)	Output Perhitungan		Output Pengukuran		Kesalahan %
	DEC	HEX	DEC	HEX	
0.25	26	1A	27	1B	3.85
0.5	51	33	50	32	1.96
0.75	77	4D	78	4E	1.30
1	102	66	103	67	0.98
1.25	128	80	127	7F	0.78
1.5	153	99	154	9A	0.65
1.75	179	B3	180	B4	0.56
2	204	CC	205	CD	0.49
2.25	230	E6	229	E5	0.43
2.5	255	FF	256	FF	0.39
Kesalahan rata-rata %					1.14

Dari Tabel 5.4 dapat diketahui bahwa ADC PCF8591 mampu mengkonversi masukan tegangan analog dari 0 V sampai 2,5 V menjadi data dalam bentuk heksadesimal. Nilai kesalahan menunjukkan nilai penyimpangan data digital keluaran ADC dengan nilai yang sebenarnya. Nilai kesalahan maksimum berdasarkan *datasheet* yang diberikan ke ADC PCF8591 adalah sebesar 1,5 LSB. Perhitungan nilai desimal pada saat $V_{in}=1,5$ V, keluaran ADC adalah sebagai berikut:

$$\begin{aligned}\text{Nilai keluaran (desimal)} &= \frac{V_{in}}{V_{ref}} \times 2^n \\ &= \frac{1,5V}{2,5V} \times 2^8 = 153(\text{desimal}) = 10011001(\text{biner}) \\ &= 99(\text{heksadesimal})\end{aligned}$$

Berdasarkan Tabel 5.4 dapat dilihat bahwa persentase kesalahan terbesar adalah 3.85%. Sedangkan kesalahan rata-rata adalah 1.14%. Dengan kesalahan rata-rata kurang dari 5%, maka ADC PCF8591 ini dapat dikatakan bekerja dengan baik sesuai dengan aplikasi yang direncanakan.

5.6 Pengujian Rangkaian Pengkondisi Sinyal

5.6.1 Tujuan Pengujian

Pengujian ini bertujuan untuk menganalisis ketepatan rangkaian pengkondisi sinyal sensor berat dalam merespon perubahan berat.

5.6.2 Peralatan Pengujian

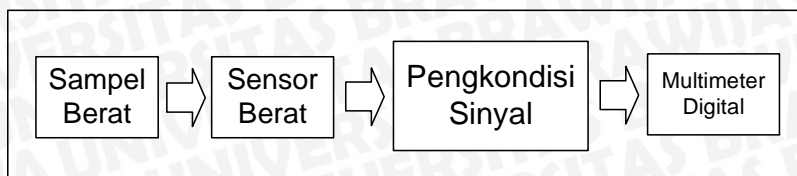
Peralatan pengujian yang diperlukan adalah sebagai berikut:

- Catu daya +5 volt,
- Sampel berat,
- Sensor berat *load cell*,
- Rangkaian pengkondisi sinyal,
- Multimeter digital.

5.6.3 Prosedur Pengujian

Prosedur pengujian yang dilakukan adalah sebagai berikut:

- Menyusun rangkaian sesuai dengan diagram blok pengujian. Diagram blok pengujian rangkaian pengkondisi sinyal ditunjukkan dalam Gambar 5.8,
- Memberi catu daya pada rangkaian,
- Memberikan sampel berat yang berbeda-beda pada sensor berat,
- Mengukur keluaran pengkondisi sinyal dengan multimeter,
- Mengamati dan mencatat data yang terukur pada multimeter.



Gambar 5.8 Diagram Blok Pengujian Rangkaian Pengkondisi Sinyal

5.6.4 Hasil Pengujian

Hasil pengujian rangkaian pengkondisi sinyal sensor berat ditunjukkan dalam Tabel 5.5.

Tabel 5.5 Hasil Pengujian Rangkaian Pengkondisi Sinyal Sensor Berat

Berat (gram)	Selisih V_{in} (mV)	Penguatan (Gain)	V_{out} (V)		% Kesalahan (Error)
			Teori	Praktik	
0.5	0.03	2777.78	0.08	0.07	12.40
1	0.06	2777.78	0.17	0.15	10.00
2	0.12	2777.78	0.33	0.32	4.00
3.5	0.21	2777.78	0.58	0.57	2.29
5	0.30	2777.78	0.83	0.81	2.80
6.5	0.39	2777.78	1.08	1.06	2.15
8	0.48	2777.78	1.33	1.31	1.75
10	0.60	2777.78	1.67	1.64	1.60
11	0.66	2777.78	1.83	1.81	1.27
12	0.72	2777.78	2.00	2.02	1.00
13	0.78	2777.78	2.17	2.15	0.68
15	0.90	2777.78	2.50	2.51	0.40
Kesalahan rata-rata					3.36

Analisis dilakukan dengan menghitung persentase kesalahan hasil pengukuran dengan hasil perhitungan menggunakan Persamaan (5-1) dan Persamaan (5-2). Berdasarkan Tabel 5.5 dapat dilihat bahwa persentase kesalahan terbesar adalah 12,40%. Sedangkan kesalahan rata-rata pada pengujian ini adalah 3,36%. Dengan kesalahan rata-rata kurang dari 5% maka rangkaian pengkondisi sinyal sensor berat dapat dikatakan bekerja dengan baik sesuai dengan aplikasi yang direncanakan.

- Analisis untuk berat sebesar 15 gram

Berdasarkan perhitungan diketahui $A_v(\text{gain}) = 2777,78$, $S_+ = 2,50045V$, $S_- = 2,49955V$. Maka akan didapatkan nilai tegangan keluaran sebesar:

$$V_{out} = (V_1 - V_2) \left(1 + \frac{2}{\alpha}\right)$$

$$V_{out} = (2,50045V - 2,49955V)(2777,78)$$

$$V_{\text{out}} = (0,9\text{mV}) \times 2777,78$$

$$V_{\text{out}} = 2,5 \text{ V}$$

Sedangkan untuk mendapatkan nilai persentase pada berat 15 gram dilakukan perhitungan sebagai berikut:

$$\begin{aligned} \% \text{ Kesalahan (error)} &= \left| \frac{\text{Hasil Perhitungan} - \text{Hasil Pengukuran}}{\text{Hasil Perhitungan}} \right| \times 100\% \\ &= \left| \frac{2,50 - 2,51}{2,50} \right| \times 100\% = 0,40\% \end{aligned}$$

5.7 Pengujian Port Paralel

5.7.1. Tujuan Pengujian

Pengujian ini bertujuan untuk menganalisis keberhasilan port paralel sebagai media antarmuka mikrokontroler dengan *printer*.

5.7.2. Peralatan Pengujian

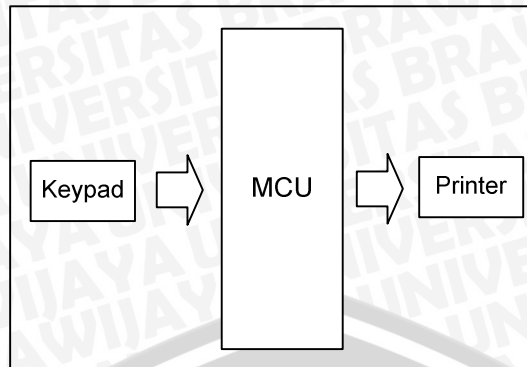
Peralatan pengujian yang diperlukan adalah sebagai berikut:

- Minimum sistem mikrokontroler AT89S52,
- Keypad 4×4,
- Konektor DB25,
- *Printer*,
- Kertas,
- Catu daya 5V dan 220V.

5.7.3. Prosedur Pengujian

Prosedur pengujian yang dilakukan adalah sebagai berikut:

- Menyusun rangkaian sesuai dengan diagram blok pengujian. Diagram blok pengujian antarmuka port paralel ditunjukkan dalam Gambar 5.9,
- Memberi catu daya pada rangkaian pengujian,
- Memasang kertas pada printer,
- Menghubungkan port P3.0-P3.7 mikrokontroler dengan port paralel pin 2 – pin 9, port P2.1 mikrokontroler dengan pin 1 (*strobe*) dan port P2.2 mikrokontroler dengan pin 11 (*busy*),
- Menghubungkan port P1.0-P1.7 mikrokontroler dengan keypad,
- Memasukkan data karakter ASCII ke mikrokontroler melalui keypad,
- Mengamati data yang tercetak pada kertas hasil pencetakan dari *printer*,



Gambar 5.9 Diagram Blok Pengujian Antarmuka Port Paralel

5.7.4. Hasil Pengujian

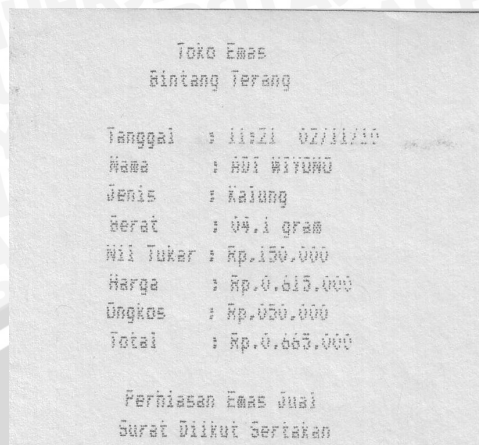
Hasil dari pengujian antarmuka port paralel ditunjukkan pada Tabel 5.6. Data dimasukkan dengan keypad memakai kode ASCII ke mikrokontroler dan mikrokontroler memberi sinyal *strobe*, kemudian data dari mikrokontroler dikirim ke *printer* melalui port paralel. Data karakter yang dikirim oleh mikrokontroler tersebut disimpan dan diterima oleh *printer*. Sedangkan jika diberikan data 0Dh, maka *printer* akan mencetak karakter yang telah disimpan. Dan jika diberikan data 0Ah maka *printer* meletakkan hasil cetakan pada baris selanjutnya. Saat *printer* mencetak data, *printer* juga mengirim sinyal *busy* ke mikrokontroler.

Tabel 5.6 Hasil Pengujian Antarmuka Port Paralel

Data	Pin Strobe	Keterangan
41h		Simpan Karakter 'A'
42h		Simpan Karakter 'B'
43h		Simpan Karakter 'C'
44h		Simpan Karakter 'D'
0Dh		Cetak ABCD
0Ah		Line Feed

Hasil *print-out* dari penimbangan perhiasan emas ditunjukkan dalam Gambar 5.10. Hasil *print-out* ini menyertakan tanggal dan waktu penimbangan, nama pembeli/konsumen, jenis perhiasan emas, berat perhiasan emas, nilai tukar perhiasan

emas, harga dari perhiasan emas, ongkos/biaya pembuatan perhiasan emas, total harga perhiasan emas.



Gambar 5.10 Sertifikat Perhiasan

5.8 Pengujian Sistem secara Keseluruhan

5.8.1. Tujuan Pengujian

Pengujian ini bertujuan untuk menganalisis apakah sistem yang telah dibuat mampu bekerja sesuai dengan perancangan baik perancangan perangkat keras (*hardware*) maupun perangkat lunak (*software*).

5.8.2. Peralatan Pengujian

Peralatan pengujian yang diperlukan adalah sebagai berikut:

- Sumber tegangan AC 220V,
- Timbangan digital,
- Timbangan Alat,
- Sampel berat,
- Printer,
- Bahan sertifikat (kertas sertifikat)

5.8.3. Prosedur Pengujian

Prosedur pengujian yang dilakukan adalah sebagai berikut:

- Menghubungkan timbangan alat dengan printer,
- Memberi catu daya pada timbangan alat dan printer,
- Menjalankan perangkat lunak yang telah dibuat,
- Melakukan penimbangan dengan member sampel berat,
- Mengamati dan mencatat data hasil penimbangan.

5.8.4. Hasil Pengujian

Data hasil pengujian ditunjukkan dalam Tabel 5.7.

Tabel 5.7 Data Hasil Pengujian Timbangan Alat secara keseluruhan

No.	Berat (gram)	Timbangan Digital (gram)	Timbangan Alat (gram)	Kesalahan (%)
1	0.5	0.5	0.4	20.00
2	1	1.1	1	9.09
3	2	2	1.9	5.00
4	3.5	3.5	3.4	2.86
5	5	5	4.9	2.00
6	6.5	6.6	6.7	1.52
7	8	8.1	8.2	1.23
8	10	10	9.9	1.00
9	11	10.9	11	0.92
10	12	12.1	12.2	0.83
11	13	13.2	13.3	0.76
12	15	15	15.1	0.67
Kesalahan rata-rata (%)				3.82

Analisis dilakukan dengan menghitung persentase kesalahan hasil penimbangan menggunakan timbangan digital dengan hasil penimbangan menggunakan timbangan alat. Persentase kesalahan terbesar dari pengujian alat secara keseluruhan dapat diperoleh dengan perhitungan sebagai berikut:

$$\begin{aligned} \% \text{ Kesalahan} &= (|\text{Timbangan Digital} - \text{Timbangan Alat}| / \text{Timbangan Digital}) \times 100\% \\ &= (|0,5 - 0,4| / 0,5) \times 100\% \\ &= 20\% \end{aligned}$$

Sedangkan persentase kesalahan rata-rata diperoleh dengan perhitungan sebagai berikut:

$$\begin{aligned} \% \text{ Kesalahan rata-rata} &= \sum \% \text{ Kesalahan} / \text{Banyaknya Pengujian} \\ &= (45,84\%) / 12 \\ &= 3,82\% \end{aligned}$$

Berdasarkan data hasil pengujian secara keseluruhan dapat diambil kesimpulan bahwa timbangan alat berfungsi dengan baik sesuai dengan perancangan. Timbangan alat mampu mengukur berat perhiasan emas, mampu menghitung jumlah total harga perhiasan emas dan mampu mencetak sertifikat perhiasan.

BAB VI

PENUTUP

6.1. Kesimpulan

Berdasarkan pengujian yang telah dilakukan baik pengujian per blok rangkaian maupun pengujian sistem secara keseluruhan dapat diambil kesimpulan sebagai berikut:

- 1) Dalam penelitian ini, rangkaian pengkondisi sinyal dibentuk dari tiga buah op-amp dan beberapa tahanan yang membentuk sebuah rangkaian penguat instrumentasi. Hasil dari pengujian rangkaian pengkondisi sinyal diperoleh nilai persentase kesalahan terkecil sebesar 0,4% dan kesalahan terbesar sebesar 12,4% serta nilai kesalahan rata-rata sebesar 3,36%,
- 2) Sertifikat perhiasan dicetak menggunakan *printer* jenis dot matrik. *Printer* jenis ini tidak memerlukan sebuah *driver*, sedangkan sebagai pengendali kinerja *printer*, digunakan rangkaian mikrokontroler. Dalam hal ini, komunikasi data mikrokontroler dengan *printer* menggunakan komunikasi data paralel. Sedangkan penggunaan kertas sertifikat digunakan kertas HVS dengan ukuran lebar 6 cm,
- 3) Dalam pengujian alat secara keseluruhan diperoleh nilai persentase kesalahan terkecil sebesar 0,67% pada berat 15 gram dan kesalahan terbesar sebesar 20% pada berat 0,5 gram serta nilai kesalahan rata-rata sebesar 3,82%.

6.2. Saran

Dalam pengembangan alat lebih lanjut perlu diperhatikan hal-hal sebagai berikut:

- 1) Penggunaan kertas sertifikat sebaiknya menggunakan kertas yang ada simbol atau logo dari toko perhiasan emas yang bersangkutan,
- 2) Penggunaan ADC yang lebih tinggi (jumlah bit), akan diperoleh tingkat ketelitian yang lebih baik,
- 3) Alat pencetak sertifikat perhiasan dapat digantikan dengan *printer* yang lebih kecil agar lebih *portable*.



DAFTAR PUSTAKA

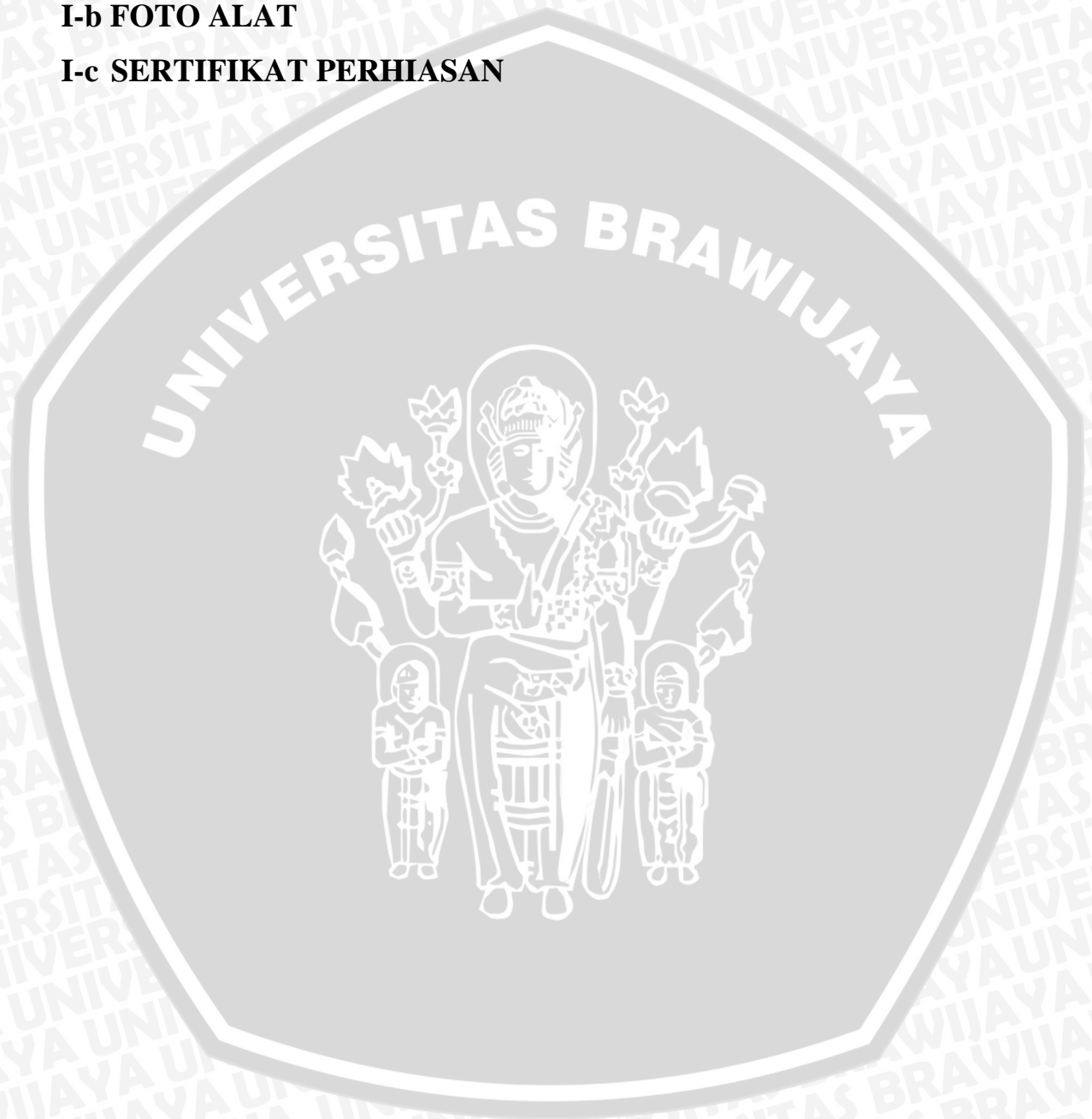
- Anonymous, "M1632 MODULE LCD 16X2 (M1632)",
<http://www.delta-electronic.com> (diakses 10 September 2007)
- Atmel. 2001. "Datasheet 8-bit Mikrokontroler With 8K Byte AT89S52". Atmel Inc.
<http://www.atmel.com>. (diakses 27 januari 2003)
- Budiharto, Widodo; Firmansyah, Sigit. 2005. *Elektronika Digital dan Mikroprosesor*. Andi Offset : Yogyakarta.
- Cooper, William D, 1999, *Instrumentasi Elektronika dan Pengukuran*, Edisi Kedua, diterjemahkan oleh Ir. Sahat Pakpahan, Erlangga, Jakarta.
- Coughlin, R. F. & Frederick F. Driscoll. 1985. *Penguat Operasional Dan Rangkaian Terpadu Linier*. (Herman Widodo Soemitro, Trans). Jakarta: Erlangga
- Maxim. 2008. "Datasheet DS1307 64 x 8, Serial, I2C Real-Time Clock". USA: Maxim Integrated Product, Inc.
<http://www.maxim-ic.com> (diakses 27 januari 2003)
- Moh. Ibnu Malik. 1997. *Bereksperimen dengan Mikrokontroler 8031*. Jakarta: PT Elex Media Komputindo
- Nalwan, Paulus Andi. 2003. *Teknik Antarmuka dan Pemrograman Mikrokontroler AT89S51*. Jakarta: Elex Media Komputindo.
- National Semiconductor. 2000. "Datasheet LM124/LM224/LM324/LM2902 Low Power Quad Operational Amplifiers". National Semiconductor Corp.
<http://www.national.com> (diakses 8 juni 2010)
- Philips. 2003. Koninklijke Philips Electronics: PCF8591 8-bit A/D and D/A Converter. Netherlands
<http://www.semiconductors.philips.com>. (diakses 27 januari 2003)
- Seatronics, 2005, "Epson LX-800",
<http://www.rentit.biz/specs/pdf/lx800.pdf>, tanggal akses 10 Februari 2008
- Standart Keypad. USA: Grayhill inc.
<http://www.grayhill.com> (diakses 27 januari 2003)
- SMBus and I²C Bus Design*
<http://www.semiconductors.philips.com/i2c> (diakses 4 mei 2010)

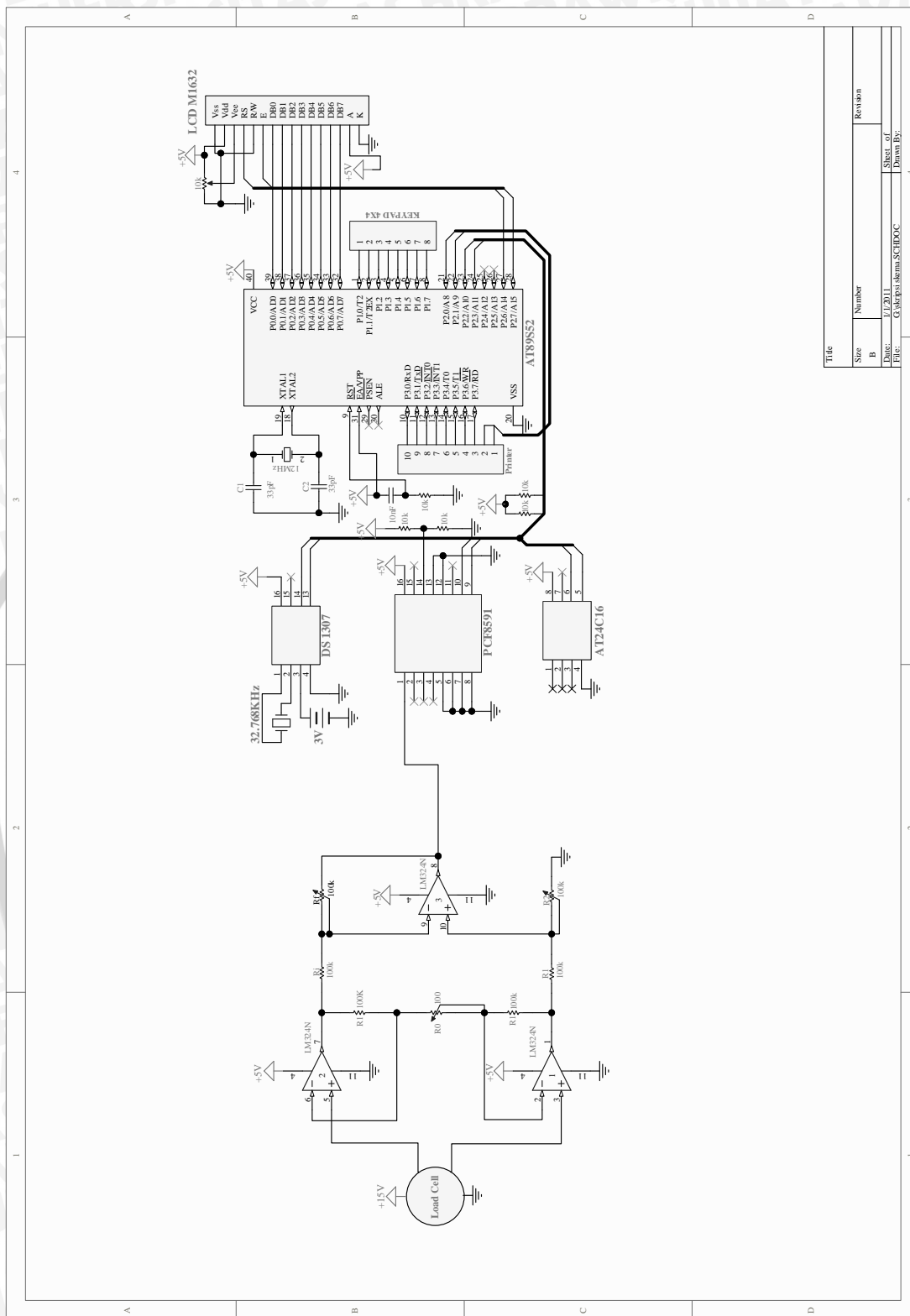
PERANGKAT KERAS

I-a SKEMA RANGKAIAN KESELURUHAN

I-b FOTO ALAT

I-c SERTIFIKAT PERHIASAN





Title	
Size	Number
B	
Date:	Sheet of
File:	Drawn By:
	Revision

I-a SKEMA RANGKAIAN KESELURUHAN





I-b FOTO ALAT

Toko Emas
Bintang Terang

Tanggal : 11/21 02/11/10
Nama : ARI WIYONO
Jenis : Kalung
Berat : 04.1 gram
Nil Tukar : Rp.150.000
Harga : Rp.0.615.000
Ungkos : Rp.050.000
Total : Rp.0.665.000

Perhiasan Emas Asli
Swarit Dilikut Seritakan

I-c SERTIFIKAT PERHIASAN

PERANGKAT LUNAK

II-a *Flowchart* Pemilihan Jenis Perhiasan Emas

II-b *Flowchart* Kalibrasi

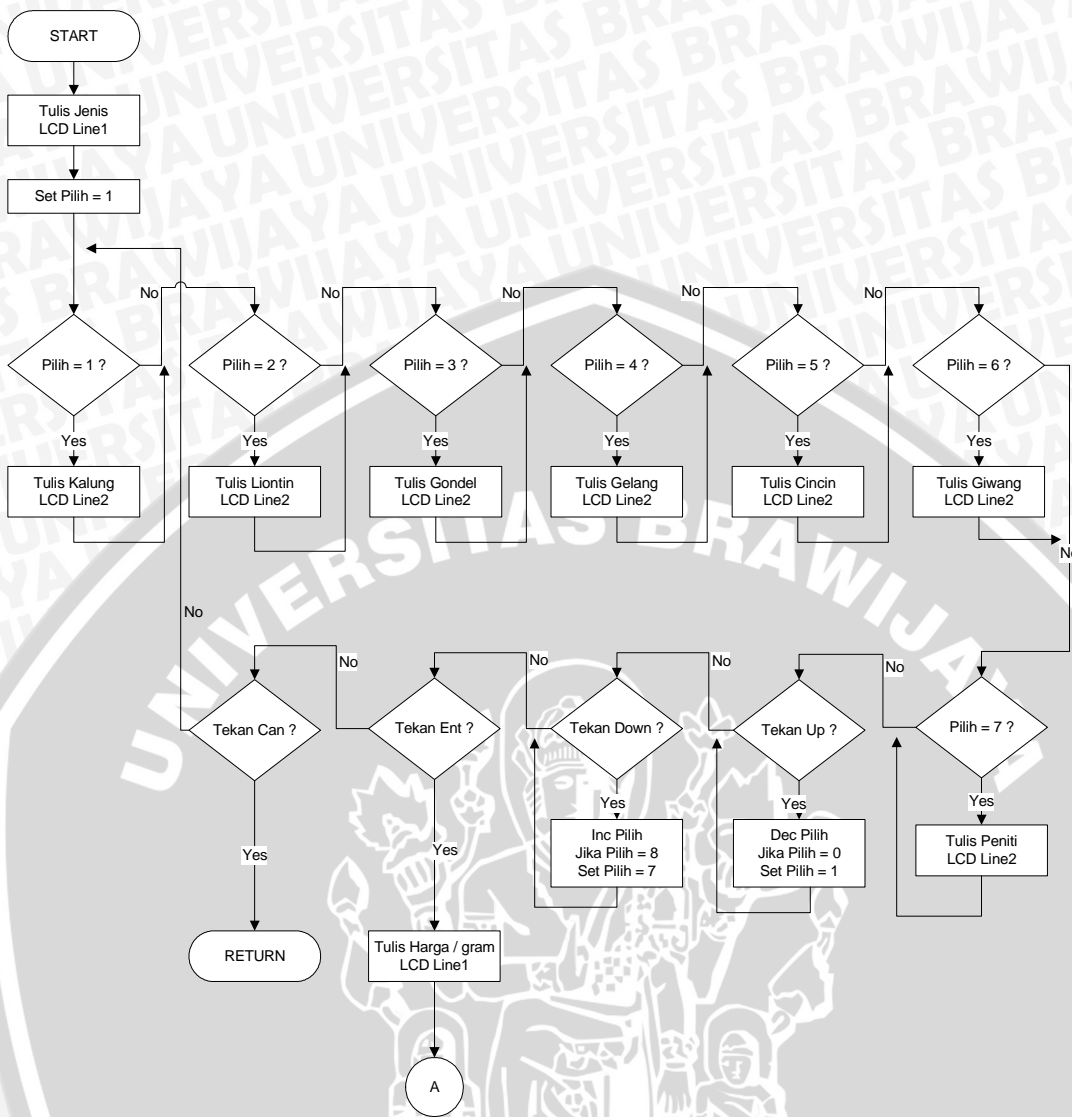
II-c *Flowchart* Entry Nama

II-d *Flowchart* Entry Nilai Tukar

II-e *Flowchart* Pengukuran Berat

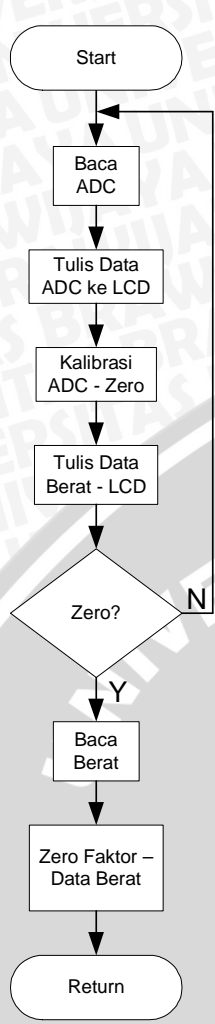
II-f *Listing* Program



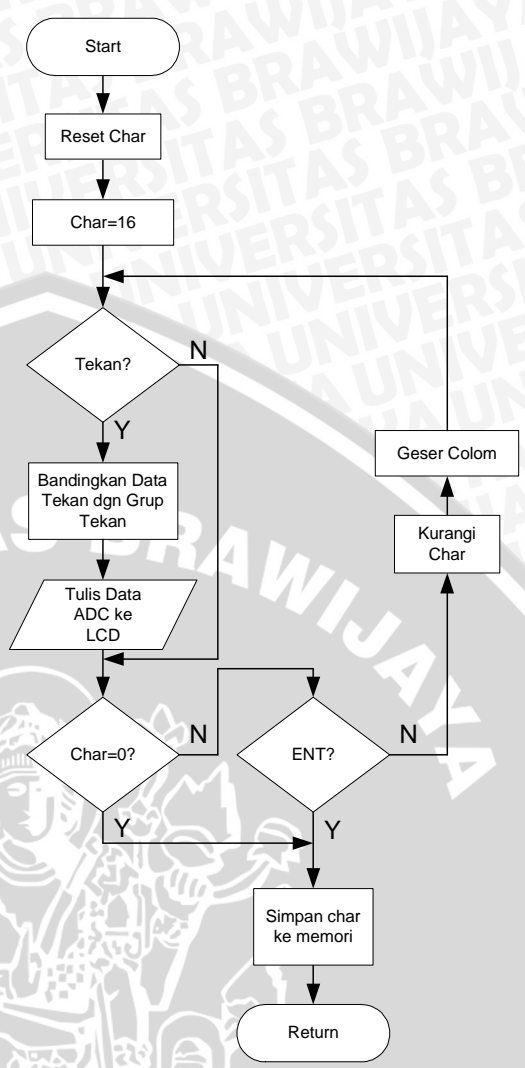


II-a Flowchart Pemilihan Jenis Perhiasan Emas



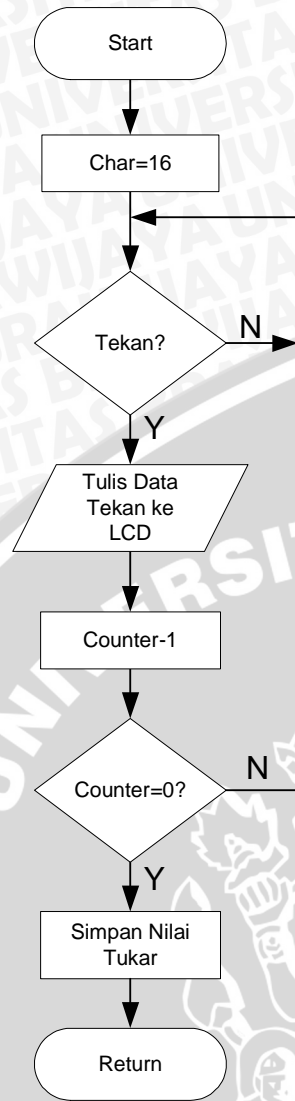


II-b Flowchart Kalibrasi

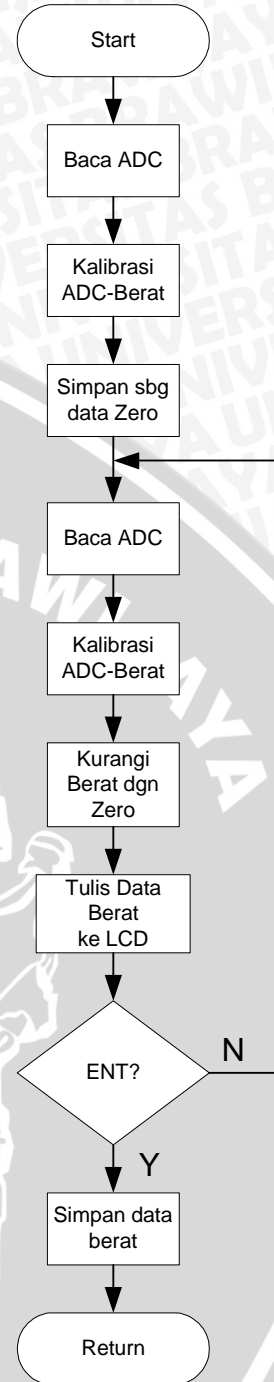


II-c Flowchart Entry Nama





II-d Flowchart entry Nilai Tukar



II-e Flowchart Pengukuran

II-g LISTING PROGRAM

```

org 00h ; program dimulai pada alamat 00h
;
ISDA Bit P2.0 ; P2C data ADC, RTC,Memory
ISCL Bit P2.1 ; P2C clock ADC, RTC, Memory
Strb Bit P2.2 ; Strobe Printer
Busi Bit P2.3 ; Busy Printer
Rest Bit P2.6 ; RS LCD
Enbl Bit P2.7 ; E LCD
Stch Bit 20h.0 ; status tulis character
;
Djam Equ 30h ; data jam
Dmnt Equ 31h ; data menit
Ddtk Equ 32h ; data detik
Dtgl Equ 33h ; data tanggal
Dbln Equ 34h ; data bulan
Dcen Equ 35h ; data century
Dthn Equ 36h ; data tahun
;-----
Bufr Equ 37h ; buffer
Bf00 Equ 38h ;\
Bf01 Equ 39h ;|
Bf02 Equ 3Ah ;| buffer
Bf03 Equ 3Bh ;|
Bf04 Equ 3Ch ;|
Bf05 Equ 3Dh ;/\
Bf06 Equ 3Eh ;/
Bf10 Equ 3Fh ;\
Bf11 Equ 40h ;| buffer
Bf12 Equ 41h ;/\
Bf20 Equ 42h ;\
Bf21 Equ 43h ;|
Bf22 Equ 44h ;| buffer
Bf23 Equ 45h ;|
Bf24 Equ 46h ;|
Bf25 Equ 47h ;/\
Bf30 Equ 48h ;\
Bf31 Equ 49h ;|
Bf32 Equ 4Ah ;| buffer
Bf33 Equ 4Bh ;|
Bf34 Equ 4Ch ;|
Bf35 Equ 4Dh ;/\
Bf40 Equ 4Eh ;\
Bf41 Equ 4Fh ;|
Bf42 Equ 50h ;| buffer
Bf43 Equ 51h ;|
Bf44 Equ 52h ;/\
;-----
Tmo0 Equ 60h ;
Tmo1 Equ 61h ;
Dtkp Equ 62h ; data keypad
Juml Equ 63h ;
Colm Equ 64h ;
Dadc Equ 65h ;
Dbrt Equ 66h ;
Zero Equ 67h ;
Cntr Equ 68h ; counter
Char Equ 69h ; character LCD
;
Urut Equ 6Ah ; urut address
Adbm Equ 6Bh ; address base
Admm Equ 6Ch ; address memory
Dtmn Equ 6Dh ; data memory
;
Ctkp Equ 6Eh ; character printer
;
Dly0 Equ 70h ; delay 0
Dly1 Equ 71h ; delay 1
Dly2 Equ 72h ; delay 2
Dly3 Equ 73h ; delay 3
;-----
init: lcall lcd_in ; inialisasi LCD
;
mulai: mov DPTR,#tpnama ; [Tulis nama
;
;-----
lcall line1 ;|
mov Char,#16 ;|
lcall tulis ;|
mov DPTR,#tpnims ;| Tulis NIM
lcall line2 ;|
mov Char,#16 ;|
lcall tulis ;|
lcall delay2 ;|
mov DPTR,#tpjurs ;| Tulis jurusan
lcall line1 ;|
mov Char,#16 ;|
lcall tulis ;|
mov DPTR,#tpuniv ;| Tulis universitas
lcall line2 ;|
mov Char,#16 ;|
lcall tulis ;|
lcall delay2 ;|
lcall lcdclr ;|
mov Cntr,#50 ;|
;-----
loop: lcall bc_jam ;
mov DPTR,#angka ;
mov P0,#084h ;
lcall w_ins ;
mov A,Djam ;
lcall nil ;
mov P0,#' ' ;
lcall w_chr ;
mov A,Dmnt ;
lcall nil ;
mov P0,#' ' ;
lcall w_chr ;
mov A,Ddtk ;
lcall nil ;
mov P0,#0C3h ;
lcall w_ins ;
mov A,Dtgl ;
lcall nil ;
mov P0,#' / ' ;
lcall w_chr ;
mov A,Dbln ;
lcall nil ;
mov P0,#' / ' ;
lcall w_chr ;
mov A,Dcen ;
lcall nil ;
mov A,Dthn ;
lcall nil ;
mov P0,#0D0h ;
lcall w_ins ;
lcall delay1 ;
djnz Cntr,loop ;
ljmp mulai ;
;-----
tlnama: lcall lcdclr ;
mov DPTR,#tptlpm ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
lcall tg_lps ;
mov Dly3,#5 ;
lcall delay3 ;
;
lcall lcdclr ;
lcall rstchr ;
mov Jum1,#16 ;
mov Colm,#080h ;
tlnm00: mov Dtkp,#10 ;
lcall tg_tkn ;
cjne R0,#11,tlnm01 ;
mov SP,#07h ;
ljmp mulai ;
tlnm01: cjne R0,#12,tlnm02 ;

```

```

    lcall wrmnma ;
    ljmp menu ;
tlmm02: lcall senchr ;
tlmm03: mov P0,Colm ;
    lcall w_ins ;
    lcall wr_chr ;
    lcall bfchr0 ;
    mov P0,Colm ;
    lcall w_ins ;
    lcall tg_lps ;
    mov Tmo0,#10 ;
    mov Tmo1,#0 ;
tlmm04: lcall scnkpd ;
    cjne R0,#10,tlmm05 ;
    djnz Tmo1,tlmm04 ;
    djnz Tmo0,tlmm04 ;
    ljmp tlmm12 ;
tlmm05: cjne R0,#11,tlmm06 ;
    mov SP,#07h ;
    ljmp mulai ;
tlmm06: cjne R0,#12,tlmm07 ;
    lcall wrmnma ;
    ljmp menu ;
tlmm07: cjne R0,#13,tlmm08 ;
    ljmp tlmm04 ;
tlmm08: cjne R0,#14,tlmm09 ;
    ljmp tlmm04 ;
tlmm09: cjne R0,#15,tlmm10 ;
    ljmp tlmm04 ;
tlmm10: cjne R0,#16,tlmm11 ;
    ljmp tlmm04 ;
tlmm11: lcall senchr ;
    jnb Stch,tlmm03 ;
    inc Colm ;
    djnz Jum1,tlmm03 ;
    lcall wrmnma ;
    ljmp menu ;
tlmm12: inc Colm ;
    mov P0,Colm ;
    lcall w_ins ;
    djnz Jum1,tlmm13 ;
    lcall wrmnma ;
    ljmp menu ;
tlmm13: ljmp tlmm00 ;
-----
bfchr0: mov R4,Colm ;
    cjne R4,#080h,bfch00 ;
    mov Bf00,A ;
bfch00: cjne R4,#081h,bfch01 ;
    mov Bf01,A ;
bfch01: cjne R4,#082h,bfch02 ;
    mov Bf02,A ;
bfch02: cjne R4,#083h,bfch03 ;
    mov Bf03,A ;
bfch03: cjne R4,#084h,bfch04 ;
    mov Bf04,A ;
bfch04: cjne R4,#085h,bfch05 ;
    mov Bf05,A ;
bfch05: cjne R4,#086h,bfch06 ;
    mov Bf10,A ;
bfch06: cjne R4,#087h,bfch07 ;
    mov Bf11,A ;
bfch07: cjne R4,#088h,bfch08 ;
    mov Bf12,A ;
bfch08: cjne R4,#089h,bfch09 ;
    mov Bf20,A ;
bfch09: cjne R4,#08Ah,bfch0A ;
    mov Bf21,A ;
bfch0A: cjne R4,#08Bh,bfch0B ;
    mov Bf22,A ;
bfch0B: cjne R4,#08Ch,bfch0C ;
    mov Bf23,A ;
bfch0C: cjne R4,#08Dh,bfch0D ;
    mov Bf24,A ;
bfch0D: cjne R4,#08Eh,bfch0E ;
    mov Bf30,A ;
bfch0E: cjne R4,#08Fh,bfch0F ;
    mov Bf31,A ;
bfch0F: ret ;
-----
scnchr: clr Stch ;
    mov A,R0 ;
    mov B,Dtkp ;
    clr C ;
    subb A,B ;
    jnz scchr0 ;
    ljmp scchr1 ;
scchr0: setb Stch ;
    ljmp scchr3 ;
scchr1: djnz R1,scchr2 ;
    mov R1,Cntr ;
    ljmp scchr3 ;
scchr2: inc DPTR ;
    ljmp scchr4 ;
scchr3: mov Dtkp,R0 ;
    mov A,Dtkp ;
    lcall sklmpk ;
scchr4: clr A ;
    ret ;
-----
sklmpk: cjne A,#1,sklmp0 ;
    mov DPTR,#klmpk0 ;
    mov Cntr,#2 ;
    mov R1,Cntr ;
sklmp0: cjne A,#2,sklmp1 ;
    mov DPTR,#klmpk1 ;
    mov Cntr,#4 ;
    mov R1,Cntr ;
sklmp1: cjne A,#3,sklmp2 ;
    mov DPTR,#klmpk2 ;
    mov Cntr,#4 ;
    mov R1,Cntr ;
sklmp2: cjne A,#4,sklmp3 ;
    mov DPTR,#klmpk3 ;
    mov Cntr,#4 ;
    mov R1,Cntr ;
sklmp3: cjne A,#5,sklmp4 ;
    mov DPTR,#klmpk4 ;
    mov Cntr,#4 ;
    mov R1,Cntr ;
sklmp4: cjne A,#6,sklmp5 ;
    mov DPTR,#klmpk5 ;
    mov Cntr,#4 ;
    mov R1,Cntr ;
sklmp5: cjne A,#7,sklmp6 ;
    mov DPTR,#klmpk6 ;
    mov Cntr,#5 ;
    mov R1,Cntr ;
sklmp6: cjne A,#8,sklmp7 ;
    mov DPTR,#klmpk7 ;
    mov Cntr,#4 ;
    mov R1,Cntr ;
sklmp7: cjne A,#9,sklmp8 ;
    mov DPTR,#klmpk8 ;
    mov Cntr,#5 ;
    mov R1,Cntr ;
sklmp8: cjne A,#0,sklmp9 ;
    mov DPTR,#klmpk9 ;
    mov Cntr,#2 ;
    mov R1,Cntr ;
sklmp9: ret ;
-----
rstchr: mov Bf00,#' ;
    mov Bf01,#' ;
    mov Bf02,#' ;
    mov Bf03,#' ;
    mov Bf04,#' ;
    mov Bf05,#' ;
    mov Bf10,#' ;
    mov Bf11,#' ;
    mov Bf12,#' ;
    mov Bf20,#' ;

```

```

mov Bf21,#' ;
mov Bf22,#' ;
mov Bf23,#' ;
mov Bf24,#' ;
mov Bf30,#' ;
mov Bf31,#' ;
ret ;
;-----Pemilihan Jenis Emas-----
menu: mov DPTR,#tpjnem ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
mov R1,#1 ;
menu00: cjne R1,#1,menu01 ;
mov DPTR,#tpjns0 ; kalung
menu01: cjne R1,#2,menu02 ;
mov DPTR,#tpjns1 ; liontin
menu02: cjne R1,#3,menu03 ;
mov DPTR,#tpjns2 ; gondel
menu03: cjne R1,#4,menu04 ;
mov DPTR,#tpjns3 ; gelang
menu04: cjne R1,#5,menu05 ;
mov DPTR,#tpjns4 ; cincin
menu05: cjne R1,#6,menu06 ;
mov DPTR,#tpjns5 ; giwang
menu06: cjne R1,#7,menu07 ;
mov DPTR,#tpjns6 ; peniti
menu07: lcall line2 ;
mov Char,#16 ;
lcall tulis ;
lcall tg_lps ;
menu08: lcall senkpd ;
cjne R0,#15,menu10 ;
dec R1 ;
cjne R1,#0,menu09 ;
mov R1,#1 ;
menu09: ljmp menu00 ;
menu10: cjne R0,#16,menu12 ;
inc R1 ;
cjne R1,#8,menu11 ;
mov R1,#7 ;
menu11: ljmp menu00 ;
menu12: cjne R0,#11,menu13 ;
mov SP,#07h ;
ljmp mulai ;
menu13: cjne R0,#12,menu08 ;
mov Bf00,R1 ;
lcall wrmjns ;
lcall tg_lps ;
;-----Entry Nilai Tukar Emas-----
menu14: mov DPTR,#niltkr ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
lcall line2 ;
mov Char,#16 ;
lcall tulis ;
lcall rdmtkr ;
mov DPTR,#angka ;
mov P0,#0C6h ;
lcall w_ins ;
mov A,Bf04 ;
lcall wr_chr ;
mov A,Bf03 ;
lcall wr_chr ;
mov A,Bf02 ;
lcall wr_chr ;
mov P0,#' ;
lcall w_chr ;
mov A,Bf01 ;
lcall wr_chr ;
mov A,Bf00 ;
lcall wr_chr ;
mov P0,#0' ;
lcall w_chr ;
mov P0,#0D0h ;
lcall w_ins ;
menu15: lcall senkpd ;
cjne R0,#10,menu16 ;
ljmp menu15 ;
menu16: cjne R0,#11,menu17 ;
mov SP,#07h ;
ljmp mulai ;
menu17: cjne R0,#12,menu18 ;
ljmp menu25 ;
menu18: cjne R0,#13,menu19 ;
ljmp menu15 ;
menu19: cjne R0,#14,menu20 ;
ljmp menu15 ;
menu20: cjne R0,#15,menu21 ;
ljmp menu15 ;
menu21: cjne R0,#16,menu22 ;
ljmp menu15 ;
menu22: mov DPTR,#angka ;
mov P0,#0C6h ;
lcall w_ins ;
mov A,R0 ;
mov Bf04,A ;
lcall wr_chr ;
lcall tg_lps ;
lcall tg_tkn ;
mov A,R0 ;
mov Bf03,A ;
lcall wr_chr ;
lcall tg_lps ;
lcall tg_tkn ;
mov A,R0 ;
mov Bf02,A ;
lcall wr_chr ;
lcall tg_lps ;
mov P0,#' ;
lcall w_chr ;
lcall tg_tkn ;
mov A,R0 ;
mov Bf01,A ;
lcall wr_chr ;
lcall tg_lps ;
lcall tg_tkn ;
mov A,R0 ;
mov Bf00,A ;
lcall wr_chr ;
lcall tg_lps ;
mov P0,#0D0h ;
lcall w_ins ;
menu23: lcall senkpd ;
cjne R0,#11,menu24 ;
mov P0,#0C6h ;
lcall w_ins ;
lcall tg_tkn ;
ljmp menu22 ;
menu24: cjne R0,#12,menu23 ;
lcall tg_lps ;
lcall wrmtkr ;
;-----
menu25: mov DPTR,#brtems ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
lcall line2 ;
mov Char,#16 ;
lcall tulis ;
mov Dly3,#1 ;
lcall delay3 ;
lcall bc_brt ;
mov Zero,Dbtr ;
menu26: lcall bc_brt ;
mov DPTR,#angka ;
mov P0,#0C3h ;
lcall w_ins ;
mov A,Dbtr ;
mov B,#100 ;
div AB ;

```

```

mov Bf02,A ;
lcall wr_chr ;
mov A,B ;
mov B,#10 ;
div AB ;
mov Bf01,A ;
lcall wr_chr ;
mov PO,#' ;
lcall w_chr ;
mov A,B ;
mov Bf00,A ;
lcall wr_chr ;
mov PO,#'g' ;
lcall w_chr ;
mov PO,#0D0h ;
lcall w_ins ;
mov Dly3,#1 ;
lcall delay3 ;
menu27: lcall scnkpd ;
cjne R0,#11,menu28 ;
mov SP,#07h ;
ljmp mulai ;
menu28: cjne R0,#12,menu26 ;
lcall wrnbrt ;
lcall tg_lps ;
;-----
menu29: mov DPTR,#harga ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
lcall line2 ;
mov Char,#16 ;
lcall tulis ;
mov Bf10,Bf00 ;
mov Bf11,Bf01 ;
mov Bf12,Bf02 ;
lcall rdmtkr ;
lcall ntklbr ; nilai tukar kali berat
mov DPTR,#angka ;
mov PO,#0C5h ;
lcall w_ins ;
mov A,Bf06 ;
lcall wr_chr ;
mov PO,#' ;
lcall w_chr ;
mov A,Bf05 ;
lcall wr_chr ;
mov A,Bf04 ;
lcall wr_chr ;
mov A,Bf03 ;
lcall wr_chr ;
mov PO,#' ;
lcall w_chr ;
mov A,Bf02 ;
lcall wr_chr ;
mov A,Bf01 ;
lcall wr_chr ;
mov A,Bf00 ;
lcall wr_chr ;
mov PO,#0D0h ;
lcall w_ins ;
menu30: lcall scnkpd ;
cjne R0,#11,menu31 ;
mov SP,#07h ;
ljmp mulai ;
menu31: cjne R0,#12,menu30 ;
lcall wrmhrgr ;
lcall tg_lps ;
;-----Entry Ongkos-----
menu32: mov DPTR,#ongkos ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
lcall line2 ;
mov Char,#16 ;
lcall tulis ;
mov DPTR,#angka ;
lcall w_ins ;
mov A,Bf06 ;
lcall wr_chr ;
mov PO,#' ;
lcall w_chr ;
mov A,Bf05 ;
lcall wr_chr ;
mov A,Bf04 ;
lcall wr_chr ;
mov A,Bf03 ;
lcall wr_chr ;
mov PO,#' ;
lcall w_chr ;
mov A,Bf02 ;
lcall wr_chr ;
mov A,Bf01 ;
lcall wr_chr ;
mov A,Bf00 ;
lcall wr_chr ;
mov DPTR,#angka ;
lcall w_ins ;
mov PO,#0C6h ;
lcall tg_tkn ;
mov A,R0 ;
mov Bf04,A ;
lcall wr_chr ;
lcall tg_lps ;
lcall tg_tkn ;
mov A,R0 ;
mov Bf03,A ;
lcall wr_chr ;
lcall tg_lps ;
lcall tg_tkn ;
mov A,R0 ;
mov Bf02,A ;
lcall wr_chr ;
lcall tg_lps ;
lcall tg_tkn ;
mov PO,#' ;
lcall w_chr ;
lcall tg_tkn ;
mov A,R0 ;
mov Bf01,A ;
lcall wr_chr ;
lcall tg_lps ;
lcall tg_tkn ;
mov A,R0 ;
mov Bf00,A ;
lcall wr_chr ;
lcall tg_lps ;
mov PO,#'0' ;
lcall w_chr ;
mov PO,#0D0h ;
lcall w_ins ;
menu33: lcall scnkpd ;
cjne R0,#11,menu34 ;
ljmp menu32 ;
menu34: cjne R0,#12,menu33 ;
lcall wrmong ;
lcall tg_lps ;
;-----
menu35: mov DPTR,#total ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
lcall line2 ;
mov Char,#16 ;
lcall tulis ;
mov Bf20,Bf00 ;
mov Bf21,Bf01 ;
mov Bf22,Bf02 ;
mov Bf23,Bf03 ;
mov Bf24,Bf04 ;
lcall rdmhrgr ;
lcall tothrg ;
mov DPTR,#angka ;
mov PO,#0C5h ;
lcall w_ins ;
mov A,Bf06 ;
lcall wr_chr ;
mov PO,#' ;
lcall w_chr ;
mov A,Bf05 ;
lcall wr_chr ;
mov A,Bf04 ;
lcall wr_chr ;
mov A,Bf03 ;
lcall wr_chr ;
mov PO,#' ;
lcall w_chr ;
mov A,Bf02 ;
lcall wr_chr ;
mov A,Bf01 ;
lcall wr_chr ;
mov A,Bf00 ;
lcall wr_chr ;

```

```

lcall wr_chr ;
mov PO,#0D0h ;
lcall w_ins ;
menu36: lcall scnkpdl ;
cjne R0,#11,menu37 ;
mov SP,#07h ;
ljmp mulai ;
menu37: cjne R0,#12,menu36 ;
lcall wrmtot ;
lcall tg_lps ;
;
menu38: lcall ledclr ;
lcall rdmurt ;
mov A,Urut ;
mov B,#40 ;
div AB ;
jz menu39 ;
mov DPTR,#membhs ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
mov Dly3,#10 ;
lcall delay3 ;
ljmp menu40 ;
;
menu39: mov DPTR,#savmem ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
mov DPTR,#angka ;
mov PO,#08Dh ;
lcall w_ins ;
mov A,Urut ;
lcall nilai ;
;
mov DPTR,#petamm ;
mov A,Urut ;
mov B,#2 ;
mul AB ;
mov Bufr,A ;
movc A,@A+DPTR ;
mov Adbm,A ;
inc Bufr ;
mov A,Bufr ;
movc A,@A+DPTR ;
mov Admm,A ;
;
lcall rdmmma ;
lcall bc_jam ;
lcall wrdnmj ;
;
lcall rdmjns ;
lcall wrdjns ;
;
lcall rdmtkr ;
lcall wrdtkr ;
;
lcall rdmbrt ;
lcall wrdbrt ;
;
lcall rdmhrg ;
lcall wrdhrgr ;
;
lcall rdmong ;
lcall wrdong ;
;
lcall rdmtot ;
lcall wrdtot ;
;
mov Dly3,#5 ;
lcall delay3 ;
lcall rdmurt ;
inc Urut ;
lcall wrmurt ;
;
menu40: mov DPTR,#prnspl ;
;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
menu41: lcall scnkpdl ;
cjne R0,#11,menu42 ;
mov SP,#07h ;
ljmp mulai ;
menu42: cjne R0,#12,menu41 ;
lcall tg_lps ;
mov DPTR,#tpplwt ;
lcall line2 ;
mov Char,#16 ;
lcall tulis ;
;
mov DPTR,#prtKem ;
mov Ctkp,#23 ;
lcall tpsprn ;
mov Ctkp,#23 ;
lcall tpsprn ;
lcall line ;
;
mov DPTR,#tnggal ;
mov Ctkp,#12 ;
lcall cetak ;
mov DPTR,#angka ;
mov A,Djam ;
mov B,#10 ;
div AB ;
lcall ct_chr ;
mov A,B ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ' ;
lcall strobe ;
mov A,Dmnt ;
mov B,#10 ;
div AB ;
lcall ct_chr ;
mov A,B ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ' ;
lcall strobe ;
jb Busi,$ ;
mov P3,#' ' ;
lcall strobe ;
mov A,Dtgl ;
mov B,#10 ;
div AB ;
lcall ct_chr ;
mov A,B ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ' ;
lcall strobe ;
mov A,Dbln ;
mov B,#10 ;
div AB ;
lcall ct_chr ;
mov A,B ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ' ;
lcall strobe ;
mov A,Dthn ;
mov B,#10 ;
div AB ;
lcall ct_chr ;
mov A,B ;
lcall ct_chr ;
lcall line ;
;
mov DPTR,#namapb ;
mov Ctkp,#12 ;
lcall cetak ;
lcall rdmmma ;

```




```

jb Busi,$ ;
mov P3,Bf00 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf01 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf02 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf03 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf04 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf05 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf10 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf11 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf12 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf20 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf21 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf22 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf23 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf24 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf30 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf31 ;
lcall strobe ;
lcall line ;
;
mov DPTR,#prjns ;
mov Ctkp,#12 ;
lcall cetak ;
lcall rdmjns ;
mov A,Bf00 ;
ctjn00: cjne A,#1,ctjn01 ;
mov DPTR,#prjns0 ;
ctjn01: cjne A,#2,ctjn02 ;
mov DPTR,#prjns1 ;
ctjn02: cjne A,#3,ctjn03 ;
mov DPTR,#prjns2 ;
ctjn03: cjne A,#4,ctjn04 ;
mov DPTR,#prjns3 ;
ctjn04: cjne A,#5,ctjn05 ;
mov DPTR,#prjns4 ;
ctjn05: cjne A,#6,ctjn06 ;
mov DPTR,#prjns5 ;
ctjn06: cjne A,#7,ctjn07 ;
mov DPTR,#prjns6 ;
ctjn07: mov Ctkp,#8 ;
lcall cetak ;
lcall line ;
;
mov DPTR,#prbrt ;
mov Ctkp,#12 ;
lcall cetak ;
lcall rdmbtr ;
mov DPTR,#angka ;
mov A,Bf02 ;
lcall ct_chr ;
mov A,Bf01 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ' ;
lcall strobe ;
mov A,Bf00 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ' ;
lcall strobe ;
jb Busi,$ ;
mov P3,#'g' ;
lcall strobe ;
jb Busi,$ ;
mov P3,#'r' ;
lcall strobe ;
jb Busi,$ ;
mov P3,#'a' ;
lcall strobe ;
jb Busi,$ ;
mov P3,#'m' ;
lcall strobe ;
lcall line ;
;
mov DPTR,#prnltk ;
mov Ctkp,#15 ;
lcall cetak ;
lcall rdmtkr ;
mov DPTR,#angka ;
mov A,Bf04 ;
lcall ct_chr ;
mov A,Bf03 ;
lcall ct_chr ;
mov A,Bf02 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ' ;
lcall strobe ;
mov A,Bf01 ;
lcall ct_chr ;
mov A,Bf00 ;
lcall ct_chr ;
mov A,#0 ;
lcall ct_chr ;
lcall line ;
;
mov DPTR,#prhrng ;
mov Ctkp,#15 ;
lcall cetak ;
lcall rdmhrng ;
mov DPTR,#angka ;
mov A,Bf06 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ' ;
lcall strobe ;
mov A,Bf05 ;
lcall ct_chr ;
mov A,Bf04 ;
lcall ct_chr ;
mov A,Bf03 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ' ;
lcall strobe ;
mov A,Bf02 ;
lcall ct_chr ;
mov A,Bf01 ;
lcall ct_chr ;
mov A,Bf00 ;
lcall ct_chr ;
lcall line ;

```

```

;
mov DPTR,#prong ;
mov Ctkp,#15 ;
lcall cetak ;
lcall rdmgong ;
mov DPTR,#angka ;
mov A,Bf04 ;
lcall ct_chr ;
mov A,Bf03 ;
lcall ct_chr ;
mov A,Bf02 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
mov A,Bf01 ;
lcall ct_chr ;
mov A,Bf00 ;
lcall ct_chr ;
mov A,#0 ;
lcall ct_chr ;
lcall line ;

;
mov DPTR,#prttl ;
mov Ctkp,#15 ;
lcall cetak ;
lcall rdmtot ;
mov DPTR,#angka ;
mov A,Bf06 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
mov A,Bf05 ;
lcall ct_chr ;
mov A,Bf04 ;
lcall ct_chr ;
mov A,Bf03 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
mov A,Bf02 ;
lcall ct_chr ;
mov A,Bf01 ;
lcall ct_chr ;
mov A,Bf00 ;
lcall ct_chr ;
lcall line ;

;
lcall line ;
mov DPTR,#pesan ;
mov Ctkp,#23 ;
lcall tlsprn ;
mov Ctkp,#23 ;
lcall tlsprn ;
lcall line ;
lcall line ;
lcall line ;
lcall line ;
lcall line ;
lcall line ;
lcall line ;
lcall line ;
lcall line ;
lcall line ;
mov SP,#07h ;
ljmp mulai ;

;
tlsprn: lcall cetak ;
lcall line ;
ret ;

;
cetak: clr A ;
lcall ct_chr ;
inc DPTR ;
djnz Ctkp,cetak ;

;
ret ;

;
ct_chr: movc A,@A+DPTR ;
jb Busi,$ ;
mov P3,A ;
lcall strobe ;
ret ;

;
line: jb Busi,$ ;
mov P3,#0Ah ;
lcall strobe ;
ret ;

;
enter: jb Busi,$ ;
mov P3,#0Dh ;
lcall strobe ;
ret ;

;
strobe: setb Strb ;
clr Strb ;
lcall delay0 ;
ret ;

;
setime: lcall lcdclr ;
mov DPTR,#tpsttm ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
lcall tg_lps ;

;
mov Dly3,#5 ;
lcall delay3 ;

;
hplprn: lcall lcdclr ;
mov DPTR,#tphplp ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
lcall tg_lps ;

;
hplpr0: lcall scnkpdp ;
cjne R0,#11,hplpr1 ;
mov SP,#07h ;
ljmp mulai ;

;
hplpr1: cjne R0,#13,hplpr2 ;
ljmp setime ;

;
hplpr2: cjne R0,#12,hplpr0 ;
mov DPTR,#tpplwt ;
lcall line2 ;
mov Char,#16 ;
lcall tulis ;
mov Bf00,#'-' ;
mov Bf01,#'-' ;
mov Bf02,#'-' ;
mov Bf03,#'-' ;
mov Bf04,#'-' ;
mov Bf05,#'-' ;
mov Bf10,#'-' ;
mov Bf11,#'-' ;
mov Bf12,#'-' ;
mov Bf20,#'-' ;
mov Bf21,#'-' ;
mov Bf22,#'-' ;
mov Bf23,#'-' ;
mov Bf24,#'-' ;
mov Bf30,#'-' ;
mov Bf31,#'-' ;
mov Dtgl,#0 ;
mov Dbln,#0 ;
mov Dthn,#0 ;
mov Djam,#0 ;
mov Dmnt,#0 ;
mov Urut,#0 ;
mov DPTR,#petamm ;

;
hplpr3: mov A,Urut ;
mov B,#2 ;
mul AB ;

```

```

mov Bufr,A ;
movc A,@A+DPTR ;
mov Adbm,A ;
inc Bufr ;
mov A,Bufr ;
movc A,@A+DPTR ;
mov Admm,A ;
lcall wrdnmj ;
inc Urut ;
mov A,Urut ;
cjne A,#40,hplpr3 ;
mov Urut,#0 ;
lcall wrmurt ;
mov SP,#07h ;
ljmp mulai ;
;
laporn: lcall lcdclr ;
mov DPTR,#tplprn ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
lcall tg_lps ;
laprn0: lcall scnkpd ;
cjne R0,#11,laprn1 ;
mov SP,#07h ;
ljmp mulai ;
laprn1: cjne R0,#14,laprn2 ;
ljmp kalbrs ;
laprn2: cjne R0,#12,laprn0 ;
mov Urut,#0 ;
laprn3: mov DPTR,#petamm ;
mov A,Urut ;
mov B,#2 ;
mul AB ;
mov Bufr,A ;
movc A,@A+DPTR ;
mov Adbm,A ;
inc Bufr ;
mov A,Bufr ;
movc A,@A+DPTR ;
mov Admm,A ;
lcall rddnmj ;
lcall lcdclr ;
;
lcall line1 ;
mov P0,Bf00 ;
lcall w_chr ;
mov P0,Bf01 ;
lcall w_chr ;
mov P0,Bf02 ;
lcall w_chr ;
mov P0,Bf03 ;
lcall w_chr ;
mov P0,Bf04 ;
lcall w_chr ;
mov P0,Bf05 ;
lcall w_chr ;
mov P0,Bf10 ;
lcall w_chr ;
mov P0,Bf11 ;
lcall w_chr ;
mov P0,Bf12 ;
lcall w_chr ;
mov P0,Bf20 ;
lcall w_chr ;
mov P0,Bf21 ;
lcall w_chr ;
mov P0,Bf22 ;
lcall w_chr ;
mov P0,Bf23 ;
lcall w_chr ;
mov P0,Bf24 ;
lcall w_chr ;
mov P0,Bf30 ;
lcall w_chr ;
mov P0,Bf31 ;
;
lcall w_chr ;
;
mov DPTR,#angka ;
mov P0,#0C1h ;
lcall w_ins ;
mov A,Djam ;
lcall nil ;
mov P0,#' ;
lcall w_chr ;
mov A,Dmnt ;
lcall nil ;
mov P0,#0C7h ;
lcall w_ins ;
mov A,Dtgl ;
lcall nil ;
mov P0,#' ;
lcall w_chr ;
mov A,Dbln ;
lcall nil ;
mov P0,#' ;
lcall w_chr ;
mov A,Dthn ;
lcall nil ;
;
mov P0,#0D0h ;
lcall w_ins ;
lcall tg_lps ;
laprn4: lcall scnkpd ;
cjne R0,#11,laprn5 ;
mov SP,#07h ;
ljmp mulai ;
laprn5: cjne R0,#15,laprn7 ;
dec Urut ;
mov A,Urut ;
cjne A,#255,laprn6 ;
mov Urut,#0 ;
laprn6: ljmp laprn3 ;
laprn7: cjne R0,#16,laprn9 ;
inc Urut ;
mov A,Urut ;
cjne A,#40,laprn8 ;
mov Urut,#39 ;
laprn8: ljmp laprn3 ;
;
laprn9: cjne R0,#12,laprn4 ;
mov DPTR,#prnslp ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
lcall tg_lps ;
laprnA: lcall scnkpd ;
cjne R0,#11,laprnB ;
mov SP,#07h ;
ljmp mulai ;
laprnB: cjne R0,#12,laprnA ;
lcall tg_lps ;
mov DPTR,#tpplwt ;
lcall line2 ;
mov Char,#16 ;
lcall tulis ;
;
mov DPTR,#prtKem ;
mov Ctkp,#23 ;
lcall tlsprn ;
mov Ctkp,#23 ;
lcall tlsprn ;
lcall line ;
;
mov DPTR,#petamm ;
mov A,Urut ;
mov B,#2 ;
mul AB ;
mov Bufr,A ;
movc A,@A+DPTR ;
mov Adbm,A ;
inc Bufr ;

```

```

mov A,Bufr ;
movc A,@A+DPTR ;
mov Admm,A ;
lcall rddnmj ;
mov DPTR,#tnggal ;
mov Ctkp,#12 ;
lcall cetak ;
mov DPTR,#angka ;
mov A,Djam ;
mov B,#10 ;
div AB ;
lcall ct_chr ;
mov A,B ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
mov A,Dmnt ;
mov B,#10 ;
div AB ;
lcall ct_chr ;
mov A,B ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
mov A,Dtgl ;
mov B,#10 ;
div AB ;
lcall ct_chr ;
mov A,B ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
mov A,Dbln ;
mov B,#10 ;
div AB ;
lcall ct_chr ;
mov A,B ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
mov A,Dthn ;
mov B,#10 ;
div AB ;
lcall ct_chr ;
mov A,B ;
lcall ct_chr ;
lcall line ;
;
mov DPTR,#namapb ;
mov Ctkp,#12 ;
lcall cetak ;
jb Busi,$ ;
mov P3,Bf00 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf01 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf02 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf03 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf04 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf05 ;
lcall strobe ;
;
jb Busi,$ ;
mov P3,Bf10 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf11 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf12 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf20 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf21 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf22 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf23 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf24 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf30 ;
lcall strobe ;
jb Busi,$ ;
mov P3,Bf31 ;
lcall strobe ;
lcall line ;
;
mov DPTR,#prjns ;
mov Ctkp,#12 ;
lcall cetak ;
mov DPTR,#petamm ;
mov A,Urut ;
mov B,#2 ;
mul AB ;
mov Bufr,A ;
movc A,@A+DPTR ;
mov Adbm,A ;
inc Bufr ;
mov A,Bufr ;
movc A,@A+DPTR ;
mov Admm,A ;
mov Cntr,#21 ;
ctdtm0: inc Admm ;
djnz Cntr,ctdtm0 ;
lcall rddjns ;
mov A,Bf00 ;
ctjn10: cjne A,#1,ctjn11 ;
mov DPTR,#prjns0 ;
ctjn11: cjne A,#2,ctjn12 ;
mov DPTR,#prjns1 ;
ctjn12: cjne A,#3,ctjn13 ;
mov DPTR,#prjns2 ;
ctjn13: cjne A,#4,ctjn14 ;
mov DPTR,#prjns3 ;
ctjn14: cjne A,#5,ctjn15 ;
mov DPTR,#prjns4 ;
ctjn15: cjne A,#6,ctjn16 ;
mov DPTR,#prjns5 ;
ctjn16: cjne A,#7,ctjn17 ;
mov DPTR,#prjns6 ;
ctjn17: mov Ctkp,#8 ;
lcall cetak ;
lcall line ;
;
mov DPTR,#prbrt ;
mov Ctkp,#12 ;
lcall cetak ;
mov DPTR,#petamm ;
mov A,Urut ;
mov B,#2 ;
mul AB ;

```

```

mov Bufr,A ;
movc A,@A+DPTR ;
mov Adbm,A ;
inc Bufr ;
mov A,Bufr ;
movc A,@A+DPTR ;
mov Admm,A ;
mov Cntr,#27 ;
ctdtm1: inc Admm ;
djnz Cntr,ctdtm1 ;
lcall rddbrt ;
mov DPTR,#angka ;
mov A,Bf02 ;
lcall ct_chr ;
mov A,Bf01 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
mov A,Bf00 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
jb Busi,$ ;
mov P3,#'g' ;
lcall strobe ;
jb Busi,$ ;
mov P3,#'r' ;
lcall strobe ;
jb Busi,$ ;
mov P3,#'a' ;
lcall strobe ;
jb Busi,$ ;
mov P3,#'m' ;
lcall strobe ;
lcall line ;
;
mov DPTR,#prnltk ;
mov Ctkp,#15 ;
lcall cetak ;
mov DPTR,#petamm ;
mov A,Urut ;
mov B,#2 ;
mul AB ;
mov Bufr,A ;
movc A,@A+DPTR ;
mov Adbm,A ;
inc Bufr ;
mov A,Bufr ;
movc A,@A+DPTR ;
mov Admm,A ;
mov Cntr,#22 ;
ctdtm2: inc Admm ;
djnz Cntr,ctdtm2 ;
lcall rddtkr ;
mov DPTR,#angka ;
mov A,Bf04 ;
lcall ct_chr ;
mov A,Bf03 ;
lcall ct_chr ;
mov A,Bf02 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
mov A,Bf01 ;
lcall ct_chr ;
mov A,Bf00 ;
lcall ct_chr ;
mov A,#0 ;
lcall ct_chr ;
lcall line ;
;
mov DPTR,#prhrg ;
mov Ctkp,#15 ;
;
lcall cetak ;
mov DPTR,#petamm ;
mov A,Urut ;
mov B,#2 ;
mul AB ;
mov Bufr,A ;
movc A,@A+DPTR ;
mov Adbm,A ;
inc Bufr ;
mov A,Bufr ;
movc A,@A+DPTR ;
mov Admm,A ;
mov Cntr,#37 ;
ctdtm3: inc Admm ;
djnz Cntr,ctdtm3 ;
lcall rddhrg ;
mov DPTR,#angka ;
mov A,Bf06 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
mov A,Bf05 ;
lcall ct_chr ;
mov A,Bf04 ;
lcall ct_chr ;
mov A,Bf03 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
mov A,Bf02 ;
lcall ct_chr ;
mov A,Bf01 ;
lcall ct_chr ;
mov A,Bf00 ;
lcall ct_chr ;
lcall line ;
;
mov DPTR,#prong ;
mov Ctkp,#15 ;
lcall cetak ;
mov DPTR,#petamm ;
mov A,Urut ;
mov B,#2 ;
mul AB ;
mov Bufr,A ;
movc A,@A+DPTR ;
mov Adbm,A ;
inc Bufr ;
mov A,Bufr ;
movc A,@A+DPTR ;
mov Admm,A ;
mov Cntr,#30 ;
ctdtm4: inc Admm ;
djnz Cntr,ctdtm4 ;
lcall rddong ;
mov DPTR,#angka ;
mov A,Bf04 ;
lcall ct_chr ;
mov A,Bf03 ;
lcall ct_chr ;
mov A,Bf02 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ;
lcall strobe ;
mov A,Bf01 ;
lcall ct_chr ;
mov A,Bf00 ;
lcall ct_chr ;
mov A,#0 ;
lcall ct_chr ;
lcall line ;
;
mov DPTR,#prttl ;

```

```

mov Ctkp,#15 ;
lcall cetak ;
mov DPTR,#petamm ;
mov A,Urut ;
mov B,#2 ;
mul AB ;
mov Bufr,A ;
movc A,@A+DPTR ;
mov Adbm,A ;
inc Bufr ;
mov A,Bufr ;
movc A,@A+DPTR ;
mov Admm,A ;
mov Cntr,#42 ;
ctdtm5: inc Admm ;
djnz Cntr,ctdtm5 ;
lcall rddtot ;
mov DPTR,#angka ;
mov A,Bf06 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ' ;
lcall strobe ;
mov A,Bf05 ;
lcall ct_chr ;
mov A,Bf04 ;
lcall ct_chr ;
mov A,Bf03 ;
lcall ct_chr ;
jb Busi,$ ;
mov P3,#' ' ;
lcall strobe ;
mov A,Bf02 ;
lcall ct_chr ;
mov A,Bf01 ;
lcall ct_chr ;
mov A,Bf00 ;
lcall ct_chr ;
lcall line ;
;
lcall line ;
mov DPTR,#pesan ;
mov Ctkp,#23 ;
lcall tlsprn ;
mov Ctkp,#23 ;
lcall tlsprn ;
lcall line ;
lcall line ;
lcall line ;
lcall line ;
lcall line ;
lcall line ;
lcall line ;
lcall line ;
mov SP,#07h ;
ljmp laprn3 ;
;
kalbrs: lcall lcdclr ;
mov DPTR,#tpklbr ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
lcall tg_lps ;
kalbr0: lcall scnkpd ;
cjne R0,#11,kalbr1 ;
mov SP,#07h ;
ljmp mulai ;
kalbr1: cjne R0,#14,kalbr2 ;
ljmp laporn ;
kalbr2: cjne R0,#12,kalbr0 ;
kalbr3: lcall lcdclr ;
mov DPTR,#tpadbr ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
;
mov DPTR,#angka ;
kalbr4: lcall bc_brt ;
mov P0,#0C1h ;
lcall w_ins ;
mov A,Dadc ;
lcall nilai ;
mov P0,#0CAh ;
lcall w_ins ;
mov A,Dbrt ;
mov B,#100 ;
div AB ;
lcall wr_chr ;
mov A,B ;
mov B,#10 ;
div AB ;
lcall wr_chr ;
mov P0,#' ' ;
lcall w_chr ;
mov A,B ;
lcall wr_chr ;
mov P0,#'g' ;
lcall w_chr ;
mov P0,#0D0h ;
lcall w_ins ;
mov Dly3,#1 ;
lcall delay3 ;
kalbr5: lcall scnkpd ;
cjne R0,#11,kalbr6 ;
mov SP,#07h ;
ljmp mulai ;
kalbr6: cjne R0,#16,kalbr4 ;
;
zerofk: lcall lcdclr ;
mov DPTR,#tpfktr ;
lcall line1 ;
mov Char,#16 ;
lcall tulis ;
mov DPTR,#tpcnen ;
lcall line2 ;
mov Char,#16 ;
lcall tulis ;
zerof0: lcall scnkpd ;
cjne R0,#11,zerof1 ;
mov SP,#07h ;
ljmp mulai ;
zerof1: cjne R0,#12,zerof2 ;
lcall bc_brt ;
mov Zero,Dbrt ;
mov DPTR,#tpplwt ;
lcall line2 ;
mov Char,#16 ;
lcall tulis ;
mov Dly3,#2 ;
lcall delay3 ;
ljmp zerofk ;
zerof2: cjne R0,#15,zerof0 ;
ljmp kalbr3 ;
;
ntklbr: mov A,Bf00 ; nilai tukar kali berat
mov B,Bf10 ;
mul AB ;
lcall hslsmp ;
mov Bf20,A ;
mov A,Bf01 ;
mov B,Bf10 ;
mul AB ;
mov B,Bufr ;
add A,B ;
lcall hslsmp ;
mov Bf21,A ;
mov A,Bf02 ;
mov B,Bf10 ;
mul AB ;
mov B,Bufr ;
add A,B ;
lcall hslsmp ;

```



```

lcall dtatx
mov A,Bf24 ; data memory
lcall dtatx
mov A,Bf30 ; data memory
lcall dtatx
mov A,Bf31 ; data memory
lcall dtatx
lcall i2cstp ; i2c stop
lcall delay0
ret
;
rdmnmma: mov A,#0D0h ; AT24C16 write address
lcall adrtx ;
mov A,#008 ; address memory
lcall dtatx ;
mov A,#0D1h ; AT24C16 read address
lcall adrtx ;
lcall dtarx ; terima data
mov Bf00,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf01,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf02,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf03,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf04,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf05,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf10,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf11,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf12,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf20,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf21,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf22,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf23,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf24,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf30,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf31,A ;
lcall i2cstp ; i2c stop
lcall delay0 ;
ret
;
wrmjns: mov A,#0D0h ; AT24C16 write address
lcall adrtx ;
mov A,#024 ; address memory
lcall dtatx ;
mov A,Bf00 ; data memory
lcall dtatx ;
lcall i2cstp ; i2c stop
lcall delay0 ;
ret
;
rdmjns: mov A,#0D0h ; AT24C16 write address
lcall adrtx ;
mov A,#024 ; address memory
lcall dtatx ;
mov A,#0D1h ; AT24C16 read address
lcall adrtx ;
lcall dtarx ; terima data
mov Bf00,A ;
lcall i2cstp ; i2c stop
lcall delay0 ;
ret
;
wrmtkr: mov A,#0D0h ; AT24C16 write address
lcall adrtx ;
mov A,#025 ; address memory
lcall dtatx ;
mov A,Bf00 ; data memory
lcall dtatx ;
mov A,Bf01 ; data memory
lcall dtatx ;
mov A,Bf02 ; data memory
lcall dtatx ;
mov A,Bf03 ; data memory
lcall dtatx ;
mov A,Bf04 ; data memory
lcall dtatx ;
lcall i2cstp ; i2c stop
lcall delay0 ;
ret
;
rdmtkr: mov A,#0D0h ; AT24C16 write address
lcall adrtx ;
mov A,#025 ; address memory
lcall dtatx ;
mov A,#0D1h ; AT24C16 read address
lcall adrtx ;
lcall dtarx ; terima data
mov Bf00,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf01,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf02,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf03,A ;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf04,A ;
lcall i2cstp ; i2c stop
lcall delay0 ;
ret
;
wrmbrt: mov A,#0D0h ; AT24C16 write address
lcall adrtx ;
mov A,#030 ; address memory
lcall dtatx ;
mov A,Bf00 ; data memory
lcall dtatx ;
mov A,Bf01 ; data memory
lcall dtatx ;
mov A,Bf02 ; data memory
lcall dtatx ;
lcall i2cstp ; i2c stop
lcall delay0 ;
ret
;
rdmbrt: mov A,#0D0h ; AT24C16 write address
lcall adrtx ;
mov A,#030 ; address memory
lcall dtatx ;
mov A,#0D1h ; AT24C16 read address
lcall adrtx ;

```

```

lcall dtarx      ; terima data
mov Bf00,A
lcall givack     ; beri ack
lcall dtarx     ; terima data
mov Bf01,A
lcall givack     ; beri ack
lcall dtarx     ; terima data
mov Bf02,A
lcall i2cstp    ; i2c stop
lcall delay0
ret
;
; wrmhrg: mov A,#0D0h ; AT24C16 write address
lcall adrtx
mov A,#033      ; address memory
lcall dtatx
mov A,Bf00      ; data memory
lcall dtatx
mov A,Bf01      ; data memory
lcall dtatx
mov A,Bf02      ; data memory
lcall dtatx
mov A,Bf03      ; data memory
lcall dtatx
mov A,Bf04      ; data memory
lcall dtatx
mov A,Bf05      ; data memory
lcall dtatx
mov A,Bf06      ; data memory
lcall dtatx
lcall i2cstp    ; i2c stop
lcall delay0
ret
;
; rdmhrg: mov A,#0D0h ; AT24C16 write address
lcall adrtx
mov A,#033      ; address memory
lcall dtatx
mov A,#0D1h    ; AT24C16 read address
lcall adrtx
lcall dtarx    ; terima data
mov Bf00,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf01,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf02,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf03,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf04,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf05,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf06,A
lcall i2cstp   ; i2c stop
lcall delay0
ret
;
; wrmong: mov A,#0D0h ; AT24C16 write address
lcall adrtx
mov A,#040      ; address memory
lcall dtatx
mov A,#0D1h    ; AT24C16 read address
lcall adrtx
lcall dtarx    ; terima data
mov Bf00,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf01,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf02,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf03,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf04,A
lcall i2cstp   ; i2c stop
lcall delay0
ret
;
; wrmtot: mov A,#0D0h ; AT24C16 write address
lcall adrtx
mov A,#045      ; address memory
lcall dtatx
mov A,Bf00      ; data memory
lcall dtatx
mov A,Bf01      ; data memory
lcall dtatx
mov A,Bf02      ; data memory
lcall dtatx
mov A,Bf03      ; data memory
lcall dtatx
mov A,Bf04      ; data memory
lcall dtatx
mov A,Bf05      ; data memory
lcall dtatx
mov A,Bf06      ; data memory
lcall dtatx
lcall i2cstp   ; i2c stop
lcall delay0
ret
;
; rdmtot: mov A,#0D0h ; AT24C16 write address
lcall adrtx
mov A,#045      ; address memory
lcall dtatx
mov A,#0D1h    ; AT24C16 read address
lcall adrtx
lcall dtarx    ; terima data
mov Bf00,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf01,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf02,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf03,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf04,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf05,A
lcall givack   ; beri ack
lcall dtarx   ; terima data
mov Bf06,A
lcall givack   ; beri ack

```

```

;
lcall dtarx ; terima data
mov Bf06,A
lcall i2cstp ; i2c stop
lcall delay0
ret
;
wrmurt: mov A,#0D0h ; AT24C16 write address
lcall adrtx
mov A,#055 ; address memory
lcall dtarx
mov A,Urut ; data memory
lcall dtarx
lcall i2cstp ; i2c stop
lcall delay0
ret
;
rdmurt: mov A,#0D0h ; AT24C16 write address
lcall adrtx
mov A,#055 ; address memory
lcall dtarx
mov A,#0D1h ; AT24C16 read address
lcall adrtx
lcall dtarx ; terima data
mov Urut,A
lcall i2cstp ; i2c stop
lcall delay0
ret
;
wrdnmj: mov Dtm,Bf00 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf01 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf02 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf03 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf04 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf05 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf10 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf11 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf12 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf20 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf21 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf22 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf23 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf24 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf30 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf31 ; data memory
lcall wr_mem
inc Admm
;
rddnmj: mov A,Adbm ; AT24C16 write address
lcall adrtx
mov A,Admm ; address memory
lcall dtarx
inc Adbm
mov A,Adbm ; AT24C16 read address
lcall adrtx
lcall dtarx ; terima data
mov Bf00,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf01,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf02,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf03,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf04,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf05,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf10,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf11,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf12,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf20,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf21,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf22,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf23,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf24,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf30,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf31,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Dtgl,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Dbln,A

```

```

;
lcall givack ; beri ack
lcall dtarx ; terima data
mov Dthn,A
lcall givack ; beri ack
lcall dtarx ; terima data
;
mov Djam,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Dmnt,A
lcall i2cstp ; i2c stop
ret
;
wr djns: inc Admm
mov Dtm,Bf00 ; data memory
lcall wr_mem
ret
;
rd djns: mov A,Adbm ; AT24C16 write address
lcall adrtx
mov A,Admm ; address memory
lcall dtatx
inc Adbm
mov A,Adbm ; AT24C16 read address
lcall adrtx
lcall dtarx ; terima data
mov Bf00,A
lcall i2cstp ; i2c stop
ret
;
wr dtkr: inc Admm
mov Dtm,Bf00 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf01 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf02 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf03 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf04 ; data memory
lcall wr_mem
ret
;
rd dtkr: mov A,Adbm ; AT24C16 write address
lcall adrtx
mov A,Admm ; address memory
lcall dtatx
inc Adbm
mov A,Adbm ; AT24C16 read address
lcall adrtx
lcall dtarx ; terima data
mov Bf00,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf01,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf02,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf03,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf04,A
lcall i2cstp ; i2c stop
ret
;
wr dbrt: inc Admm
mov Dtm,Bf00 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf01 ; data memory
lcall wr_mem
ret
;
inc Admm
mov Dtm,Bf02 ; data memory
lcall wr_mem
ret
;
rd dbrt: mov A,Adbm ; AT24C16 write address
lcall adrtx
mov A,Admm ; address memory
lcall dtatx
inc Adbm
mov A,Adbm ; AT24C16 read address
lcall adrtx
lcall dtarx ; terima data
mov Bf00,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf01,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf02,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf03,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf04,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf05,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf06,A
lcall i2cstp ; i2c stop
ret
;

```

```

wrdong: inc Admm
mov Dtm,Bf00 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf01 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf02 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf03 ; data memory
lcall wr_mem
inc Admm
mov Dtm,Bf04 ; data memory
lcall wr_mem
ret

;
rddong: mov A,Adbm ; AT24C16 write address
lcall adrtx
mov A,Admm ; address memory
lcall dtatx
inc Adbm
mov A,Adbm ; AT24C16 read address
lcall adrtx
lcall dtarx ; terima data
mov Bf00,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf01,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf03,A
lcall givack ; beri ack
lcall dtarx ; terima data
mov Bf04,A
lcall i2cstp ; i2c stop
ret

;
wr_mem: mov A,Adbm ; AT24C16 write address
lcall adrtx
mov A,Admm ; address hardware
memory
lcall dtatx
mov A,Dtm ; data memory
lcall dtatx
lcall i2cstp ; i2c stop
lcall wt_wr
ret

;
wt_wr: mov Dly0,#000
mov Dly1,#025
wtwr: lcall delay0
dijnz Dly1,wtwr
ret

;
adrtx: lcall i2cstr ; send address
lcall putbit
ret

;
dtatx: lcall putbit ; send data
ret

;
dtarx: lcall getbit ; get data
ret

;
putbit: mov R6,#8
putbt: RLC A
mov ISDA,C
setb ISCL
clr ISCL
dijnz R6,putbt
setb ISDA
lcall getack
ret

;
getbit: mov R6,#8
getbt: setb ISCL
mov C,ISDA
RLC A
clr ISCL
dijnz R6,getbt
setb ISDA
ret

;
getack: setb ISDA ; tunggu ack
setb ISCL ; \
ackbit: mov C,ISDA ; | D=1, C=1
je ackbit ; | tunggu D=0,
C=0 ; /
ret

;
givack: clr ISDA ; kirim ack
setb ISCL ; \
clr ISCL ; | D=0, C=1, C=0, D=1
setb ISDA ; /

```

```

ret
;
i2cstr: setb ISCL ; i2c start
setb ISDA ;\
clr ISDA ; | C=1, D=1,
D=0, C=0
clr ISCL ;/
ret
;
i2cstp: clr ISDA ; i2c stop
setb ISCL ;\ D=0, C=1, D=1, C=0
setb ISDA ;/
clr ISCL
lcall delay0
ret
;
nilai: mov B,#100
div AB
lcall wr_chr
mov A,B
nil: mov B,#10
div AB
lcall wr_chr
mov A,B
lcall wr_chr
ret
;
line1: mov P0,#080h
lcall w_ins
ret
;
line2: mov P0,#0C0h
lcall w_ins
ret
;
tulis: clr A
lcall wr_chr
inc DPTR
djnz Char,tulis
ret
;
wr_chr: movc A,@A+DPTR
mov P0,A
lcall w_chr
ret
;
w_ins: clr Enbl
clr Rest
setb Enbl
clr Enbl
lcall delay0
ret
;
w_chr: clr Enbl
setb Rest
setb Enbl
clr Enbl
lcall delay0
ret
;
lcd_in: mov Dly3,#1
lcall delay3
mov P0,#01h ; Display Clear
lcall w_ins
mov P0,#38h ; Function Set
lcall w_ins
mov P0,#0Eh ; Display On, Cursor,
Blink
lcall w_ins
mov P0,#06h ; Entry Mode
lcall w_ins
mov P0,#02h ; Cursor Home
lcall w_ins
ret
;
lcdclr: mov P0,#01h ; Display Clear
ret
;
lcall w_ins
lcall delay0
lcall delay0
lcall delay0
ret
;
scnkpd: mov R0,#10
lcall delay0
col1: mov P1,#1111110b
mov A,P1
c1b1: cjne A,#11101110b,c1b2
mov R0,#1
c1b2: cjne A,#11011110b,c1b3
mov R0,#2
c1b3: cjne A,#10111110b,c1b4
mov R0,#3
c1b4: cjne A,#01111110b,col2
mov R0,#13
;
col2: mov P1,#11111101b
mov A,P1
c2b1: cjne A,#11101101b,c2b2
mov R0,#4
c2b2: cjne A,#11011101b,c2b3
mov R0,#5
c2b3: cjne A,#10111101b,c2b4
mov R0,#6
c2b4: cjne A,#01111101b,col3
mov R0,#14
;
col3: mov P1,#11111011b
mov A,P1
c3b1: cjne A,#11101101b,c3b2
mov R0,#7
c3b2: cjne A,#11011011b,c3b3
mov R0,#8
c3b3: cjne A,#10111011b,c3b4
mov R0,#9
c3b4: cjne A,#01111011b,col4
mov R0,#15
;
col4: mov P1,#11110111b
mov A,P1
c4b1: cjne A,#11100111b,c4b2
mov R0,#11
c4b2: cjne A,#11010111b,c4b3
mov R0,#0
c4b3: cjne A,#10110111b,c4b4
mov R0,#12
c4b4: cjne A,#01110111b,back
mov R0,#16
back: ret
;
tg_tkn: lcall scnkpd
tgtkn0: cjne R0,#16,tgtn1
ljmp tg_tkn
tgtkn1: cjne R0,#15,tgtn2
ljmp tg_tkn
tgtkn2: cjne R0,#14,tgtn3
ljmp tg_tkn
tgtkn3: cjne R0,#13,tgtn4
ljmp tg_tkn
tgtkn4: cjne R0,#10,tgtn5
ljmp tg_tkn
tgtkn5: ret
;
tg_lps: lcall scnkpd
cjne R0,#10,tg_lps
ret
;
delay0: djnz Dly0,delay0
ret
;
delay1: lcall scnkpd
cjne R0,#12,dely10
ljmp tlnama

```

```
dely10: cjne R0,#13,dely11
        ljmp setime
dely11: cjne R0,#14,dely12
        ljmp laporn
dely12: djnz Dly1,delay1
        ret
;
dely2: mov Dly2,#20
dely2: lcall delay1
        djnz Dly2,dely2
        ret
;
dely3: lcall delay0
        djnz Dly1,delay3
        djnz Dly3,delay3
        ret
```

```
tpnama: DB 'Dwi Joko W'
tpnims: DB 'NIM:0810632005'
tpjurs: DB 'Teknik Elektro'
tpuniv: DB 'Brawijaya Malang'
tpilpm: DB 'Tulis Pembeli'
tpjnem: DB 'JENIS EMAS'
tpjns0: DB 'Kalung'
tpjns1: DB 'Liontin'
tpjns2: DB 'Gondel'
tpjns3: DB 'Gelang'
tpjns4: DB 'Cincin'
tpjns5: DB 'Giwang'
tpjns6: DB 'Peniti'
niltkr: DB 'Nilai Tukar Emas'
        DB 'Rp.-----'
brtems: DB 'BERAT EMAS'
        DB 'gram'
harga: DB 'HARGA'
        DB 'Rp.-----'
ongkos: DB 'ONGKOS'
        DB 'Rp.-----'
total: DB 'TOTAL HARGA'
        DB 'Rp.-----'
savmem: DB 'Save Memory 000'
membhs: DB 'Memory Habis'
prnslp: DB 'PRINT SLIP'
```

```
prjns0: DB 'Kalung'
prjns1: DB 'Liontin'
prjns2: DB 'Gondel'
prjns3: DB 'Gelang'
prjns4: DB 'Cincin'
prjns5: DB 'Giwang'
prjns6: DB 'Peniti'
```

```
prtKem: DB 'Toko Emas'
        DB 'Bintang Terang'
```

```
tnggal: DB 'Tanggal'
namapb: DB 'Nama'
prjns: DB 'Jenis'
prbrt: DB 'Berat'
prnltk: DB 'Nil Tukar : Rp.'
prhrg: DB 'Harga : Rp.'
prong: DB 'Ongkos : Rp.'
prtll: DB 'Total : Rp.'
pesan: DB 'Perhiasan Emas Jual'
        DB 'Surat Diikut Sertakan'
```

```
tpsttm: DB 'Setting Time'
tpklbr: DB 'Kalibrasi'
tpadbr: DB 'ADC Berat'
tpfkr: DB 'Zero Faktor'
tpcnen: DB 'Can/Ent'
tpplwt: DB 'Please Wait...'
tplprn: DB 'Laporan'
tphlpl: DB 'Hapus Laporan'
angka: DB '0123456789'
```

```
klmpk0: DB '1' ; 2
klmpk1: DB 'ABC2' ; 4
klmpk2: DB 'DEF3' ; 4
klmpk3: DB 'GH4' ; 4
klmpk4: DB 'JKL5' ; 4
klmpk5: DB 'MNO6' ; 4
klmpk6: DB 'QRS7' ; 5
klmpk7: DB 'TUV8' ; 4
klmpk8: DB 'WXYZ9' ; 5
klmpk9: DB '0' ; 2
```

```
petamm:
DB
0A0h,000h,0A0h,032h,0A0h,064h,0A0h,096h,0A0h,0C8h
DB
0A2h,000h,0A2h,032h,0A2h,064h,0A2h,096h,0A2h,0C8h
DB
0A4h,000h,0A4h,032h,0A4h,064h,0A4h,096h,0A4h,0C8h
DB
0A6h,000h,0A6h,032h,0A6h,064h,0A6h,096h,0A6h,0C8h
DB
0A8h,000h,0A8h,032h,0A8h,064h,0A8h,096h,0A8h,0C8h
DB
0AAh,000h,0AAh,032h,0AAh,064h,0AAh,096h,0AAh,0C8h
DB
0ACh,000h,0ACh,032h,0ACh,064h,0ACh,096h,0ACh,0C8h
DB
0AEh,000h,0AEh,032h,0AEh,064h,0AEh,096h,0AEh,0C8h
;
end
```



LAMPIRAN III

- III-a** *Datasheet Load Cell*
- III-b** *Datasheet LM324*
- III-c** *Datasheet PCF8591*
- III-d** *Datasheet AT89S52*
- III-e** *Datasheet DS1307*
- III-f** *Datasheet AT24C16*
- III-g** *Datasheet LCD M1632*
- III-h** *Datasheet SMBUS And I²C Bus Design*
- III-i** *Datasheet Parallel Port*



Filename: 3PERENCANAAN DAN PEMBUATAN ALAT PENIMBANG
PERHIASAN EMAS DILENGKAPI DENGAN PENCETAK SERTIFIKAT
PERHIASAN.docx

Directory: C:\Users\Dwi Joko Waluyo\Desktop

Template: C:\Users\Dwi Joko
Waluyo\AppData\Roaming\Microsoft\Templates\Normal.dotm

Title:

Subject:

Author: DWIJOKOWALOYO

Keywords:

Comments:

Creation Date: 01-Jan-11 13:03:00

Change Number: 6

Last Saved On: 01-Jan-11 15:22:00

Last Saved By: Dwi Joko Waluyo

Total Editing Time: 60 Minutes

Last Printed On: 01-Jan-11 15:23:00

As of Last Complete Printing

Number of Pages: 104

Number of Words: 28,044 (approx.)

Number of Characters: 159,852 (approx.)

