

15.260/H/02



**PERENCANAAN DAN REALISASI
EVALUASI TINGKAT GIZI BALITA
BERBASIS PC**

TUGAS AKHIR



RSE
621.398 4
Sri
P-1
2000

Oleh :

SRI DAMAYANTI
NRP. 2297 030 005

M. TAMAM NAFIRI
NRP. 2297 030 073

**BIDANG STUDI KOMPUTER KONTROL
PROGRAM DIPLOMA III
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2000**

17 30 ass-

29/01/2001
H

PERENCANAAN DAN REALISASI EVALUASI TINGKAT GIZI BALITA BERBASIS PC^r

TUGAS AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Menyelesaikan Studi
Pada
Bidang Studi Komputer Kontrol
Program Studi Diploma III
Jurusan Teknik Elektro
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
S u r a b a y a

Mengetahui / Menyetujui



Dosen Pembimbing

PRI ARIEF SARDJONO, ST, MT.

NIP. 132 135 222

SURABAYA
NOPEMBER, 2000



ABSTRAK

Dalam tugas akhir ini dirancang dan direalisasikan sistem berbasis PC untuk aplikasi di Puskesmas yang berupa evaluasi tingkat gizi balita.

Sistem yang dikembangkan terdiri dari perangkat keras dan perangkat lunak. Perangkat keras berupa modul pengukuran berat badan. Perangkat lunak yang dikembangkan adalah evaluasi tingkat gizi balita serta perangkat lunak untuk koordinasi perangkat keras.

Potensiometer geser digunakan sebagai sensor berat modul pengukuran berat badan dengan meletakkan pada mekanik timbangan manual yang telah dimodifikasi. Sedang perangkat lunak dikembangkan dengan menggunakan program aplikasi delphi developer 5. Digunakan pula metode regresi untuk menganalisa dan mengolah hasil pengukuran sesuai dengan respon sensor berat yang digunakan.

Sistem yang dikembangkan ini diharapkan dapat meningkatkan efisiensi waktu dan tenaga serta menyediakan informasi yang akurat dan dibutuhkan, sehingga pelayanan kesehatan ibu dan anak serta sistem administrasi/ koordinasi puskesmas dapat berfungsi dengan lebih baik.



KATA PENGANTAR

Alhamdulillah, kami panjatkan puji syukur kehadirat Allah SWT atas rahmat dan izinNya yang telah terlimpahkan, sehingga tugas akhir ini dapat terselesaikan dengan baik. Semoga salawat dan salam selalu terlimpahkan kepada junjungan kita Nabi Besar Muhammad saw, yang dengan perjuangan dan pengorbanan beliau, kita telah mendapat jalan dan petunjuk yang benar.

Tugas akhir yang berjudul :

“PERENCANAAN DAN REALISASI EVALUASI TINGKAT GIZI

BALITA BERBASIS PC “

ini merupakan salah satu syarat bagi kelulusan studi diploma III di Bidang Studi Komputer Kontrol, Program Studi Diploma III Jurusan Teknik Elektro Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya.

Kami mengucapkan terima kasih yang sebesar-besarnya kepada beberapa pihak yang telah berperan dan sangat membantu dalam proses pembuatan tugas akhir ini, terutama kepada :

1. Bpk Ir.Rusdhianto,EAK MT, selaku Ketua Program Studi Diploma III Teknik Elektro Komputer Kontrol , Fakultas Teknologi Industri, Institut Teknologi 10 Nopember Surabaya.
2. Bpk. Ir H.M.Djoko Santoso dan Ir.Henny Utami selaku dosen wali.
3. Bpk. Tri Arief Sardjono ST,MT , selaku dosen pembimbing tugas akhir.



4. Bpk. Prof. Dr. Mauridhi H., Bpk. Eko Mulyanto ST,
Bpk. Maulidyanto ST, selaku dosen penguji tugas akhir kami.
5. Sahabat-sahabatku terkasih Computer Control angkatan '97 atas
bantuan dan dorongannya.

Tiada gading yang tak retak, begitu pula dengan penulisan tugas akhir ini, tentu masih banyak kekurangan kesalahan yang semua itu tidak lepas dari keterbatasan dalam diri kami sendiri, namun harapan kami semoga tugas akhir ini mampu memberi sumbangan yang bermanfaat terhadap siapa saja yang membacanya.

Surabaya, Nopember 2000

Penulis



UCAPAN TERIMA KASIH

Terima Kasih Yanti Kepada :

Allah SWT,
Bapak & Ibu yang selalu memberikan dukungan serta do'a dan selalu tabah dalam kehidupan yang keras ini.
Kakakku, mas Arief, mbak Rakhma & mas Denny,
adikku Ninok & Ujang serta my little nephew Fifi yang selalu menghibur.
Sahabat setiaiku : Siti Mariyam, Harits, Ratih

Terima Kasih Tamam Kepada :

Allah SWT, Atas segala kelancaran yang diberikan pada pengerjaan tugas akhir ini.
Seluruh keluargaku (Ayah, Ibu, Adik dan yang lain) atas segala dorongan yang telah diberikan.

Terima Kasih Kami Berdua Kepada :

- **Crew Gebang Lor 81** : 'Bapa' Sigit & Eddy 'Lonte' atas kurungannya, 'Mbah' Syarief atas konsultasi software, Ari 'Acong' yang mau nemanin main kartu, Nanang 'Zemox' & Rudy 'Bothak Kecil', Komeng, Huda 'Bothak Besar', Kokom, Ocop 'Kartu', Ko an, Mahfud, Antok, Afrien, Edi 'Pencer', 'Kopral' Wakhid atas motivasinya, Irfan atas kerelaan karpetnya. Bulik's family (GL 81), atas segala dukungan yang diberikan.
- **Kêthek Gank** : Alf 'Bobi' Wijaya & Gede 'Gedhek', Adi 'Tompel' & Sugik 'Beruk', Teguh 'Mbulet' & Agung 'Budheng', atas bantuan moril n' spirituil-nya.
- **Workshop Team Gebang Lor 24** : Ben-ben 'Suglewo', Harun 'Gendruwo', Dhony 'Kambing', Arke 'Bhowo', Tedy 'Kunam', Cahyo 'Lesus', Sarwo 'Stepper', Hilmi 'Koplo', Widi 'Sucakil', Slamet 'Mendolo', Sigit 'Pedro', 'Gaduk' Sodik, Umarlda, Daeng 'Bogel', 'Tompel' Cahyo, 'Bapak' Edi & Arek EI'97, Sofyan 'Ndas', Usman, 'Gundhul Mayak' Fauzi, atas bantuan tenaga, alat 'n tempatnya.
- **Mitra 99 (rename Bu Tedjo)** : 'Jamie' Jamrotin, Dhawang, Zein 'Cuklek', 'Bagong' Bagus, Farid 'Fowler', Fanani, Om 'Porno' Purnomo, Bimo 'Flip-Flop', Mr.Dony Agus, Nanang 'Virus', Lukman 'Papa Be-Ef', Didik 'Suroto', Didik 'Kimpot', Teezar 'Hombreng', Anam 'Curul', for motivations, books, consultations, etc.



-
- **Keputih Gang Makam** : 'Dodol' Pranadyanta, Yogi 'Beruang Kutub' , Adi 'Turangga' , Arief, atas segala bantuan yang diberikan.
 - **CC '97 Quin's** : Anitha, Eza, Dyah, Aniek, Dilla (pernah ada), Ita, Rensy, Kartini, Ari, Yuli, Yuni, Faridah, Veveka, Erna, Isbandiyah, Hadiana, Femi, Lia', Nonie, Trias, Deeta, Wiwik, Dien, Titin, Dewi, Lulus. Thank's for your's Motivation.
 - **CC '97 Rest** : 'Ongkek' Agung, 'Pak' Yadi, Ferry, 'Lik' Wahyu, Tatox's, Tafid, Aris, Karno, Ryan, David, Rahmat 'Jenglot', Lukman 'Mandul',.



DAFTAR ISI

	Hal.
HALAMAN JUDUL	i
HALAMAN PENGESAHAN	ii
ABSTRAK	iii
KATA PENGANTAR	iv
UCAPAN TERIMA KASIH	vi
DAFTAR ISI	viii
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiii
DAFTAR GRAFIK	xiv
BAB I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Maksud dan Tujuan	3
1.3 Permasalahan	3
1.4 Pembatasan Masalah	4
1.5 Metodologi	4
1.6 Sistematika Pembahasan	5
BAB II. TEORI PENUNJANG	6
2.1. Pendahuluan	6
2.2. <i>Programable Peripheral Interface (PPI) 8255</i>	6



2.3.	<i>Analog To Digital (ADC) 0804</i>	13
2.4.	Sensor Berat.....	15
2.5.	Pengkondisian Sinyal.....	17
2.5.1.	Non Inverting Amplifier.....	17
2.5.2.	Filter.....	18
BAB III.	PERENCANAAN DAN REALISASI ALAT	20
3.1.	Blok Diagram Sistem.....	20
3.2.	Perencanaan Perangkat Keras (<i>Hardware</i>)	21
3.2.1.	Perencanaan PPI Card 8255.....	21
3.2.2.	Unit Rangkaian Sensor Berat Dan Pengkondisian Sinyal Timbangan Balita.....	24
3.2.3.	Unit ADC 0804.....	26
3.3.	Perencanaan dan Pembuatan Perangkat Lunak.....	28
BAB IV.	PENGUJIAN DAN PENGUKURAN ALAT	30
4.1.	Pengujian Modul Pengukuran.....	30
4.1.1.	Pengujian PPI 8255.....	30
4.1.2.	Pengujian Sensor Berat.....	34
4.1.3.	Pengujian Rangkaian Pengkondisian Sinyal... 34	
4.1.4.	Pengujian Rangkaian ADC 0804.....	35
4.2.	Pengujian Perangkat Lunak (<i>Software</i>).....	43



BAB V.	PENUTUP.....	48
	5.1 Kesimpulan.....	48
	5.2 Saran.....	49

DAFTAR PUSTAKA

LAMPIRAN

1. Listing program
2. Pengambilan Data ADC
3. PPI
4. Data Sheet ADC 0804
5. Data Sheet LF356

**DAFTAR GAMBAR**

Gambar 1.1. Grafik Kartu Menuju Sehat (KMS) Balita.....	1
Gambar 2.1. Konfigurasi pin PPI 8255.....	7
Gambar 2.2. Format Control Word PPI 8255.....	10
Gambar 2.3. Format Bit SET/ RESET PPI 8255.....	11
Gambar 2.4. Konfigurasi Pin-pin ADC 0804.....	13
Gambar 2.5 Mekanik Timbangan	15
Gambar 2.6. Struktur Dasar Potensiometer Geser.....	16
Gambar 2.7. Rangkaian <i>Non Inverting Amplifier</i>	17
Gambar 2.8. Rangkaian LPF -40 db/decade.....	18
Gambar 3.1. Diagram Blok Modul Timbangan Balita Digital.....	20
Gambar 3.2. Blok Diagram PPI Card.....	23
Gambar 3.3. Rangkaian Sensor Berat dan Pengkondisi Sinyal Timbangan Balita.....	24
Gambar 3.4. Konfigurasi Pin IC LF 356	25
Gambar 3.4. Rangkaian ADC 0804 (8 bit)	28
Gambar 3.5. Flowchart Alat.....	29
Gambar 4.1. Cara Pengujian PPI 8255.....	32
Gambar 4.3. Diagram Blok Proses Penentuan Persamaan Regresi.....	35
Gambar 4.4 Diagram Blok Pengujian Berat Setelah Melalui Proses Regresi..	38
Gambar 4.5. Tampilan Menu Utama.....	44
Gambar 4.6. Tampilan Menu Data ID Balita.....	45



Gambar 4.7. Tampilan Menu Ruang Periksa.....	45
Gambar 4.8. Tampilan Menu Grafik Evaluasi Tingkat Gizi.....	46
Gambar 4.9. Tampilan Menu Timbangan.....	46
Gambar 4.10. Tampilan Menu Laporan Penimbangan Bulanan.....	47



DAFTAR TABEL

Tabel 1.1. Status Gizi Bayi dan Balita.....	1
Tabel 1.2. Evaluasi Pertumbuhan bayi dan Balita.....	2
Tabel 2.1 Operasi Dasar PPI 8255.....	8
Tabel 2.2. Formasi Sinyal untuk Inisialisasi.....	9
Tabel 3.1. Pemilihan port PPI.....	22
Tabel 4.1. Hasil keluaran tiap Port PPI 8255 'Logika 1'.....	33
Tabel 4.2. Hasil Pengujian untuk Penentuan Regresi.....	36
Tabel 4.3. Perhitungan Regresi untuk $0 \leq y_1 \leq 1$ atau $7 \leq x \leq 19$	38
Tabel 4.4. Perhitungan Regresi untuk $1 \leq y_1 \leq 3$ atau $19 \leq x \leq 45.33$	39
Tabel 4.5. Perhitungan Regresi untuk $3 \leq y_1 \leq 4$ atau $45.33 \leq x \leq 61.66$	39
Tabel 4.6. Perhitungan Regresi untuk $4 \leq y_1 \leq 6$ atau $61.66 \leq x \leq 85.33$	39
Tabel 4.7. Perhitungan Regresi untuk $6 \leq y_1 \leq 7$ atau $85.33 \leq x \leq 98.66$	40
Tabel 4.8. Perhitungan Regresi untuk $7 \leq y_1 \leq 9$ atau $98.66 \leq x \leq 126.33$	40
Tabel 4.9. Perhitungan Regresi untuk $9 \leq y_1 \leq 10$ atau $126.33 \leq x \leq 140$	40
Tabel 4.10. Perhitungan Regresi untuk $10 \leq y_1 \leq 11$ atau $140 \leq x \leq 159.33$	41
Tabel 4.11. Perhitungan Regresi untuk $11 \leq y_1 \leq 13$ atau $140 \leq x \leq 159.33$	41
Tabel 4.12. Perhitungan Regresi untuk $13 \leq y_1 \leq 14$ atau $182.66 \leq x \leq 194.66$	41
Tabel 4.13. Perhitungan Regresi untuk $14 \leq y_1 \leq 16$ atau $194.6 \leq x \leq 223$	42
Tabel 4.14. Perhitungan Regresi untuk $16 \leq y_1 \leq 17$ atau $223 \leq x \leq 226.66$	42
Tabel 4.15. Perhitungan Regresi untuk $17 \leq y_1 \leq 19$ atau $226.66 \leq x \leq 236$	42
Tabel 4.16. Perhitungan Regresi untuk $19 \leq y_1 \leq 20$ atau $236 \leq x \leq 240$	43
Tabel 4.17. Perbandingan Data Setelah Regresi dan Data Referensi.....	43



DAFTAR GRAFIK

Grafik 4.1. Kurva data ADC – Berat Referensi Modul Timbangan Balita 37



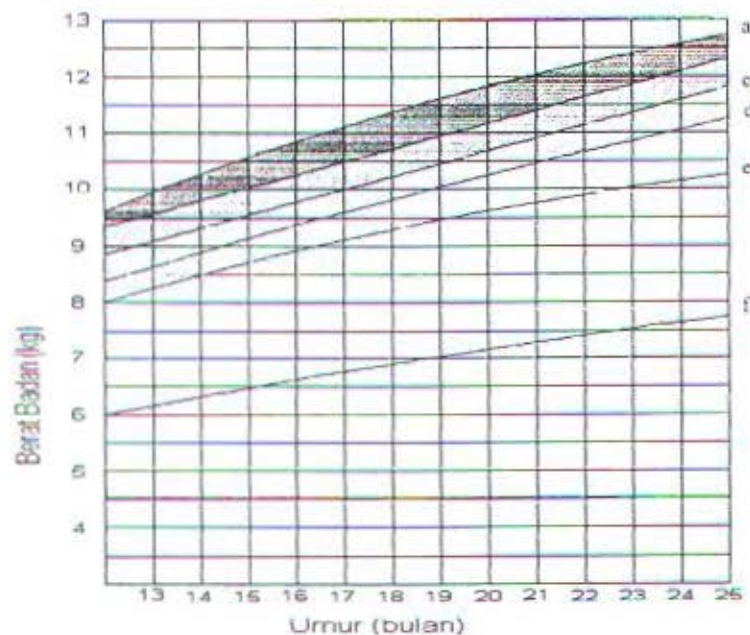
BAB I

PENDAHULUAN

1.1. Latar Belakang

Masalah gizi membutuhkan penanganan yang segera sedangkan salah satu cara yang paling mudah digunakan untuk menilai keadaan gizi bayi dan balita ialah dengan cara mengukur berat badannya.

Di Indonesia penilaian status gizi bayi dan balita mengacu pada grafik yang terdapat pada Kartu Menuju Sehat Balita (KMS balita).



Gambar 1.1. Grafik Kartu Menuju Sehat (KMS) Balita

Pada gambar 1.1 terlihat bahwa pada KMS Balita terdapat sumbu tegak lurus yang terletak di sebelah kiri yang menunjukkan berat badan anak dalam



kilogram (Kg) dan sumbu datar yang terletak pada bagian bawah yang menunjukkan umur anak dalam bulan 0 sampai 5 tahun. Disamping itu, terdapat garis evaluasi yang dipakai untuk menentukan status gizi. Garis tersebut menghubungkan sisi kiri ke sisi kanan dan terdiri dari 3 garis yaitu garis yang paling atas (a), garis yang terletak di tengah (e) dan garis yang terletak paling bawah (f). Di antara garis yang paling atas (a) dan yang ditengah (e) terdapat 4 (empat) daerah yang mempunyai gradasi warna yang dimulai dari daerah yang berwarna paling gelap di bagian paling atas dan berangsur menjadi lebih terang (a-b, b-c, c-d, d-e).

Penilaian keadaan gizi bayi dan balita tergantung posisi hasil pengukuran berat badan dan umur terhadap garis evaluasi pada grafik KMS seperti yang terlihat pada tabel 1.1.

Tabel 1.1. Status Gizi Bayi dan Balita

Posisi hasil Pengukuran	Status Gizi
Pada daerah yang berwarna (a-e)	Baik
Antara garis tengah dan garis paling bawah (e-f)	Kurang baik
Di bawah garis evaluasi yang paling bawah	Buruk

Untuk mengevaluasi pertumbuhan bayi dan balita hendaknya dilakukan penimbangan setiap bulan secara teratur. Hasil pengukuran berat badan anak pada grafik KMS pada tiap penimbangan dihubungkan sehingga terbentuk garis pertumbuhan. Evaluasi pertumbuhan bayi dan balita dapat dilihat pada tabel 1.2.



Tabel 1.2. Evaluasi Pertumbuhan bayi dan Balita

Arah Garis Pertumbuhan	Pertumbuhan Anak
<ul style="list-style-type: none">• Mengikuti daerah yang berwarna senada (a, b, c, d, e).• Sejajar dengan garis evaluasi (di antara a dan f).• Mengarah ke daerah yang lebih gelap (di antara a dan f).	Baik
<ul style="list-style-type: none">• Mengarah ke daerah yang lebih terang.• Berat badan tidak naik pada penimbangan 3 kali berturut-turut.	Kurang/ Tidak baik

1.2. Maksud dan Tujuan

Sistem yang akan dirancang dan direalisasikan ini berupa modul timbangan balita yang diharapkan dapat memberikan diagnosa dini adanya kekurangan gizi dan evaluasi status gizi yang lebih akurat.

1.3. Permasalahan

Adanya beberapa hambatan yang sering kita jumpai seperti kehilangan Kartu Menuju Sehat (KMS) Balita yang digunakan sebagai acuan dalam melakukan evaluasi status gizi akan memperlambat diagnosa kekurangan gizi pada bayi dan balita yang bersangkutan.

Permasalahan yang timbul adalah :

1. Bagaimana merancang suatu modul timbangan balita yang mempunyai fungsi untuk mengukur berat badan bayi dan balita yang dihubungkan dengan PC.



2. Bagaimana mengevaluasi status gizinya dan menyimpan hasil pengukuran dan evaluasinya ke basis data bayi dan balita yang diukur.

1.4. Pembatasan Masalah

Perencanaan dan realisasi evaluasi tingkat gizi balita berbasis PC merupakan gabungan hardware dan software yang dikontrol dengan suatu rangkaian interfacing. Mengingat banyaknya parameter yang perlu diperhatikan maka dalam tugas akhir ini dilakukan pembatasan masalah sebagai berikut :

1. Perencanaan hardware hanya meliputi modul timbangan balita yang dihubungkan dengan PC.
2. Perencanaan software meliputi hal-hal yang terdapat pada Kartu Menuju Sehat (KMS) Balita dengan menggunakan bahasa pemrograman Delphi 5.
3. Dalam perencanaan dan perealisasi ini tidak dibahas sistem mekanik secara mendetil.

1.5. Metodologi

Dalam perencanaan dan realisasi evaluasi tingkat gizi bayi dan balita berbasis PC ini dilakukan dengan pendekatan hardware dan software. Secara hardware dilakukan dengan mempelajari sistem mekanik timbangan yang diinterfacekan ke PC, sedangkan pendekatan software dengan mempelajari Delphi 5 untuk mengevaluasi status gizi dan menyimpan hasil pengukurannya.



1.6. Sistematika Laporan

Dalam menyusun tugas akhir ini pembahasannya disajikan dalam rangkaian sebagai berikut :

- BAB I : Membahas tentang pendahuluan yang terdiri dari latar belakang, maksud dan tujuan, pembatasan masalah, metodologi, dan sistematika laporan tugas akhir ini.
- BAB II : Membahas tentang teori penunjang yang berhubungan dengan komponen penyusun perencanaan dan realisasi evaluasi tingkat gizi bayi dan balita berbasis PC.
- BAB III : Membahas tentang perencanaan hardware dan software yang terdiri dari cara kerja dari masing-masing bagiannya secara keseluruhan.
- BAB IV : Membahas tentang pengujian dan sekaligus pengukuran alat.
- BAB V : Merupakan penutup yang terdiri dari kesimpulan dari tugas akhir ini serta saran-saran yang membangun.



BAB II

TEORI PENUNJANG

2.1. Pendahuluan

Modul timbangan balita ini berfungsi untuk mengukur berat badan bayi dan balita, mengevaluasi status gizinya dan menyimpan hasil pengukuran dan evaluasinya ke basis data bayi/ balita yang diukur.

Modul timbangan balita terdiri dari bagian mekanik dan bagian elektronik. Bagian mekanik yaitu bagian yang menerima beban, menggunakan timbangan bayi/ balita manual yang telah dimodifikasi sedemikian rupa sehingga sensor berat dapat dipasang pada bagian tuas yang bergerak. Bentuk dasar dari mekanik timbangan manual dan sensor berat yang digunakan potensiometer geser akan dijelaskan pada bab ini.

Pada bagian ini diuraikan juga tentang PPI 8255, ADC 0804 serta rangkaian pengkondisian sinyal.

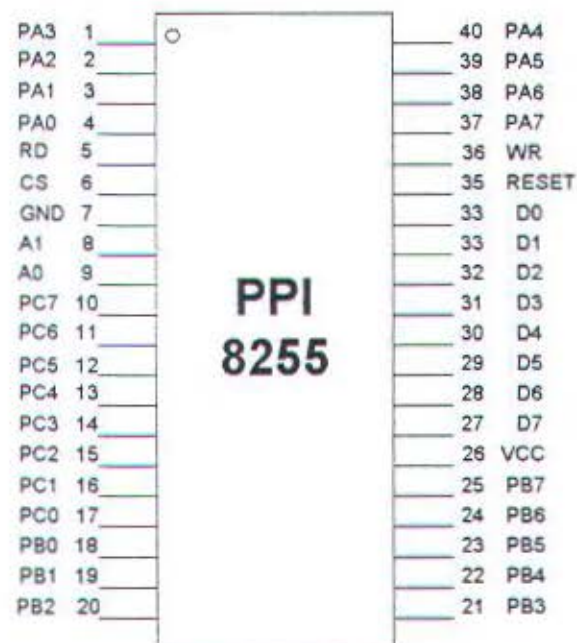
2.2. Programmable Peripheral Interface (PPI) 8255

PPI 8255 adalah suatu piranti *Parallel Input/Output* dalam satu chip serbaguna yang dapat diprogram fungsi input/outputnya. Awalnya, PPI 8255 dibuat oleh *Intel* untuk digunakan bersama dengan mikroprosesor buatan *Intel*. Tetapi karena komponen ini berbasis *standart bus* dan relatif mudah tata cara



perangkaiannya maka kebanyakan tipe mikroprosesor lain juga dapat memanfaatkannya.

PPI 8255 mempunyai 24 pin I/O yang terdiri dari 3 Port, yaitu Port A (8 pin), Port B (8 pin) dan Port C (8 pin). Port A dan Port C pada PC7 s.d. PC4 tergabung dalam Grup Kontrol A, sedang Port B dan Port C dari PC3 s.d. PC0 tergabung dalam Grup Kontrol B. PPI 8255 ini dapat dioperasikan dalam 3 mode, yaitu Mode 0, Mode 1 dan Mode 2. Konfigurasi pin PPI 8255 ditunjukkan dalam gambar 2.1.



Gambar 2.1. Konfigurasi pin PPI 8255

Deskripsi tentang fungsi pin 8255 adalah sebagai berikut :

PA7 - PA0 : Terminal I/O Port A.

PB7 - PB0 : Terminal I/O Port B.



PC7 – PC0 : Terminal I/O Port C.

\overline{CS} : *Chip Select* (input, active low).

Bila \overline{CS} di-set '0' maka PPI sedang dihubungi CPU.

\overline{RD} : *Read* (input, active low).

\overline{RD} di-set '0' menunjukkan bahwa CPU sedang membaca data dari PPI 8255.

\overline{WR} : *Write* (input, active low).

\overline{WR} di-set '0' menunjukkan bahwa CPU sedang menulis data ke PPI 8255.

A0 & A1 : Port Select 0 & Port Select 1 (input).

Kombinasi A0 & A1 digunakan untuk memilih Port mana yang bekerja.

+5V & GND : Terminal Tegangan Supply & Ground.

Tabel 2.1 Operasi Dasar PPI 8255

A1	A0	\overline{RD}	\overline{WR}	\overline{CS}	Operasi yang dibentuk
0	0	0	1	0	READ : Port A ke Data Bus
0	1	0	1	0	READ : Port B ke Data Bus
1	0	0	1	0	READ : Port C ke Data Bus
0	0	1	0	0	WRITE : Data Bus ke Port A
0	1	1	0	0	WRITE : Data Bus ke Port B
1	0	1	0	0	WRITE : Data Bus ke Port C
1	1	1	0	0	WRITE : Data Bus ke <i>Control Word Register</i> (<i>INITIALIZATION</i>)
x	x	x	x	1	Data Bus 8255 berada dalam kondisi <i>tristate</i> .
1	1	0	1	0	Kondisi Illegal
x	x	1	1	0	Data Bus 8255 berada dalam kondisi <i>tristate</i> .



PPI 8255 mengenal 3 mode operasi, yaitu :

1. Mode 0 = *Basic Input/Output*.
2. Mode 1 = *Strobed Input/Output*, dan
3. Mode 2 = *Bi-Directional Bus*.

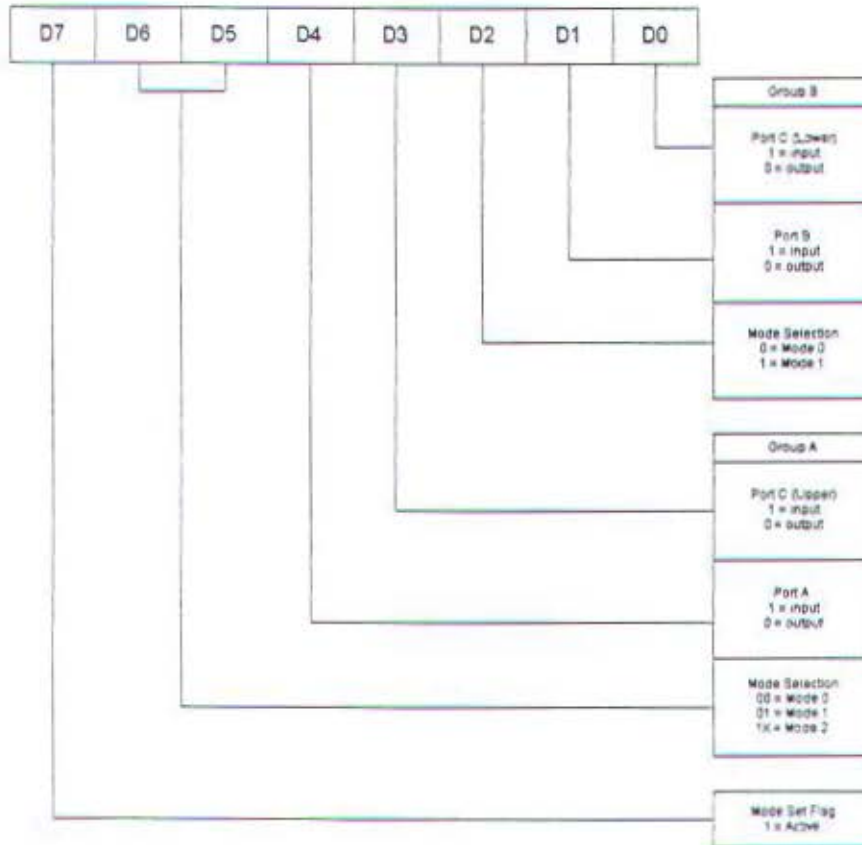
Mode-mode operasi ini dapat dipilih dengan memberikan '*Control Word*' pada saat inisialisasi. Formasi sinyal untuk inisialisasi ditunjukkan oleh tabel 2.2.

Tabel 2.2. Formasi Sinyal untuk Inisialisasi

A1	A0	\overline{RD}	\overline{WR}	\overline{CS}	Operasi yang dibentuk
1	1	1	0	0	WRITE : Data Bus ke <i>Control Word Register</i> (<i>INITIALIZATION</i>)

Control Word yang diberikan cukup sekali pada awal PPI 8255 ini akan diaktifkan. Selain itu selama program sedang berjalan, PPI 8255 sewaktu-waktu dapat diubah mode operasinya dengan memberikan sebuah *Control Word* lagi. Tersedianya fasilitas ini memungkinkan PPI 8255 dapat dimanfaatkan untuk melayani berbagai keperluan *interfacing* dengan program-program *subroutine* yang berbeda dalam sebuah sistem rangkaian.

Pada PPI 8255 bila input RESET diaktifkan ('1'), semua Port akan di-set pada input Mode (24 pin Port menjadi *high impedance*). Setelah RESET, otomatis PPI akan menggunakan semua Port-nya sebagai Input Port tanpa perlu diinisialisasi (diberikan *Control Word*) lagi.



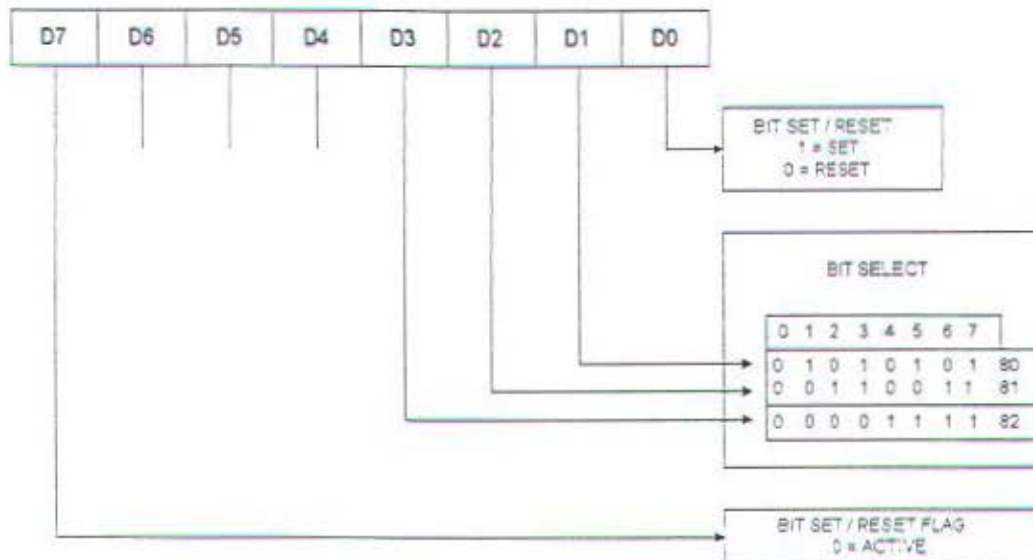
Gambar 2.2. Format Control Word PPI 8255

Gambar 2.2 memperlihatkan formasi *Control Word* dengan fungsi yang berbeda-beda. Formasi ini dibedakan dalam dua bagian utama, yaitu yang mendefinisikan fungsi I/O pada Grup A (Port A & Port C Upper) dan yang mendefinisikan fungsi I/O pada Grup B (Port B & Port C Lower).

Bit D7 berfungsi untuk menunjukkan mode set dari *Control Word*. Bila D7 = '1', *Control Word* yang diberikan adalah untuk menge-set FLAG. Artinya set untuk inisialisasi fungsi terminal-terminal. Sedang bila D7 = '0', *Control Word*



yang diberikan adalah untuk langkah Set/ Reset Port C dengan format seperti yang ditunjukkan gambar 2.3 di bawah ini .



Gambar 2.3. Format Bit SET/ RESET PPI 8255

Bit Set/ Reset ini juga dapat diberikan sewaktu-waktu di dalam program untuk mengubah mode operasi Port C sesuai dengan yang diinginkan. Pada operasi mode 1 atau mode 2 biasanya disediakan juga sinyal-sinyal kontrol yang dapat digunakan sebagai interrupt ke CPU. Dalam hal ini, sesuai dengan tujuannya, Port C dapat dimanfaatkan. Seringkali dalam pemrograman, sinyal-sinyal interrupt ini harus diubah statusnya dari enable ke disable, atau sebaliknya. Maka dengan adanya Bit Set/Reset ini programmer menjadi lebih mudah dalam mengoperasikan atau membatalkan permintaan interrupt berbagai peralatan Input/Output melalui program tanpa mengubah struktur dari sistem interrupt.



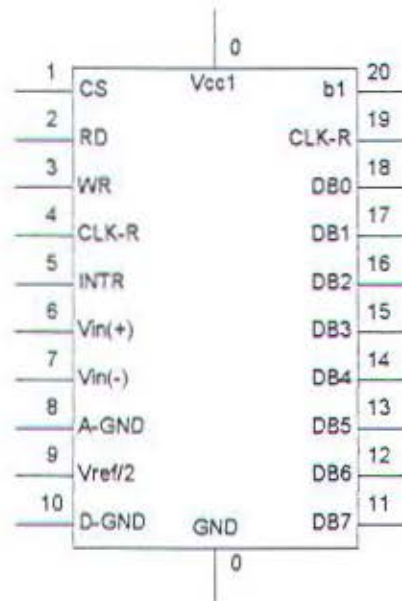
Definisi dari mode-mode PPI 8255 adalah sebagai berikut :

1. Mode 0 : Yaitu mode operasi dimana semua port dapat dioperasikan sebagai masukan atau keluaran. Ketika port akan difungsikan sebagai masukan atau keluaran tanpa strobe, maka 8255A diinisialisasi pada mode 0.
2. Mode 1 : Yaitu mode operasi dengan menggunakan port A dan port B yang bekerja sebagai masukan ataupun keluaran, dengan menggunakan sebagian jalur-jalur pada port C, dilengkapi dengan jabat tangan otomatis (automatic handshaking). Jika diinginkan port A dan port B bekerja sebagai masukan dan keluaran strobe maka port akan diinisialisasi pada mode 1.
3. Mode 2 : Yaitu mode operasi dengan menggunakan port A yang bekerja sebagai masukan dan keluaran dua arah (bidirectional), juga untuk menerima dan mengeluarkan data, dilengkapi dengan sistem handshaking. Dalam mode 2 hanya port A yang dapat diinisialisasi.



2.3 Analog To Digital (ADC) 0804

ADC 0804 didesain secara khusus agar dapat digunakan pada sistem yang berbasis mikroprocessor. Oleh karena itu memiliki kaki-kaki atau pin-pin yang diperlukan oleh mikroprocessor yaitu *Chip select* (\overline{CS}), *Read* (\overline{RD}), *Write* (\overline{WR}). Outputnya dapat langsung dihubungkan ke data bus, hal ini dimungkinkan karena di dalam ADC 0804 pada saluran outputnya terdapat *Octal Tristate Latch*. Output *Latch* ini di-*enable* dengan memberikan logika nol pada kedua kaki \overline{CS} dan kaki \overline{RD} (pada saat ini CPU sedang membaca data dari ADC 0804). Konfigurasi dari kaki-kaki atau pin-pin ADC 0804 seperti terlihat di bawah ini.



Gambar 2.4. Konfigurasi Pin-pin ADC 0804

ADC 0804 ini mempergunakan catu daya tunggal sebesar 5 Volt yang diberikan pada kaki 20 untuk V_{cc} dan kaki 10 untuk ground. *Clock* generator dari ADC ini dapat dibangkitkan dengan menambahkan komponen-komponen seperti



yang diuraikan pada bab 3 (lihat gambar 3.3). Misalnya resistor yang dipasang pada kaki 19 dan kapasitor pada kaki 4 kemudian dihubungkan ke ground. Hal ini sesuai dengan besarnya frekuensi yang ditentukan dengan rumus :

$$f = \frac{1}{(1,1 \times R \times C)}$$

Dimana : $R = 10 \text{ K}\Omega$

$C = 150 \text{ pF}$

Dengan demikian frekuensi dari ADC 0804 ini sebesar 640 KHz. Sedang frekuensi luar dihubungkan ke kaki 4 (Clock in). Tegangan input analog masuk melalui kaki Vin (+). Hal ini dikarenakan beda tegangan pada input analog tersebut tidak boleh melebihi Vcc-nya. Kaki 9 (Vref/2) dihubungkan ke tegangan yang besarnya setengah tegangan (V) referensinya, tegangan input Vref/2 ini perlu untuk akurasi dari hasil output yang tepat.

Kaki 5 adalah kaki input interrupt yang mempunyai sign aktif low, output ini akan bernilai logika low apabila digunakan untuk menginterrupt CPU sebagai tanda untuk mengawali program pelayanan interrupt dan dapat digunakan untuk mengaktifkan port dari PPI 8255.

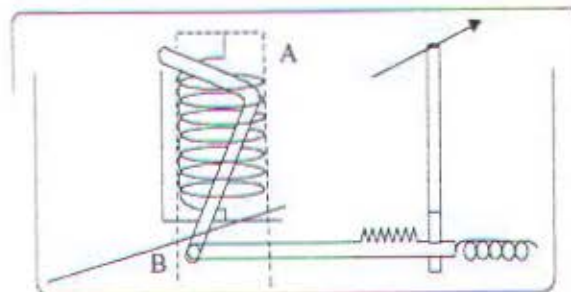
Kaki 8 (A-ground) adalah ground input analog, sedangkan kaki 10 (D-ground) adalah ground untuk rangkaian digital. Hal ini akan memberikan resolusi yang lebih baik dari kedua rangkaian dalam sistem ini.

2.4 Sensor Berat

Sensor berat yang digunakan pada timbangan balita adalah potensiometer geser. Sensor berat ini dipasang pada timbangan badan manual (bath scale) yang telah dimodifikasi bagian mekaniknya.

a. Bagian Mekanik

Timbangan balita memiliki bentuk mekanik yang serupa sebagaimana ditunjukkan pada gambar 2.5.



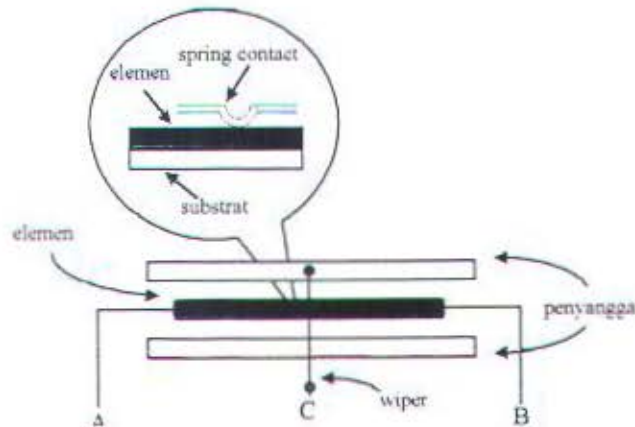
Gambar 2.5. Mekanik Timbangan

Secara sederhana prinsip kerja mekanik timbangan manual adalah sebagai berikut : beban yang berada di atas timbangan akan diteruskan melalui penyangga A menuju pelat tumpuan yang dihubungkan dengan pegas timbangan. Hal ini akan menyebabkan pegas dan ujung pelat tumpuan akan tertarik ke bawah. Perubahan ini akan diteruskan oleh penggerak B sebagai translasi (ke kanan) yang selanjutnya akan menggerakkan jarum penunjuk berat.

Gerakan penggerak B inilah yang akan digunakan untuk menggerakkan potensiometer geser. Peletakan sensor potensiometer geser ini disesuaikan dengan bentuk mekanik yang ada mengingat adanya sedikit perbedaan pada beberapa jenis timbangan badan manual.

b. Potensiometer Geser

Potensiometer geser mempunyai tiga kaki, dua kaki tetap dan satu kaki bergerak yang disebut dengan wiper sebagaimana ditunjukkan gambar 2.6.



Gambar 2.6 Struktur Dasar Potensiometer Geser

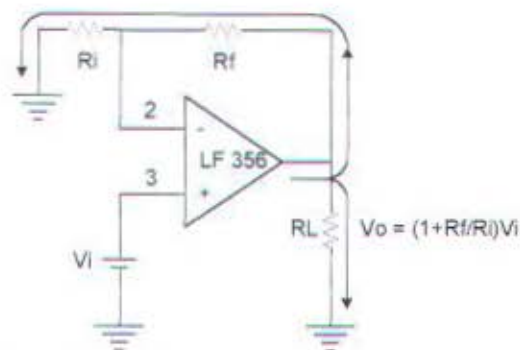
Resistansi dari potensiometer geser merupakan fungsi dari posisi wiper yang bergerak sepanjang elemen resistansi. Elemen resistansi bisa terbuat dari carbon composition layer, metal oxide film atau wired wound assembly. Resistansi total R_t diukur dari terminal tetap A-B. Terminal C (wiper) ditempatkan pada satu titik yang membagi dua bagian elemen yaitu AC dan CB. Perbandingan dari dua resistansi AC dan CB ini tergantung dari posisi wiper C di sepanjang elemen A-B.

2.5 Pengkondisian Sinyal

Rangkaian pengkondisi sinyal yang dirancang dan direalisasikan pada sensor potensiometer geser terdiri dari rangkaian penguat (*amplifier*) dan rangkaian *filter*. Rangkaian yang digunakan adalah *Non Inverting Amplifier* sedangkan rangkaian *filter* adalah *low pass filter* (LPF).

2.5.1 Non Inverting Amplifier

Keluaran dari sensor biasanya sangat kecil dan membutuhkan penguat. Untuk keperluan tersebut maka dibutuhkan *amplifier* (penguat). Rangkaian penguat yang dipilih adalah rangkaian *Non Inverting Amplifier*. Hal ini sesuai dengan keluaran sensor dan polaritas masukan ADC. Rangkaian *Non Inverting Amplifier* ditunjukkan pada gambar 2.7.



Gambar 2.7. Rangkaian *Non Inverting Amplifier*

Dari gambar 2.7 dapat diturunkan persamaan untuk mencari penguatan dan tegangan keluaran yang dihasilkan. Diasumsikan bahwa op-amp yang digunakan adalah ideal sehingga tegangan diferensial antara kedua input op-amp adalah nol dan tidak ada arus yang masuk kedua inputnya.



$$E_d = V_3 - V_2 \cong 0 \quad \dots\dots\dots (1)$$

$$V_2 = V_3 = V_i \quad \dots\dots\dots (2)$$

Arus yang melewati R_i adalah $I = V_i/R_i$ dengan arah sesuai arah panah dan tegangan yang timbul pada R_f adalah :

$$V_{R_f} = I \times R_f \quad \dots\dots\dots (3)$$

$$V_{R_f} = \frac{V_i}{R_i} \times R_f = \frac{R_f}{R_i} \times V_i \quad \dots\dots\dots (4)$$

Tegangan output dari non inverting amplifier ini adalah :

$$V_o = V_{R_f} + V_i \quad \dots\dots\dots (5)$$

$$V_o = \frac{R_f}{R_i} \times V_i + V_i \quad \dots\dots\dots (6)$$

$$V_o = \left[\frac{R_f}{R_i} + 1 \right] \times V_i \quad \dots\dots\dots (7)$$

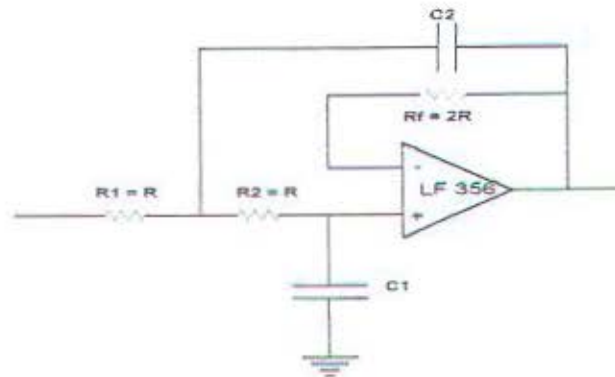
Penguatannya adalah :

$$A_{CL} = \frac{V_o}{V_i} = \frac{R_f}{R_i} + 1 \quad \dots\dots\dots (8)$$

2.5.2 Filter

Jenis filter yang digunakan pada sistem ini adalah low pass filter aktif -40 dB/decade. Gambar 2.7 menunjukkan low pass filter aktif -40 dB/dec.





Gambar 2.8. Rangkaian LPF -40 db/decade

Adapun prosedur perancangan LPF -40 dB/dec ini adalah sebagai berikut :

1. Menentukan frekuensi cut off ω_c atau f_c .
2. Menentukan nilai C_1 diantara 100 pF dan 0.1 μ F, $C_2=2C_1$.
3. Menghitung nilai $R = 0.707/\omega_c C$, $R_f = 2R$.

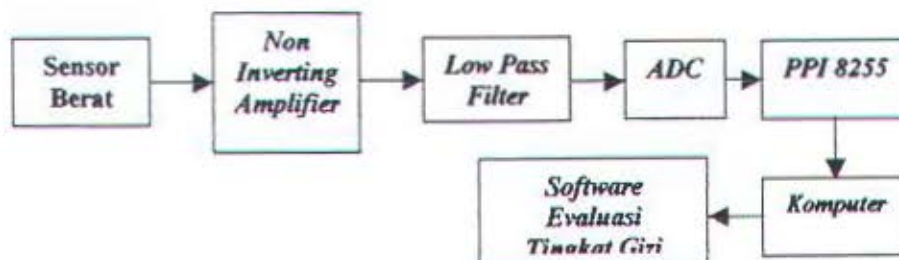
BAB III

PERENCANAAN DAN REALISASI ALAT

Dalam bab ini, sistem yang dikembangkan terdiri dari perangkat keras (*hardware*) dan perangkat lunak (*software*). Perangkat keras berupa modul timbangan balita sedangkan perangkat lunak yang dikembangkan adalah evaluasi tingkat gizi balita serta perangkat lunak untuk koordinasi perangkat keras yang dibutuhkan.

3.1 Blok Diagram Sistem

Diagram blok dari modul timbangan balita yang dirancang dan direalisasikan ditunjukkan pada gambar 3.1.



Gambar 3.1. Diagram Blok Modul Timbangan Balita Digital

Adapun bagian-bagian dari modul timbangan balita antara lain :

1. Rangkaian sensor berat, yang dipasang pada timbangan badan manual.
2. Rangkaian *non inverting amplifier*, sebagai penguat sinyal dari sensor berat.
3. Rangkaian *filter*, untuk mengurangi noise yang timbul.
4. Rangkaian ADC 0804, sebagai pengkonversi sinyal analog menjadi data digital.



3.2. Perencanaan Perangkat Keras (*Hardware*)

3.2.1. Perencanaan PPI Card 8255

Perencanaan PPI Card sebagai rangkaian interfacing karena selain mudah dalam perancangan *hardware* dan *software* juga banyak di pasaran dan harganya terjangkau. IC ini terdiri dari 24 pin *I/O* dimana terbagi menjadi tiga group masing-masing 8 buah yaitu Port A, Port B, Port C.

Pada PPI 8255 ini terdapat pin chip select (*CS*) yang aktif low, yang berfungsi untuk mengaktifkan PPI jika pada pin tersebut diberikan pulsa low ('0'). Karena alamat port pada peralatan *I/O* tidak semuanya digunakan dalam desain suatu komputer, tetapi hanya 10 bit terendah yang digunakan (A0-A9) maka diperlukan pemilihan alamat tertentu, sehingga pemakaian alamat *I/O* port untuk peralatan yang direncanakan tidak terjadi pada alamat yang sama dengan alamat *I/O* port yang lain. Apabila terjadi tumpang tindih, akan berakibat fatal bagi system board maupun pada interface – interface yang bersangkutan. Untuk itu dipilih alamat dari prototype Card (300H – 31FH) sebagai daerah kerja PPI.

Pada PPI 8255 ini terdapat rangkaian pendukung yang diperlukan yaitu rangkaian dekoder yang digunakan untuk memberikan sinyal pada chip select dari PPI untuk mengaktifkan PPI pada alamat tertentu. Juga terdapat dip switch yang berfungsi untuk mengatur alamat dari PPI. Apabila data dari alamat yang berasal dari address bus tidak sesuai dengan kombinasi dari dip switch maka PPI tidak akan aktif dan demikian pula sebaliknya PPI akan aktif apabila data dari address bus sesuai dengan kombinasi dari dip switch.



Address bus yang digunakan untuk dekoder dimulai dari A2 – A9 sehingga PPI dapat aktif pada semua alamat dengan syarat alamat tersebut tidak melebihi kombinasi maksimal dari dip *switch* (dalam bentuk biner) dan alamat yang diberikan adalah alamat yang disediakan oleh komputer untuk peralatan input output. Sedangkan A0 dan A1 berfungsi untuk memilih port dari PPI dengan perincian sebagai berikut :

Tabel 3.1. Pemilihan port PPI

A0	A1	Port yang digunakan
0	0	Port A
0	1	Port B
1	0	Port C
1	1	Port Control Word

Sedangkan diagram blok PPI dapat dilihat pada gambar 3.2.

- Inisialisasi PPI 8255

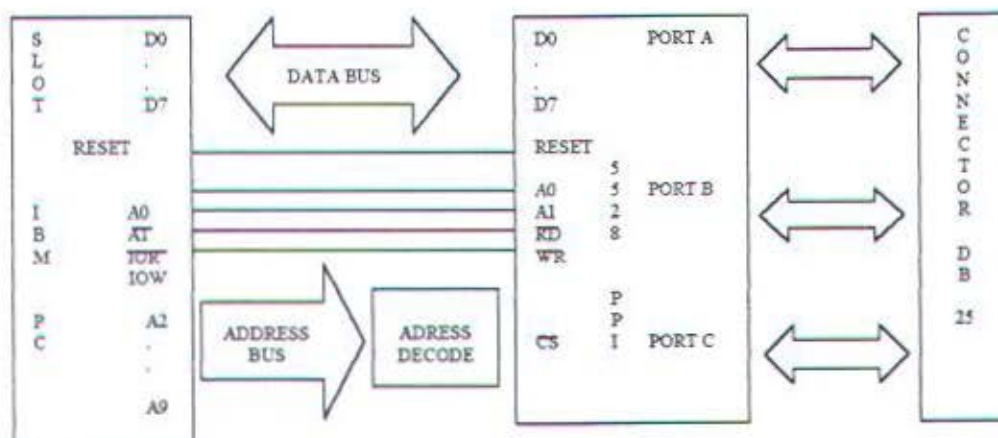
Untuk menentukan mode kerja dari PPI 8255 kita harus memberi tanda aktifnya PPI dan mendefinisikan input/output pada masing-masing port PPI 8255. PPI 8255 memiliki 3 buah port yang masing-masing memiliki 8 saluran. Beberapa hal yang harus diperhatikan dalam penggunaan PPI 8255 :

- Pin $\overline{\text{Read}}$ / $\overline{\text{Write}}$, Reset dan $\overline{\text{CS}}$ harus disiapkan yang digunakan pada *Hardware*nya
- Register control PPI 8255 yang terdiri dari 8 bit biner harus diberikan pada port data alamat register control.



- Alamat-alamat yang digunakan untuk port *I/O* dan juga control word PPI 8255 harus sesuai dengan definisi alamat pada perangkat lunak, yaitu pada prototype card.

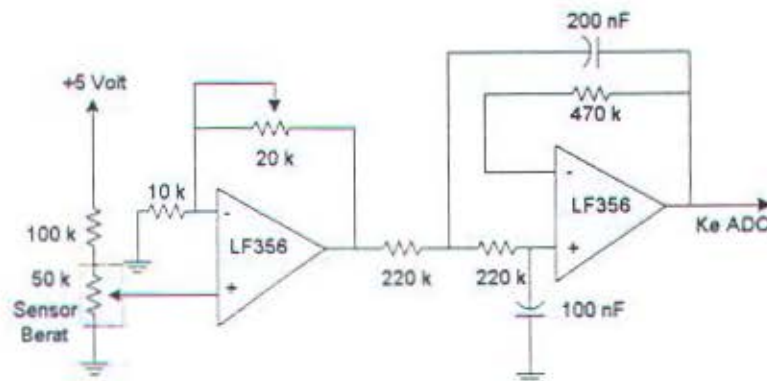
Dengan pemberian control word dan data register control akan dapat ditentukan fungsi dari setiap port PPI 8255. Karena sebuah PPI 8255 hanya memerlukan empat alamat maka daerah kerja dari prototype card dapat dipilih, mengingat urutan 2 bit terakhir (A0 – A1), misalnya 300H – 303H, 304H – 307H, 308H – 30BH, ..., 31CH – 31FH. Jadi ada delapan kemungkinan alamat yang dapat dipilih untuk PPI Card . Hal ini memungkinkan untuk menggunakan PPI 8255 lebih dari satu dengan memilih alamat yang berbeda.



Gambar 3.2. Blok Diagram PPI Card

3.2.2. Unit Rangkaian Sensor Berat dan Pengkondisian Sinyal Timbangan Balita

Sensor berat yang digunakan adalah potensiometer geser dengan nilai resistansi 50 K Ω . Gambar 3.3 menunjukkan gambar rangkaian sensor dan pengkondisi sinyal untuk timbangan balita. Sensor berat ini dipasang pada bagian tuas yang bergerak untuk memutar jarum penunjuk pada timbangan manual.



Gambar 3.3. Rangkaian Sensor Berat dan Pengkondisi Sinyal Timbangan Balita

Gerakan tuas ini akan menggerakkan potensiometer geser dan besar pergeserannya yang berupa tegangan akan dikonversikan ke dalam berat (Kg). Keluaran dari sensor berat akan dikuatkan oleh *non inverting amplifier* menggunakan IC LF 356 dengan konfigurasi pin seperti pada gambar 3.4 yang penguatannya dapat diatur dengan VR 20 K Ω .

Dengan persamaan :

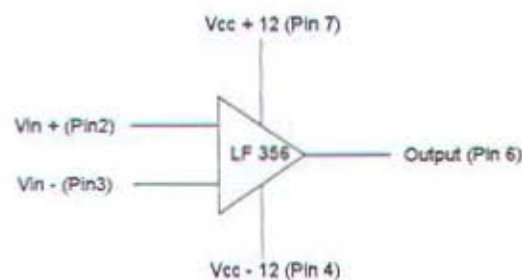
$$A_{CL} = V_o = \frac{V_o}{V_i} = \frac{R_f}{R_i} + 1$$



didapatkan perhitungan penguatannya adalah sebagai berikut :

$$A_{CL} = \frac{R_f}{R_i} + 1$$
$$A_{CL} = \frac{0,5 / d \cdot 20 \text{ K}}{10 \text{ K}} + 1$$

Sehingga penguatan yang diperoleh adalah 1X sampai dengan 3X.



Gambar 3.4. Konfigurasi Pin IC LF 356

Untuk menghindari adanya *noise* digunakan *low pass filter* -40 dB/dec menggunakan LF 356 dengan frekuensi *cut off* 5 Hz. Penentuan harga komponennya adalah sebagai berikut :

1. Frekuensi *cut off* 5 Hz, jadi $\omega_c = 2 \times \pi \times f_c = 31.415 \text{ rad/det}$.
2. Memilih C_1 dengan besar 100 nF, $C_2 = 2C_1 = 200 \text{ nF}$.
3. $R_1 = 0.707 (\omega_c \times c) = 0.707 / (31.415 \times 100 \text{ nF}) = 225 \text{ K}\Omega$ (nilai yang ada di pasaran 220 $\text{K}\Omega$)
4. $R_2 = R_1 = 220 \text{ K}\Omega$, $R_f = 2 R_1 = 450 \text{ K}\Omega$ (nilai yang ada di pasaran 470 $\text{K}\Omega$)

Selanjutnya keluaran dari *filter* siap untuk dikonversikan menjadi data digital.



3.2.3. Unit ADC 0804

Rangkaian ADC ini bertujuan untuk mengubah analog yang berasal dari input analog menjadi bentuk besaran digital 8 bit. ADC yang digunakan adalah ADC type 0804 yang mempunyai 20 kaki. ADC 0804 ini menggunakan metode *successive approximation*. ADC ini mempunyai 2 masukan yaitu $V_{in} = V_{in(+)} - V_{in(-)}$, dan mempunyai range 0 sampai 5 Volt maka untuk keperluan ini $V_{in(-)}$ digroundkan.

Untuk melakukan proses konversi maka \overline{CS} diberikan masukan *low*, sehingga masukan data terbaca oleh ADC 0804. Kemudian \overline{WR} diberi masukan *low* karena adanya pulsa *low* yang diberikan pada \overline{WR} ini ADC akan memulai konversi. Setelah proses konversi ini selesai, \overline{WR} dikembalikan pada posisi *high* dan \overline{RD} diberikan masukan *low* sehingga keluaran digital muncul pada keluaran ADC.

Clock dari ADC memakai pembangkit internal sehingga hanya dibutuhkan sebuah resistor dan kapasitor yang dihubungkan pada jalur CLK-R (kaki 19) dan CLK-IN (kaki 4). Dengan mengambil nilai $R = 10\text{ K}\Omega$ dan $C = 150\text{ pF}$, maka diperoleh frekuensi *clock* sebesar :

$$f = \frac{1}{(1,1 \times R \cdot C)}$$

$$f = \frac{1}{(1,1 \times 10\text{K} \cdot 150\text{ pF})}$$

$$f = 600\text{ Hz}$$



Frekuensi ini masih dalam batas range ADC 0804 yaitu antara 100 KHz sampai 1000 KHz.

ADC 0804 mempunyai 8 output digital yaitu $D_0 - D_7$ pada pin 11 - 18 output digital yang di hasilkan berada pada range 2^n atau pada range 0 sampai dengan 255.

A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
0	0	0	0	0	0	0	0

Sampai dengan

A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
1	1	1	1	1	1	1	1

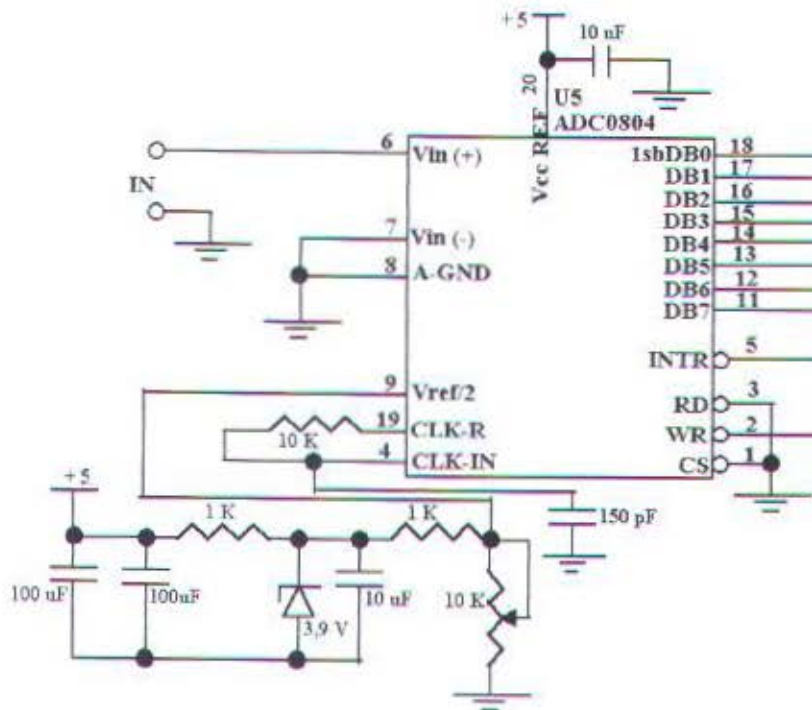
Input analog yang berupa tegangan antara 0 - 5 Volt akan di konversikan menjadi output digital dengan range 0 - 255 . Yang berarti setiap 1 data ADC

diwakili oleh $\frac{5}{256} = 0.0196$ Volt.

Modul timbangan balita dibuat dengan berat maksimal 20 Kg (sesuai dengan timbangan bayi). Untuk mendapatkan hasil 20 Kg dari data dengan range 0 - 255 maka diperlukan konversi seperti dibawah ini :

$$\frac{20 \text{ Kg}}{256} = 0.078 \text{ Kg}$$

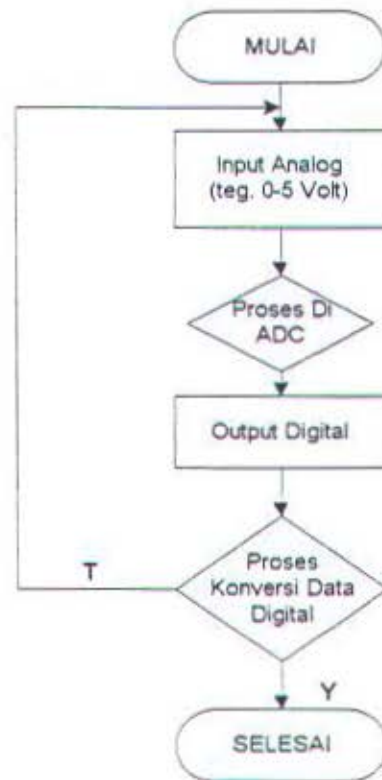
nilai diatas digunakan untuk pengambilan data sementara yang kemudian diregresi untuk mendapatkan data yang mendekati hasil sesungguhnya. Untuk keperluan *interfacing* digunakan *software* Delphi 5.0 dari Borland cooperation.



Gambar 3.4. Rangkaian ADC 0804 (8 bit)

3.3 Perencanaan Dan Pembuatan Perangkat Lunak.

Pada perencanaan dan pembuatan *software* ini, compiler yang digunakan adalah Borland Delphi 5.0, yang dibuat pada IBM PC. Dengan tujuan untuk mengendalikan peralatan (*Hardware*) guna memenuhi spesifikasi yang direncanakan. Adapun spesifikasi yang direncanakan dilukiskan pada gambar 3.5, yaitu flowchart alat. Data output dari *hardware* akan diolah pada *software*, data output tersebut akan disalurkan melalui port (alamat). Port yang digunakan adalah port A (A₀ – A₇) dari PPI card 8255.



Gambar 3.5. Flowchart Alat



BAB IV

PENGUKURAN DAN PENGUJIAN ALAT

Pengujian dilakukan terhadap modul pengukuran dan modul perangkat lunak. Pengujian modul pengukuran dilakukan setelah hasil uji coba terhadap semua perangkat keras (ADC, sensor berat, op-amp, *filter*) telah sesuai dengan spesifikasi yang diharapkan.

4.1. Pengujian Modul Pengukuran

4.1.1. Pengujian PPI 8255

Pengujian pada PPI 8255 ini menentukan berfungsi tidaknya alat. Dalam pengujian alat ini dilakukan dengan memberikan program sederhana. Cara kerja dari program tersebut adalah sebagai berikut :

1. Memberikan *Control Word* Register untuk inisialisasi PPI 8255. Nilai *Control Word* yang diberikan memfungsikan semua port sebagai output.
2. Mengaktifkan semua port dengan memberikan nilai pada tiap kaki yang berhubungan dengan port A, port B dan port C.



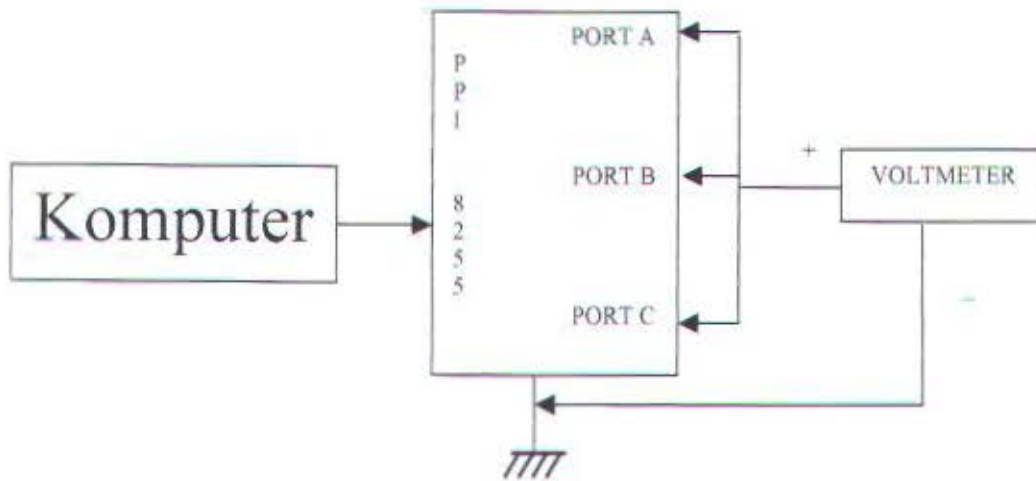
Bentuk program tersebut adalah sebagai berikut :

Program uji_PPI 8255 :

```
Uses crt, dos ;  
  
Begin  
  
Port [S303]:= $80;  
  
Port [S302]:= 255;  
  
Port [S301]:= 255;  
  
Port [S300]:= 255;  
  
Readln;  
  
Port [S302]:= 0;  
  
Port [S301]:= 0;  
  
Port [S300]:= 0;  
  
Readln;  
  
End.
```



Untuk mengetahui hasil pengujian digunakan Voltmeter, jika ada tegangan antara 5,8 antara 5,4 menunjukkan logika '1' (high) dan jika tegangan menunjukkan 0 sampai 1,4 menunjukkan logika '0' (low). Kaki positif dihubungkan ke kaki port PPI 8255 dan kaki negatif ke ground PPI 8255 seperti yang ditunjukkan oleh gambar 4.1.



Gambar 4.1. Cara Pengujian PPI 8255

Sebelum mengukur besarnya tegangan yang ada pada masing-masing port pada PPI 8255 terlebih dahulu pengecekan pemasangan pada port komputer apakah sudah benar posisinya, yang kemudian dilanjutkan pemasangan kabel voltmeter, kemudian program yang dibuat dijalankan terlebih dahulu, dan terakhir pencatatan besarnya tegangan pada tiap-tiap port.



Tabel 4.1. Hasil keluaran tiap Port PPI 8255 'Logika 1'

PORT	NOMOR KAKI	MASUKAN	KELUARAN (V)
A	PA0	1	3.73
	PA1	1	6.73
	PA2	1	6.21
	PA3	1	6.38
	PA4	1	3.72
	PA5	1	3.5
	PA6	1	3.75
	PA7	1	5.24
B	PB0	1	3.74
	PB1	1	6.22
	PB2	1	6.65
	PB3	1	4.49
	PB4	1	4.29
	PB5	1	6.95
	PB6	1	3.91
	PB7	1	4.8
C	PC0	1	6.48
	PC1	1	6.25
	PC2	1	3.9
	PC3	1	6.16
	PC4	1	6.20
	PC5	1	6.5
	PC6	1	6.02
	PC7	1	5.92

Pada hasil pengukuran ternyata hasilnya berkisar antara 3.73 volt sampai dengan 6.73 volt ini menunjukkan bahwa pengukuran masih berkisar harga dari logik '1' yang bernilai antara 3 sampai dengan 7 volt.

Sebelum mengukur besarnya tegangan pada logik '0' yang ada pada masing-masing port pada PPI 8255 terlebih dahulu pengecekan pemasangan pada port komputer apakah sudah benar posisinya, yang kemudian dilanjutkan



pemasangan kabel voltmeter, kemudian program yang dibuat dijalankan terlebih dahulu, dan terakhir pencatatan besarnya tegangan pada tiap-tiap port.

4.1.2. Pengujian Sensor Berat

Untuk pengujian sensor berat digunakan potensiometer geser dengan range antara 0 – 50 K Ω (nilai dari pabrik). Kabel positif multimeter dihubungkan dengan kaki 1 potensiometer geser dan kabel negatif multimeter dihubungkan dengan kaki 2 potensiometer geser, maka didapatkan data bahwa range potensiometer geser sesungguhnya adalah 0 – 49 K Ω . Dan keluaran dari sensor berat sangat kecil sekali yaitu ± 1 Volt.

4.1.3. Pengujian Rangkaian Pengkondisian Sinyal

Rangkaian pengkondisian sinyal yang terdiri dari *non inverting amplifier* dan *filter* digunakan sebagai penguatan sinyal keluaran dari sensor berat yang begitu kecil agar didapatkan keluaran yang dibutuhkan. Berdasarkan tegangan yang dibutuhkan maka digunakan komponen-komponen dengan nilai seperti yang ditunjukkan pada gambar 3.2 (Bab 3).

Dengan memakai persamaan (8) didapatkan penguatan sebesar $3x$.

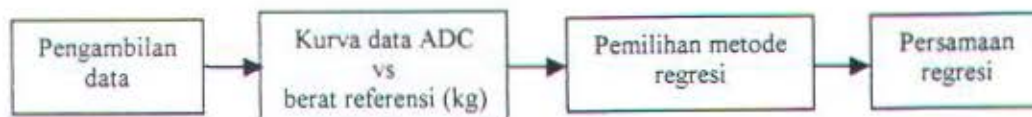
$$A_{CL} = \frac{R_f}{R_i} + 1$$

$$A_{CL} = \frac{0,5 / d 20 K}{10 K} + 1$$

4.1.4. Pengujian Rangkaian ADC 0804

Untuk pengujian rangkaian ADC 0804 ada 2 tahap pengujian yang dilakukan yaitu pengujian untuk penentuan regresi dan pengujian hasil penimbangan berat setelah dilakukan proses regresi.

Penggunaan regresi ini dilakukan mengingat sensor yang digunakan kurang linier. Gambar 4.3 menunjukkan diagram blok proses penentuan persamaan regresi.



Gambar 4.3. Diagram blok proses penentuan persamaan regresi

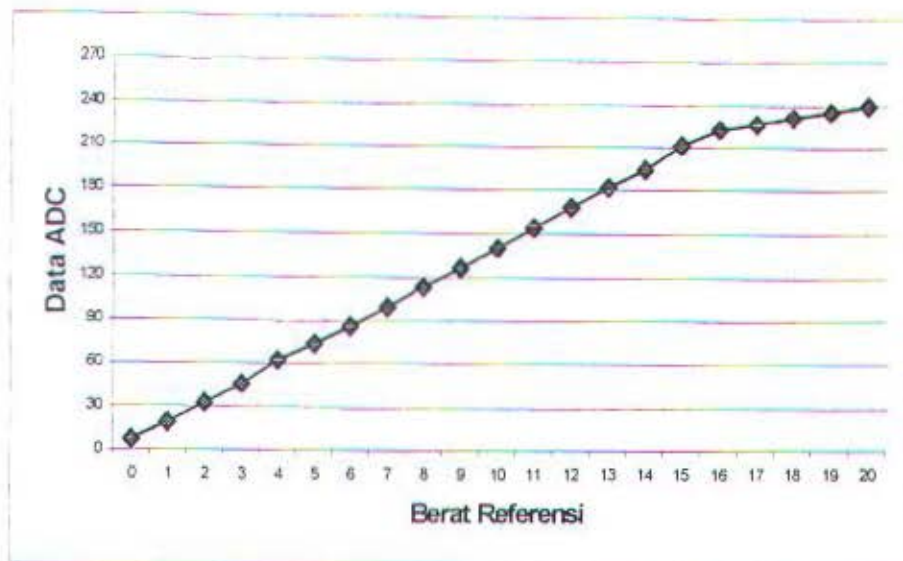
Dengan memberikan tegangan 0-5 Volt (DC) pada input ADC 0804 maka didapatkan hasil untuk penentuan regresi berupa data digital ADC dengan bantuan komputer mulai dari beban nol/tanpa beban sampai dengan beban mencapai 20 kg. (tabel 4.2)



Tabel 4.2. Hasil Pengujian untuk Penentuan Regresi

No	Berat Referensi (Kg)	DATA			
		1	2	3	Rata – rata
1	0	7	8	6	7
2	1	11	20	20	19
3	2	33	33	31	32.33
4	3	43	48	45	45.33
5	4	61	62	62	61.66
6	5	73	73	73	73
7	6	85	85	86	85.33
8	7	98	98	100	98.66
9	8	114	111	114	113
10	9	125	126	128	126.33
11	10	140	140	140	140
12	11	152	155	156	154.33
13	12	168	170	168	168.6
14	13	182	183	183	182.66
15	14	195	194	195	194.66
16	15	211	212	211	211.33
17	16	224	222	223	223
18	17	226	227	227	226.66
19	18	232	231	231	231.33
20	19	236	236	236	236
21	20	240	240	240	240

Dari tabel diatas kemudian dibuat kurva perbandingan data ADC vs berat referensi modul timbangan balita seperti yang ditunjukkan oleh grafik 4.1.



Grafik 4.1 Kurva Data ADC-Berat Referensi Modul Timbangan Balita

Dari grafik 4.1 terlihat bahwa kurva yang dihasilkan cenderung linier tetapi terdiri dari beberapa persamaan pembentuk kurva. Oleh karena itu metode regresi yang diperlukan adalah regresi linier, untuk mendapatkan data yang mendekati data sesungguhnya dengan persamaan dasar sebagai berikut :

$$Y = a + bX$$

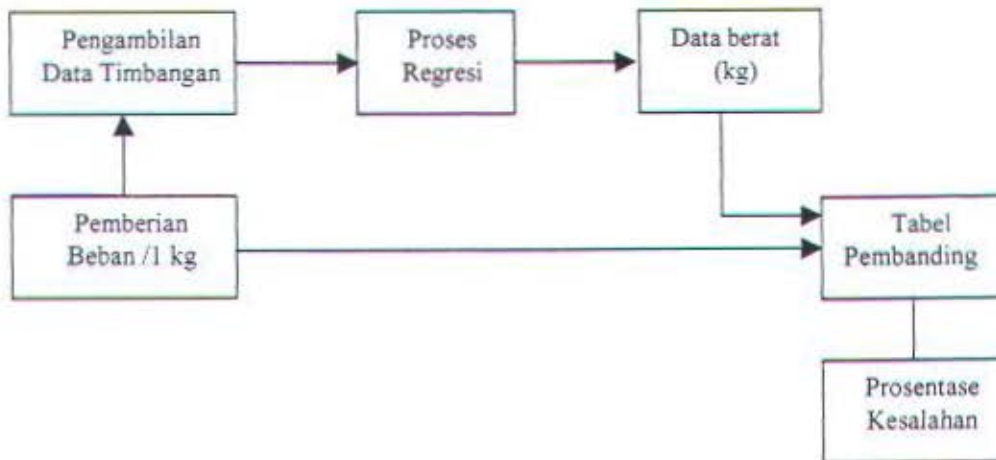
Dimana :

$$a = \frac{\sum x^2 \sum y - \sum x \sum xy}{n \sum x^2 - (\sum x)^2}$$

$$b = \frac{n \sum xy - \sum x \sum y}{n \sum x^2 - (\sum x)^2}$$



Dengan persamaan regresi yang dihasilkan, tahapan berikutnya yaitu pembacaan data berat setelah melalui proses regresi. Gambar 4.4 menunjukkan diagram blok pengujian berat setelah melalui proses regresi.



Gambar 4.4 Diagram blok pengujian berat setelah melalui proses regresi

Dengan menggunakan persamaan $Y = a + bX$ maka dari data ADC yang diperoleh sebagaimana ditunjukkan pada tabel 4.2 dan grafik 4.1 maka persamaan yang terjadi dibagi menjadi 14 bagian yaitu :

1. Untuk $0 \leq y_1 \leq 1$ atau $7 \leq x \leq 19$

Persamaannya adalah $Y = -0.5833 + 0.0833 X$

Tabel 4.3. Perhitungan Regresi untuk $0 \leq y_1 \leq 1$ atau $7 \leq x \leq 19$

n	x	y	x ²	x.y ₁
1	7	0	49	0
2	19	1	361	19
Σ	26	1	410	19



2. Untuk $1 \leq y_1 \leq 3$ atau $19 \leq x \leq 45.33$

Persamaannya adalah $Y = -0.4473 + 0.076 X$

Tabel 4.4. Perhitungan Regresi untuk $1 \leq y_1 \leq 3$ atau $19 \leq x \leq 45.33$

n	x	y	x ²	x.y ₁
1	19	1	361	19
2	32.33	2	1045.229	64.66
3	45.33	3	2054.809	135.99
Σ	96.66	6	3461.038	219.65

3. Untuk $3 \leq y_1 \leq 4$ atau $45.33 \leq x \leq 61.66$

Persamaannya adalah $Y = 0.2241 + 0.0612 X$

Tabel 4.5. Perhitungan Regresi untuk $3 \leq y_1 \leq 4$ atau $45.33 \leq x \leq 61.66$

n	x	y	x ²	x.y ₁
1	45.33	3	2054.809	135.99
2	61.66	4	3801.956	246.64
Σ	106.99	7	5856.765	382.63

4. Untuk $4 \leq y_1 \leq 6$ atau $61.66 \leq x \leq 85.33$

Persamaannya adalah $Y = -1.1924 + 0.0844 X$

Tabel 4.6. Perhitungan Regresi untuk $4 \leq y_1 \leq 6$ atau $61.66 \leq x \leq 85.33$

n	x	y	x ²	x.y ₁
1	61.66	4	3801.956	246.64
2	73	5	5329	365
3	85.33	6	7281.209	511.98
Σ	219.99	15	16412.16	1123.62



5. Untuk $6 \leq y_1 \leq 7$ atau $85.33 \leq x \leq 98.66$

Persamaannya adalah $Y = -0.4014 + 0.075 X$.

Tabel 4.7. Perhitungan Regresi untuk $6 \leq y_1 \leq 7$ atau $85.33 \leq x \leq 98.66$

n	x	y	x ²	x.y ₁
1	85.33	6	7281.209	511.98
2	98.66	7	9733.796	690.62
Σ	183.99	13	17015	1202.6

6. Untuk $7 \leq y_1 \leq 9$ atau $98.66 \leq x \leq 126.33$

Persamaannya adalah $Y = -0.1397 + 0.0722 X$

Tabel 4.8. Perhitungan Regresi untuk $7 \leq y_1 \leq 9$ atau $98.66 \leq x \leq 126.33$

n	x	y	x ²	x.y ₁
1	98.66	7	9733.796	690.62
2	113	8	12769	904
3	126.33	9	15959.27	1136.97
Σ	337.99	24	38462.06	2731.59

7. Untuk $9 \leq y_1 \leq 10$ atau $126.33 \leq x \leq 140$

Persamaannya adalah $Y = -0.2414 + 0.0732 X$

Tabel 4.9. Perhitungan Regresi untuk $9 \leq y_1 \leq 10$ atau $126.33 \leq x \leq 140$

n	x	y	x ²	x.y ₁
1	126.33	9	15959.27	1136.97
2	140	10	19600	1400
Σ	266.33	19	35559.27	2536.97





8. Untuk $10 \leq y_1 \leq 11$ atau $140 \leq x \leq 159.33$

Persamaannya adalah $Y = 0.2303 + 0.0698 X$

Tabel 4.10. Perhitungan Regresi untuk $10 \leq y_1 \leq 11$ atau $140 \leq x \leq 159.33$

n	x	y	x ²	x.y ₁
1	140	10	19600	1400
2	154.33	11	23817.75	1697.63
Σ	294.33	21	43417.75	3097.63

9. Untuk $11 \leq y_1 \leq 13$ atau $159.33 \leq x \leq 182.66$

Persamaannya adalah $Y = 0.1015 + 0.0706 X$

Tabel 4.11. Perhitungan Regresi untuk $11 \leq y_1 \leq 13$ atau $159.33 \leq x \leq 182.66$

n	x	y	x ²	x.y ₁
1	154.33	11	23817.75	1697.63
2	168.66	12	28446.2	2023.92
3	182.66	13	33364.68	2374.58
Σ	505.65	36	85628.62	6096.13

10. Untuk $13 \leq y_1 \leq 14$ atau $182.66 \leq x \leq 194.66$

Persamaannya adalah $Y = -2.2217 + 0.0833 X$

Tabel 4.12. Perhitungan Regresi untuk $13 \leq y_1 \leq 14$ atau $182.66 \leq x \leq 194.66$

n	x	y	x ²	x.y ₁
1	182.66	13	33364.68	2374.58
2	194.66	14	37892.52	2725.24
Σ	377.32	27	71257.19	5099.82



11. Untuk $14 \leq y_1 \leq 16$ atau $194.6 \leq x \leq 223$

Persamaannya adalah $Y = 0.3557 + 0.0698 X$

Tabel 4.13. Perhitungan Regresi untuk $14 \leq y_1 \leq 16$ atau $194.6 \leq x \leq 223$

n	x	y	x ²	x.y ₁
1	194.66	14	37892.52	2725.24
2	211.33	15	44660.37	3169.95
3	223	16	49729	3568
Σ	628.99	45	132281.9	9463.19

12. Untuk $16 \leq y_1 \leq 17$ atau $223 \leq x \leq 226.66$

Persamaannya adalah $Y = -44.929 + 0.2732 X$

Tabel 4.14. Perhitungan Regresi untuk $16 \leq y_1 \leq 17$ atau $223 \leq x \leq 226.66$

n	x	y	x ²	x.y ₁
1	223	16	49729	3568
2	226.66	17	51374.76	3853.22
Σ	449.66	33	101103.8	7421.22

13. Untuk $17 \leq y_1 \leq 19$ atau $226.66 \leq x \leq 236$

Persamaannya adalah $Y = -31.535 + 0.2141 X$

Tabel 4.15. Perhitungan Regresi untuk $17 \leq y_1 \leq 19$ atau $226.66 \leq x \leq 236$

n	x	y	x ²	x.y ₁
1	226.66	17	51374.76	3853.22
2	231.33	18	53513.57	4163.94
3	236	19	55696	4484
Σ	693.99	54	160584.3	12501.16



14. Untuk $19 \leq y_1 \leq 20$ atau $236 \leq x \leq 240$

Persamaannya adalah $Y = -40 + 0.25 X$

Tabel 4.16. Perhitungan Regresi untuk $19 \leq y_1 \leq 20$ atau $236 \leq x \leq 240$

n	x	y	x ²	x.y ₁
1	236	19	55696	4484
2	240	20	57600	4800
Σ	476	39	113296	9284

Tabel 4.17 menunjukkan hasil perbandingan antara data berat setelah melalui proses regresi (y₁) dan data berat referensi (y₂).

Tabel 4.17. Perbandingan Data Setelah Regresi dan Data Referensi

y ₂ (referensi)	0	1	2	3	4	5	6	7	8	9
y ₁ (hasil regresi)	0	1	2.008	3	4	4.972	6 s/d 6.0134	7	8.0243	8.974 s/d 9
Kesalahan (y ₁ - y ₂)	0	0	0.008	0	0	-0.028	0 s/d 0.0143	0	0.0243	0.0126 s/d 0

y ₂ (referensi)	10	11	12	13	14	15	16	17	18	19	20
y ₁ (hasil regresi)	10	11	12	12.996 s/d 13	14	5.116	16	17	18	19	20
Kesalahan (y ₁ - y ₂)	0	0	0	0.004 s/d 0	0	0.11	0	0	0	0	0

4.2. Pengujian Perangkat Lunak (*Software*)

Pada gambar 4.5. menunjukkan tampilan menu utama. Tombol-tombol yang tersedia antara lain : Tombol Ok dan tombol Close, tombol Ok digunakan untuk login dan identifikasi doter yang sedang berjaga saat itu sedangkan tombol close untuk keluar dari program.



Gambar 4.5. Tampilan Menu Utama

Uji coba yang telah dilakukan pada menu utama menunjukkan bahwa fasilitas dan prosedur yang disediakan telah berfungsi sesuai dengan spesifikasi yang telah ditentukan.

Pada gambar 4.6. menunjukkan tampilan menu data ID balita. Menu ini berisi identitas balita sesuai dengan yang tercantum dalam Kartu Menuju Sehat (KMS) balita, dilengkapi dengan fasilitas untuk menambah, menghapus dan mengedit data balita serta menu search berdasarkan nama dan no pendaftaran.

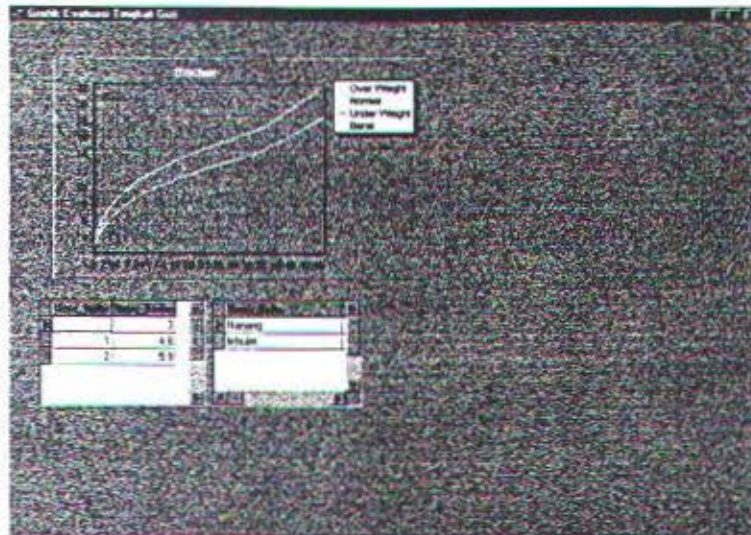


Gambar 4.6. Tampilan Menu Data ID Balita

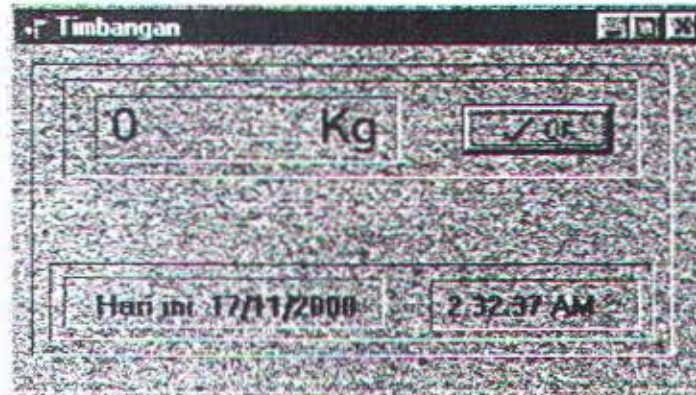
Menu yang lain adalah menu ruang periksa balita (gambar 4.7), menu grafik (gambar 4.8), menu timbangan (gambar 4.9) dan menu laporan penimbangan bulanan (gambar 4.10).



Gambar 4.6. Tampilan Menu Ruang Periksa



Gambar 4.8. Tampilan Menu Grafik Evaluasi Tingkat Gizi



Gambar 4.9. Tampilan Menu Timbangan



Gambar 4.10. Tampilan Menu Laporan Penimbangan Bulanan



BAB V**PENUTUP**

Kesimpulan yang diperoleh selama perancangan dan pengembangan sistem berbasis PC untuk pengukuran berat badan balita ini akan diuraikan pada bab ini.

Beberapa saran yang diberikan diharapkan dapat bermanfaat terutama dalam memperbaiki unjuk kerja sistem yang telah direalisasikan.

5.1. Kesimpulan

Kesimpulan yang diperoleh dari tugas akhir ini antara lain :

1. Melakukan penimbangan secara elektronik dengan spesifikasi teknis sebagai berikut :
 - Jangkauan pengukuran : 0 – 20 Kg.
 - *Amplifier* : *Non Inverting Amplifier*, $A_{CL} = 1X - 3X$
 - *Filter* : LPF 40 dB/dec, $f_c = 5$ Hz.
2. Modul timbangan balita yang dirancang dan direalisasikan ini telah berfungsi dengan baik sebagaimana parameter sebagai berikut :
 - Ketelitian : 78.125 gram.
 - Kesalahan pengukuran pada setiap range regresi berubah, yaitu antara 0 s/d 28 gram.



3. Potensiometer geser dapat digunakan sebagai sensor berat dengan melakukan modifikasi mekanik timbangan manual dan menggunakan metode regresi dalam pengolahan data.
4. Beban maksimal timbangan tergantung dari bentuk mekanik timbangan manual. Untuk timbangan badan balita ini beban maksimalnya adalah 20 Kg, sedang untuk timbangan badan orang dewasa adalah sampai dengan 120 Kg. Dan faktor yang paling berpengaruh adalah pegasnya.
5. Penggunaan sistem berbasis PC untuk pengukuran berat badan balita ini dapat membantu meningkatkan pelayanan kesehatan bagi bayi dan balita dalam hal efisiensi pengumpulan, penyimpanan, pengambilan kembali data dan presentasi data.
6. Sebagai pengganti penggunaan Kartu Menuju Sehat (KMS) Balita, maka lebih efisien dalam pelaporan, pengolahan data karena dilakukan secara otomatis dan keakuratan data dalam pelaporan lebih terjamin karena semua proses perhitungan dilakukan dengan bantuan komputer.

5.2 Saran

Beberapa saran yang dapat disampaikan berkenaan dengan sistem berbasis PC yang dirancang dan direalisasikan ini antara lain :

1. Timbangan ini kurang efisien apabila hanya berfungsi sebagai timbangan saja (mengingat display-nya adalah komputer) sehingga perlu penambahan fungsi untuk lebih memaksimalkan fungsi dari komputer itu sendiri.

DAFTAR PUSTAKA

1. Jerald G.Graeme, 1973, Applications of Operational Amplifiers, Burr-Brown Research Corporation.
2. Santoso, Retno Dwi, Ir, Msi, Analisis Regresi, Andi Offset Yogyakarta, 1996.
3. Sugiono, Ir, Aplikasi Analisis Regresi, Andi Offset Yogyakarta, 1990.
4. Pranata, Anthony, Pemrograman Borlan Delphi, Edisi I Cetakan I, Andi Yogyakarta, 1997.
5. Okianto, Dani, Panduan Belajar Delphi 3.0, Elex Media Komputindo, Jakarta 1998.
6. Datasheet Book, National Semiconduktor, 1997.
7. Microprocesor & Interface, Poltek Elektronika ITS & Garuda Indonesia, Surabaya, 1996.
8. Sardjono, Tri Arief, ST, MT, Thesis, Institut Teknologi Bandung 1996.

LISTING PROGRAM

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, DBCtrls, StdCtrls, Buttons, Mask;

type
  TForm1 = class(TForm)
    Label1: TLabel;
    Timer1: TTimer;
    Label2: TLabel;
    DBEdit1: TDBEdit;
    Label3: TLabel;
    BitBtn1: TBitBtn;
    Bevel1: TBevel;
    BitBtn2: TBitBtn;
    Bevel2: TBevel;
    Bevel3: TBevel;
    Image1: TImage;
    Bevel4: TBevel;
    procedure Timer1Timer(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

uses Unit2;

{$R *.DFM}

procedure TForm1.Timer1Timer(Sender: TObject);
var
  Present: TDateTime;
  Year, Month, Day : Word;
begin
  Present := Now;
  DecodeDate(Present, Year, Month, Day);
  Label2.Caption := " + IntToStr(Day) + ' - '
    + IntToStr(Month) + ' - ' + IntToStr(Year);
end;
```

LAMPIRAN 1

```
procedure TForm1.BitBtn1Click(Sender: TObject);
begin
    Form2.ShowModal;
end;

end.
```

LISTING PROGRAM

```
unit Unit2;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, Mask, DBCtrls, StdCtrls, Db, DBTables, Buttons;

type
  TForm2 = class(TForm)
    Edit1: TEdit;
    Button1: TButton;
    Edit2: TEdit;
    Button2: TButton;
    DBEdit1: TDBEdit;
    DBEdit2: TDBEdit;
    DBEdit3: TDBEdit;
    DBEdit4: TDBEdit;
    DBEdit5: TDBEdit;
    DBEdit6: TDBEdit;
    DBEdit7: TDBEdit;
    DBEdit8: TDBEdit;
    DBEdit9: TDBEdit;
    DBEdit10: TDBEdit;
    DBEdit11: TDBEdit;
    Timer1: TTimer;
    DataSource1: TDataSource;
    Table1: TTable;
    RadioGroup1: TRadioGroup;
    RadioButton1: TRadioButton;
    RadioButton2: TRadioButton;
    DBEdit12: TDBEdit;
    Timer2: TTimer;
    DBNavigator1: TDBNavigator;
    DBNavigator2: TDBNavigator;
    Bevel1: TBevel;
    BitBtn1: TBitBtn;
    Button3: TButton;
    Bevel2: TBevel;
    Bevel3: TBevel;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    Label8: TLabel;
    Label9: TLabel;
```

LAMPIRAN 1

```
Label10: TLabel;
Bevel4: TBevel;
procedure Timer1Timer(Sender: TObject);
procedure RadioButton1Click(Sender: TObject);
procedure RadioButton2Click(Sender: TObject);
procedure Timer2Timer(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  Form2: TForm2;

implementation

uses Unit3, Unit1;

{$R *.DFM}

procedure TForm2.Timer1Timer(Sender: TObject);
begin
  DBEdit11.Text := '' + datetoStr(Date);
end;

procedure TForm2.RadioButton1Click(Sender: TObject);
begin
  DBEdit12.Text := 'Laki-Laki'
end;

procedure TForm2.RadioButton2Click(Sender: TObject);
begin
  DBEdit12.Text := 'Perempuan'
end;

procedure TForm2.Timer2Timer(Sender: TObject);
begin
  DBEdit12.Hide;
end;

procedure TForm2.BitBtn1Click(Sender: TObject);
begin
  close;
end;

procedure TForm2.Button3Click(Sender: TObject);
begin
  Form3.Showmodal;
end;
```



LAMPIRAN 1

```
procedure TForm2.Button1Click(Sender: TObject);
begin
  if (not Table1.FindKey ([StrToInt
    (Edit1.text)])) then
    Application.MessageBox('Data Tidak Ditemukan','Informasi',MB_Ok or
      MB_IconInformation);
end;

procedure TForm2.Button2Click(Sender: TObject);
begin
  with Table1 do
  begin
    SetKey;
    FieldByName('Nama_Balita').AsString := '';
    GotoNearest;
  end;
end;

end.
```

LISTING PROGRAM

```
unit Unit3;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, DBCtrls, Grids, DBGrids, Db, DBTables, StdCtrls, Mask, Buttons;

type
  TForm3 = class(TForm)
    DBEdit1: TDBEdit;
    DBEdit2: TDBEdit;
    Label1: TLabel;
    Label2: TLabel;
    Query1: TQuery;
    DataSource1: TDataSource;
    Bevel1: TBevel;
    DBGrid1: TDBGrid;
    DBGrid2: TDBGrid;
    DBNavigator1: TDBNavigator;
    Table1: TTable;
    DataSource4: TDataSource;
    Table2: TTable;
    DataSource2: TDataSource;
    Label3: TLabel;
    Bevel2: TBevel;
    DBMemo1: TDBMemo;
    Label4: TLabel;
    Button1: TButton;
    Button2: TButton;
    Button3: TButton;
    Button4: TButton;
    BitBtn1: TBitBtn;
    procedure Button1Click(Sender: TObject);
    procedure BitBtn1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form3: TForm3;

implementation

uses Unit4, Unit5;
```

LAMPIRAN 1

{SR *.DFM}

```
procedure TForm3.Button1Click(Sender: TObject);
begin
    Form4.ShowModal;
end;
```

```
procedure TForm3.BitBtn1Click(Sender: TObject);
begin
    Close;
end;
```

```
procedure TForm3.Button2Click(Sender: TObject);
begin
    form5.ShowModal;
end;
```

```
end.
```

LISTING PROGRAM

```
unit Unit4;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  TeEngine, Series, Grids, DBGrids, Db, DBTables, ExtCtrls, TeeProcs,
  Chart, DBChart;

type
  TForm4 = class(TForm)
    DBChart1: TDBChart;
    Query1: TQuery;
    DataSource1: TDataSource;
    DBGrid1: TDBGrid;
    Series1: TLineSeries;
    Series2: TLineSeries;
    Series3: TLineSeries;
    Query2: TQuery;
    DataSource2: TDataSource;
    Query3: TQuery;
    DataSource3: TDataSource;
    Query4: TQuery;
    DataSource4: TDataSource;
    Series4: TLineSeries;
    Query5: TQuery;
    DataSource5: TDataSource;
    DBGrid2: TDBGrid;
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form4: TForm4;

implementation

{$R *.DFM}

end.
```


LISTING PROGRAM

```
unit Unit5;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, ExtCtrls;

type
  TForm5 = class(TForm)
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Edit4: TEdit;
    Edit5: TEdit;
    Edit6: TEdit;
    Edit7: TEdit;
    Edit8: TEdit;
    Edit9: TEdit;
    Edit10: TEdit;
    Edit11: TEdit;
    Edit12: TEdit;
    Edit13: TEdit;
    Edit14: TEdit;
    Edit15: TEdit;
    Edit16: TEdit;
    Edit17: TEdit;
    Edit18: TEdit;
    Edit19: TEdit;
    Edit20: TEdit;
    Edit27: TEdit;
    Edit29: TEdit;
    Edit30: TEdit;
    Edit31: TEdit;
    Edit32: TEdit;
    Edit33: TEdit;
    Edit34: TEdit;
    Edit35: TEdit;
    Edit36: TEdit;
    Edit37: TEdit;
    Edit38: TEdit;
    Edit39: TEdit;
    Edit40: TEdit;
    Edit41: TEdit;
    Edit42: TEdit;
    Edit43: TEdit;
    Edit44: TEdit;
    Edit45: TEdit;
    Edit46: TEdit;
```

LAMPIRAN 1

```
Edit47: TEdit;
Edit48: TEdit;
Edit49: TEdit;
Edit50: TEdit;
Edit51: TEdit;
Label1: TLabel;
Label2: TLabel;
Label3: TLabel;
Label4: TLabel;
Label5: TLabel;
Label6: TLabel;
Label7: TLabel;
Label8: TLabel;
Label9: TLabel;
Label10: TLabel;
Label11: TLabel;
Edit21: TEdit;
Edit22: TEdit;
Edit23: TEdit;
Edit24: TEdit;
Edit25: TEdit;
Edit26: TEdit;
Edit28: TEdit;
Label12: TLabel;
Label13: TLabel;
Label14: TLabel;
Label15: TLabel;
Label16: TLabel;
Label17: TLabel;
Label18: TLabel;
Label19: TLabel;
Label20: TLabel;
Label21: TLabel;
Label22: TLabel;
Label23: TLabel;
Label24: TLabel;
Label25: TLabel;
Label26: TLabel;
Label27: TLabel;
Label28: TLabel;
Label29: TLabel;
Label30: TLabel;
Label31: TLabel;
Label32: TLabel;
Bevel1: TBevel;
Bevel2: TBevel;
Bevel3: TBevel;
Bevel4: TBevel;
private
  { Private declarations }
public
  { Public declarations }
end;
```

LAMPIRAN 1

```
var  
  Form5: TForm5;  
  
implementation  
  
{SR *.DFM}  
  
end.
```

LISTING PROGRAM

```
unit Unit6;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, StdCtrls, Buttons;

type
  TForm6 = class(TForm)
    Label1: TLabel;
    Label2: TLabel;
    BitBtn1: TBitBtn;
    Timer1: TTimer;
    Timer2: TTimer;
    Timer3: TTimer;
    Timer4: TTimer;
    Label8: TLabel;
    Label3: TLabel;
    Bevel1: TBevel;
    Bevel2: TBevel;
    Bevel3: TBevel;
    Bevel4: TBevel;
    Bevel5: TBevel;
    Bevel6: TBevel;
    procedure Timer1Timer(Sender: TObject);
    procedure Timer2Timer(Sender: TObject);
    procedure Timer4Timer(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form6: TForm6;
  dat : byte;
  m, suhu, c : real;
  counter : integer;
  data : string;
  setup : integer;

implementation

{$R *.DFM}

procedure Outport(NoPort : word; Data : byte); assembler;
asm
  mov al, Data
```



LAMPIRAN 1

```
    mov dx, NoPort
    out dx, al
end;

function InPort(NoPort : word) : byte; assembler;
asm
    mov dx, NoPort
    in al, dx
end;

procedure TForm6.Timer1Timer(Sender: TObject);
begin
    dat:=InPort($300);
    if (dat=0) and (dat<7) then
    begin
        m:=0;
        c:=0;
        suhu:=(m*dat)+c;
        Label1.Caption:=floatToStr(suhu);
    end;
    if (dat=7) and (dat<=19) then
    begin
        m:=-0.583;
        c:=0.0833;
        suhu:=(m*dat)+c;
        Label1.Caption:=floatToStr(suhu);
    end;
    if (dat>=19) and (dat<=46) then
    begin
        m:=-0.447;
        c:=0.076;
        suhu:=(m*dat)+c;
        Label1.Caption:=floatToStr(suhu);
    end;
    if (dat>=46) and (dat<=62) then
    begin
        m:=0.2241;
        c:=0.0612;
        suhu:=(m*dat)+c;
        Label1.Caption:=floatToStr(suhu);
    end;
    if (dat>=62) and (dat<=86) then
    begin
        m:=-1.192;
        c:=0.0844;
        suhu:=(m*dat)+c;
        Label1.Caption:=floatToStr(suhu);
    end;
    if (dat>=86) and (dat<=99) then
    begin
        m:=-0.401;
        c:=0.075;
        suhu:=(m*dat)+c;
        Label1.Caption:=floatToStr(suhu);
    end;
end;
```

```
if (dat>=99) and (dat<=127) then
begin
m:=-0.14;
c:=0.0722;
suhu:=(m*dat)+c;
Label1.Caption:=floatToStr(suhu);
end;
if (dat>=127) and (dat<=140) then
begin
m:=-0.241;
c:=0.0732;
suhu:=(m*dat)+c;
Label1.Caption:=floatToStr(suhu);
end;
if (dat>=140) and (dat<=155) then
begin
m:=0.2303;
c:=0.0698;
suhu:=(m*dat)+c;
Label1.Caption:=floatToStr(suhu);
end;
if (dat>=155) and (dat<=183) then
begin
m:=0.1015;
c:=0.0706;
suhu:=(m*dat)+c;
Label1.Caption:=floatToStr(suhu);
end;
if (dat>=183) and (dat<=195) then
begin
m:=-2.222;
c:=0.0833;
suhu:=(m*dat)+c;
Label1.Caption:=floatToStr(suhu);
end;
if (dat>=195) and (dat<=223) then
begin
m:=0.3557;
c:=0.0698;
suhu:=(m*dat)+c;
Label1.Caption:=floatToStr(suhu);
end;
if (dat>=223) and (dat<=227) then
begin
m:=-44.93;
c:=0.2732;
suhu:=(m*dat)+c;
Label1.Caption:=floatToStr(suhu);
end;
if (dat>=227) and (dat<=236) then
begin
m:=-31.54;
c:=0.2141;
suhu:=(m*dat)+c;
```

LAMPIRAN 1

```
Label1.Caption:=floatToStr(suhu);
end;
if (dat>=236) and (dat<=240) then
begin
m:=-40;
c:=0.25;
suhu:=(m*dat)+c;
Label1.Caption:=floatToStr(suhu);
end;
end;

procedure TForm6.Timer2Timer(Sender: TObject);
begin
dat:=InPort($300);
m:=0.078;
c:=0;
suhu:=(m*dat)+c;
// m2:=1;
// c2:=0;
// posx:=(m2*dat)+c2;
if suhu>=setup then
begin
beep;
end;
Label1.Caption:=floatToStr(suhu);
end;

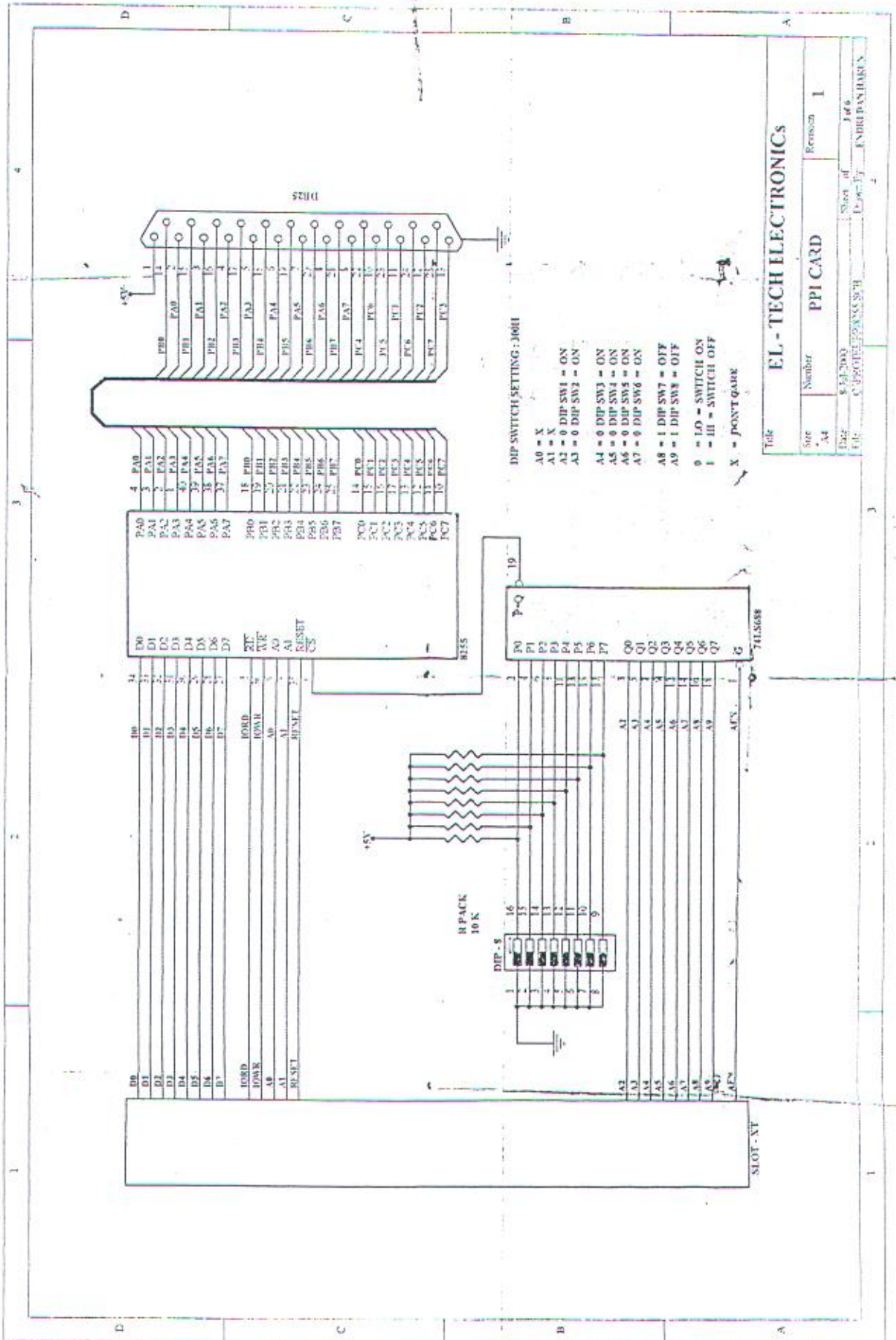
procedure TForm6.Timer4Timer(Sender: TObject);
var
Present: TDateTime;
Year, Month, Day, Hour, Min, Sec, MSec: Word;
begin
Present:=Now;
DecodeDate(Present, Year, Month, Day);
Label2.Caption := 'Hari ini ' + IntToStr(Day) + '/'
+ IntToStr(Month) + '/' + IntToStr(Year);
DecodeTime(Present, Hour, Min, Sec, MSec);
Label8.Caption := " + IntToStr(Hour) + ':'
+ IntToStr(min) + ':' + IntToStr(Sec) + ' AM' ;

//begin
//Label2.Caption := 'Hari ini ' + DateToStr(Date);
end;

end.
```

HASIL PENGUJIAN

No.	Berat Referensi Kg	DATA										Rata-rata
		1	2	3	4	5	6	7	8	9	10	
1	0	7	5	8	10	6	8	9	7	8	6	7,4
2	1	15	14	10	15	15	17	14	17	20	20	15,7
3	2	22	22	24	28	31	30	29	33	33	31	28,3
4	3	29	28	28	33	46	43	47	43	48	45	39
5	4	34	35	34	47	61	61	63	61	62	62	52
6	5	42	42	40	53	69	70	72	73	73	73	60,7
7	6	51	48	47	67	86	84	84	85	85	86	72,3
8	7	57	55	52	78	99	100	101	98	98	100	83,8
9	8	62	61	63	93	111	114	112	114	111	114	95,5
10	9	72	72	68	102	126	128	125	125	126	128	107,2
11	10	75	76	77	114	145	141	141	140	140	140	118,9
12	11	86	87	85	121	150	152	153	152	155	156	129,7
13	12	92	94	95	123	170	167	169	168	170	168	141,6
14	13	99	98	100	129	180	182	180	182	183	183	151,6
15	14	103	103	110	149	192	195	195	195	194	195	163,1
16	15	114	113	115	162	215	210	211	211	212	211	177,4
17	16	120	119	120	197	221	225	220	224	223	223	189,2
18	17	127	128	125	200	228	227	226	226	227	227	194,1
19	18	220	219	221	215	230	231	231	232	231	231	226,1
20	19	229	224	230	225	235	234	234	236	236	236	231,9
21	20	237	238	235	233	240	239	239	240	240	240	238,1

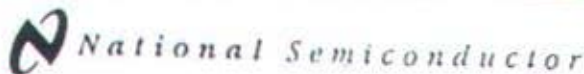


DIP SWITCH SETTING : 1011

- A0 = X
- A1 = X
- A2 = 0 DIP SW1 = ON
- A3 = 0 DIP SW2 = ON
- A4 = 0 DIP SW3 = ON
- A5 = 0 DIP SW4 = ON
- A6 = 0 DIP SW5 = ON
- A7 = 0 DIP SW6 = ON
- A8 = 1 DIP SW7 = OFF
- A9 = 1 DIP SW8 = OFF
- 0 = IO - SWITCH ON
- 1 = III - SWITCH OFF
- X = DON'T CARE

EL - TECHELECTRONICS

Title		Revision	
Part Number	PPI CARD		1
Date	8 Feb 2003	Sheet of	1 of 6
File	C:\PROJECTS\2003\AS507B	Drawn By	ENDRIYANHARY



ADC0801/ADC0802/ADC0803/ADC0804/ADC0805 8-Bit μ P Compatible A/D Converters

General Description

The ADC0801, ADC0802, ADC0803, ADC0804 and ADC0805 are CMOS 8-bit successive approximation A/D converters that use a differential potentiometric ladder—similar to the 256R products. These converters are designed to allow operation with the NSC800 and INS8000A derivative control bus with TRI-STATE* output latches directly driving the data bus. These A/Ds appear like memory locations or I/O ports to the microprocessor and no interfacing logic is needed.

Differential analog voltage inputs allow increasing the common-mode rejection and offsetting the analog zero input voltage value. In addition, the voltage reference input can be adjusted to allow encoding any smaller analog voltage span to the full 8 bits of resolution.

Features

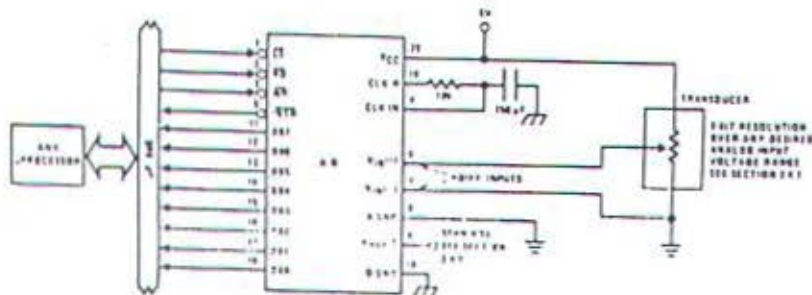
- Compatible with 8080 μ P derivatives—no interfacing logic needed • access time • 135 ns
- Easy interface to all microprocessors, or operates "stand alone"

- Differential analog voltage inputs
- Logic inputs and outputs meet both MOS and TTL voltage level specifications
- Works with 2.5V (LM336) voltage reference
- On-chip clock generator
- 0V to 5V analog input voltage range with single 5V supply
- No zero adjust required
- 0.3" standard width 20-pin DIP package
- 20-pin molded chip carrier or small outline package
- Operates ratiometrically or with 5 V_{DC}, 2.5 V_{DC}, or analog span adjusted voltage reference

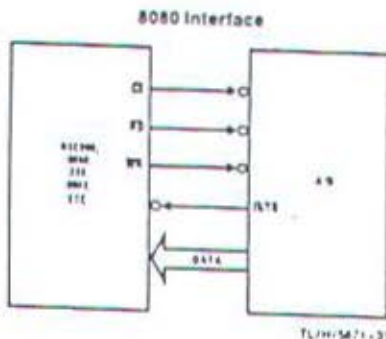
Key Specifications

- Resolution 8 bits
- Total error $\pm 1/4$ LSB, $\pm 1/2$ LSB and ± 1 LSB
- Conversion time 100 μ s

Typical Applications



TL911471-1



TL911471-2

Error Specification (Includes Full-Scale, Zero Error, and Non-Linearity)			
Part Number	Full-Scale Adjusted	V _{REF} /2 = 2.500 V _{DC} (No Adjustments)	V _{REF} /2 = No Connection (No Adjustments)
ADC0801	$\pm 1/4$ LSB		
ADC0802		$\pm 1/2$ LSB	
ADC0803	$\pm 1/2$ LSB		
ADC0804		± 1 LSB	
ADC0805			± 1 LSB

Absolute Maximum Ratings (Notes 1 & 2)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Supply Voltage (V_{CC}) (Note 3)	6.5V
Voltage	
Logic Control Inputs	-0.3V to +18V
All Other Input and Outputs	-0.3V to ($V_{CC} + 0.3V$)
Lead Temp. (Soldering, 10 seconds)	
Dual-In-Line Package (plastic)	250°C
Dual-In-Line Package (ceramic)	300°C
Surface Mount Package	
Vapor Phase (60 seconds)	215°C
Infrared (15 seconds)	220°C

Storage Temperature Range	-65°C to +150°C
Package Dissipation at $T_A = 25^\circ\text{C}$	875 mW
ESD Susceptibility (Note 10)	800V

Operating Ratings (Notes 1 & 2)

Temperature Range	$T_{MIN} \leq T_A \leq T_{MAX}$
ADC0801/02LJ, ADC0802LJ/853	$-55^\circ\text{C} \leq T_A \leq +125^\circ\text{C}$
ADC0801/02/03/04LQJ	$-40^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$
ADC0801/02/03/05LCN	$-40^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$
ADC0804LCN	$0^\circ\text{C} \leq T_A \leq +70^\circ\text{C}$
ADC0802/03/04LCV	$0^\circ\text{C} \leq T_A \leq +70^\circ\text{C}$
ADC0802/03/04LCWM	$0^\circ\text{C} \leq T_A \leq +70^\circ\text{C}$
Range of V_{CC}	4.5 V_{CC} to 6.3 V_{CC}

Electrical Characteristics

The following specifications apply for $V_{CC} = 5 V_{CC}$, $T_{MIN} \leq T_A \leq T_{MAX}$ and $f_{CLK} = 640 \text{ kHz}$ unless otherwise specified.

Parameter	Conditions	Min	Typ	Max	Units
ADC0801: Total Adjusted Error (Note 8)	With Full-Scale Adj. (See Section 2.5.2)			$\pm 1/4$	LSB
ADC0802: Total Unadjusted Error (Note 8)	$V_{REF}/2 = 2.500 V_{CC}$			$\pm 1/2$	LSB
ADC0803: Total Adjusted Error (Note 8)	With Full-Scale Adj. (See Section 2.5.2)			$\pm 1/2$	LSB
ADC0804: Total Unadjusted Error (Note 8)	$V_{REF}/2 = 2.500 V_{CC}$			± 1	LSB
ADC0805: Total Unadjusted Error (Note 8)	$V_{REF}/2$ -No Connection			± 1	LSB
$V_{REF}/2$ Input Resistance (Pin 9)	ADC0801/02/03/05 ATX0804 (Note 9)	2.5 0.75	8.0 1.1		k Ω k Ω
Analog Input Voltage Range	(Note 4) $V(+)$ or $V(-)$	$G_{IN} - 0.05$		$V_{CC} + 0.05$	V_{IC}
DC Common-Mode Error	Over Analog Input Voltage Range		$\pm 1/4$	$\pm 1/4$	LSB
Power Supply Sensitivity	$V_{CC} = 5 V_{CC} \pm 10\%$ Over Allowed $V_{IN}(+)$ and $V_{IN}(-)$ Voltage Range (Note 4)		$\pm 1/4$	$\pm 1/4$	LSB

AC Electrical Characteristics

The following specifications apply for $V_{CC} = 5 V_{CC}$ and $T_A = 25^\circ\text{C}$ unless otherwise specified.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
T_C	Conversion Time	$f_{CLK} = 640 \text{ kHz}$ (Note 6)	103		114	μs
T_C	Conversion Time	(Note 5, 6)	66		73	$1/f_{CLK}$
f_{CLK}	Clock Frequency	$V_{CC} = 5V$, (Note 5)	100	640	1460	kHz
	Clock Duty Cycle	(Note 5)	40		60	%
CR	Conversion Rate in Free-Running Mode	INTN tied to WR with $\overline{CS} = 0 V_{CC}$, $f_{CLK} = 640 \text{ kHz}$	8770		9708	conv/s
$t_{W(WR)}$	Width of WR Input (Start Pulse Width)	$\overline{CS} = 0 V_{CC}$ (Note 7)	100			ns
t_{ACC}	Access Time (Delay from Falling Edge of RD to Output Data Valid)	$C_L = 100 \text{ pF}$		135	200	ns
$t_{HI, t_{OH}}$	TRI-STATE Control (Delay from Rising Edge of RD to Hi-Z State)	$C_L = 10 \text{ pF}$, $R_L = 10k$ (See TRI-STATE Test Circuits)		125	200	ns
$t_{W, t_{HI}}$	Delay from Falling Edge of WR or RD to Reset of INTN			300	450	ns
C_{IN}	Input Capacitance of Logic Control Inputs			5	7.5	pF
C_{OUT}	TRI-STATE Output Capacitance (Data Buffers)			5	7.5	pF

CONTROL INPUTS (Note: CLK IN (Pin 4) is the input of a Schmitt trigger circuit and is therefore specified separately)

$V_{IN}(1)$	Logical "1" Input Voltage (Except Pin 4 CLK IN)	$V_{CC} = 5.25 V_{CC}$	2.0		15	V_{CC}
-------------	---	------------------------	-----	--	----	----------

AC Electrical Characteristics (Continued)

The following specifications apply for $V_{CC} = 5V_{DD}$ and $T_{MIN} \leq T_A \leq T_{MAX}$, unless otherwise specified.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
CONTROL INPUTS (Note: CLK IN (Pin 4) is the input of a Schmitt trigger circuit and is therefore specified separately)						
$V_{IN}(0)$	Logical "0" Input Voltage (Except Pin 4 CLK IN)	$V_{CC} = 4.75 V_{DD}$			0.8	V_{DD}
$I_{IN}(1)$	Logical "1" Input Current (All Inputs)	$V_{IN} = 5 V_{DD}$		0.005	1	μA_{DC}
$I_{IN}(0)$	Logical "0" Input Current (All Inputs)	$V_{IN} = 0 V_{DD}$	-1	-0.005		μA_{DC}
CLOCK IN AND CLOCK R						
V_{T+}	CLK IN (Pin 4) Positive Going Threshold Voltage		2.7	3.1	3.5	V_{DD}
V_{T-}	CLK IN (Pin 4) Negative Going Threshold Voltage		1.5	1.8	2.1	V_{DD}
V_H	CLK IN (Pin 4) Hysteresis ($V_{T+} - V_{T-}$)		0.6	1.3	2.0	V_{DD}
$V_{OUT}(0)$	Logical "0" CLK R Output Voltage	$I_O = 360 \mu A$ $V_{CC} = 4.75 V_{DD}$			0.4	V_{DD}
$V_{OUT}(1)$	Logical "1" CLK R Output Voltage	$I_O = -360 \mu A$ $V_{CC} = 4.75 V_{DD}$	2.4			V_{DD}
DATA OUTPUTS AND INTR						
$V_{OUT}(0)$	Logical "0" Output Voltage Data Outputs INTR Output	$I_{OUT} = 1.8 mA, V_{CC} = 4.75 V_{DD}$ $I_{OUT} = 1.0 mA, V_{CC} = 4.75 V_{DD}$			0.4 0.4	V_{DD} V_{DD}
$V_{OUT}(1)$	Logical "1" Output Voltage	$I_O = -360 \mu A, V_{CC} = 4.75 V_{DD}$	2.4			V_{DD}
$V_{OUT}(1)$	Logical "1" Output Voltage	$I_O = -10 \mu A, V_{CC} = 4.75 V_{DD}$	4.5			V_{DD}
I_{OUT}	TRI-STATE Disabled Output Leakage (All Data Buffers)	$V_{OUT} = 0 V_{DD}$ $V_{OUT} = 5 V_{DD}$	-3		3	μA_{DC} μA_{DC}
I_{SOURCE}		V_{OUT} Short to Gnd, $T_A = 25^\circ C$	4.5	6		mA_{DC}
I_{SINK}		V_{OUT} Short to V_{CC} , $T_A = 25^\circ C$	9.0	16		mA_{DC}
POWER SUPPLY						
I_{CC}	Supply Current (Includes Ladder Current)	$f_{CLK} = 840 kHz$, $V_{REF/2} = 1.1V$, $T_A = 25^\circ C$ and $I_{CS} = 5V$				
	ADC0801/02/03/04/CJ/05 ADC0804LCN/LCV/LCWM			1.1 1.9	1.8 2.5	mA mA

Note 1: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its specified operating conditions.

Note 2: All voltages are measured with respect to Gnd, unless otherwise specified. The separate A Gnd point should always be wired to the D Gnd.

Note 3: A zener diode exists internally from V_{CC} to Gnd and has a typical breakdown voltage of 7 V_{DD} .

Note 4: For $V_{IN}(+) \rightarrow V_{OUT}(+)$ the digital output code will be 0000 0000. Two on-chip diodes are tied to each analog input (see block diagram) which will forward conduct for analog input voltages one diode drop below ground or one diode drop greater than the V_{DD} supply. Be careful during testing at low V_{DD} levels (4.5V), as high level analog inputs (5V) can cause this input diode to conduct—especially at elevated temperatures, and cause errors for analog inputs near $1/4$ -scale. The spec allows 50 mV forward bias of either diode. This means that as long as the analog V_{IN} does not exceed the supply voltage by more than 50 mV, the output code will be correct. To achieve an absolute 0 V_{DD} to 5 V_{DD} input voltage range will therefore require a minimum supply voltage of 4.950 V_{DD} over temperature variations, initial tolerance and loading.

Note 5: Accuracy is guaranteed at $f_{CLK} = 840 kHz$. At higher clock frequencies accuracy can degrade. For lower clock frequencies, the duty cycle limits can be extended so long as the minimum clock high time interval or minimum clock low time interval is no less than 275 ns.

Note 6: With an asynchronous start pulse, up to 8 clock periods may be required before the internal clock phases are proper to start the conversion process. The start request is internally latched, see Figure 2 and section 2.0.

Note 7: The CS input is assumed to bracket the WR strobe input and therefore timing is dependent on the WR pulse width. An arbitrarily wide pulse width will hold the converter in a reset mode and the start of conversion is initiated by the low to high transition of the WR pulse (see timing diagrams).

Note 8: None of these A/Ds requires a zero adjust (see section 2.5.1). To obtain zero code at other analog input voltages see section 2.5 and Figure 5.

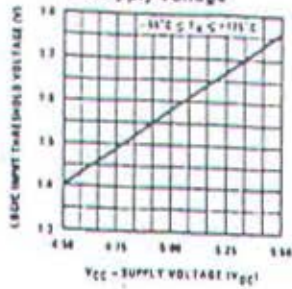
Note 9: The $V_{REF/2}$ pin is the center point of a two-resistor divider connected from V_{CC} to ground. In all versions of the ADC0801, ADC0802, ADC0803, and ADC0805, and in the ADC0804LCJ, each resistor is typically 16 k Ω . In all versions of the ADC0804 except the ADC0804LCJ, each resistor is typically 2.2 k Ω .

Note 10: Human body model, 100 pF discharged through a 1.5 k Ω resistor.

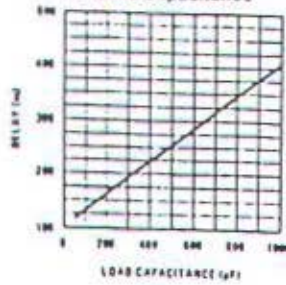


Typical Performance Characteristics

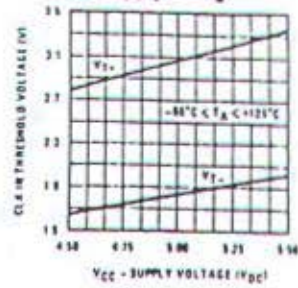
Logic Input Threshold Voltage vs. Supply Voltage



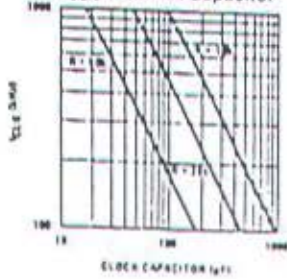
Delay From Falling Edge of \overline{RD} to Output Data Valid vs. Load Capacitance



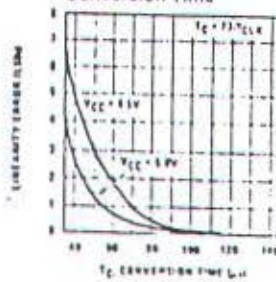
CLK IN Schmitt Trip Levels vs. Supply Voltage



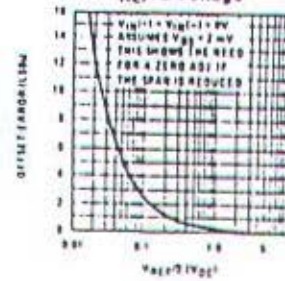
f_{CLK} vs. Clock Capacitor



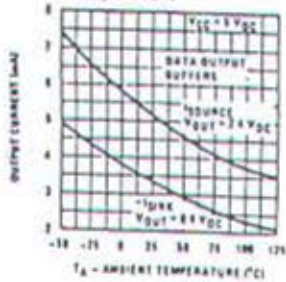
Full-Scale Error vs Conversion Time



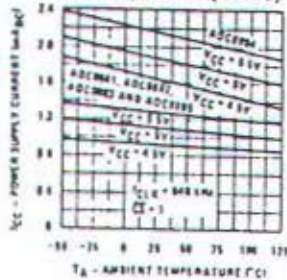
Effect of Unadjusted Offset Error vs. VREF/2 Voltage



Output Current vs Temperature



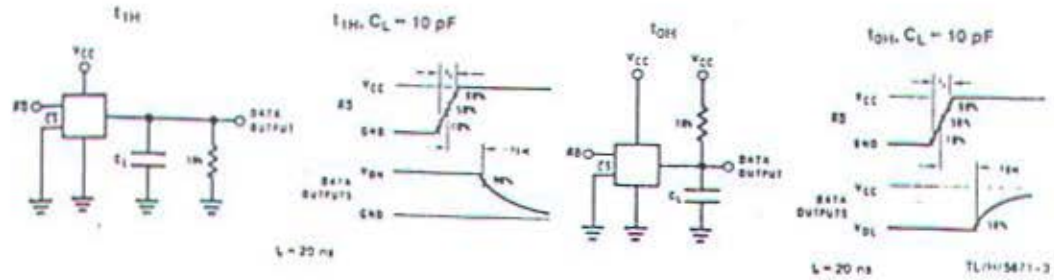
Power Supply Current vs Temperature (Note 9)



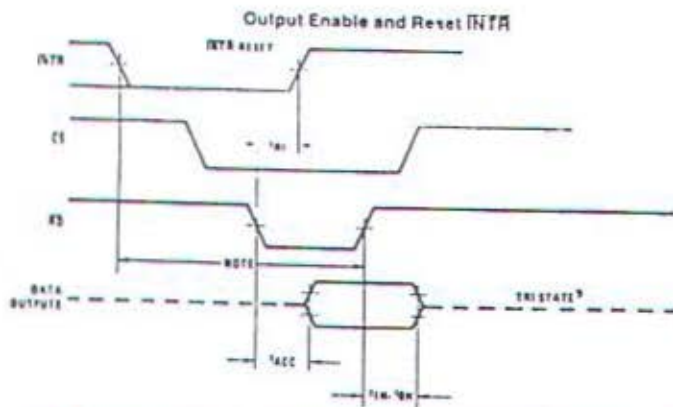
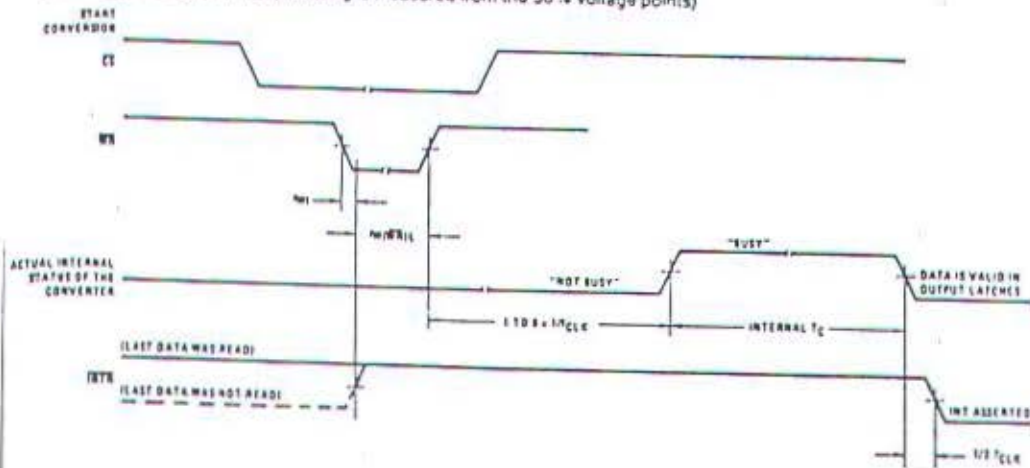
Linearity Error at Low VREF/2 Voltages



TRI-STATE Test Circuits and Waveforms



Timing Diagrams (All timing is measured from the 50% voltage points)



Note: Read strobe must occur 8 clock periods ($8/T_{CLK}$) after assertion of interrupt to guarantee reset of $\overline{\text{INTN}}$.

Karakteristik

		min	typ	max	
$V_S = \pm 15 \text{ V}$, $T_{amb} = 25 \text{ }^\circ\text{C}$					
Open loop supply	LF 355 N	I_S	2	4	mA
current consumption	LF 356 N, LF 357 N	I_S	5	10	mA
Input offset voltage	($R_G = 50 \text{ } \Omega$)	V_{IO}	3	10	mV
Input offset current		I_{IO}	3	50	pA
Input current		I_I	30	200	pA
Input resistance		R_I	10^{12}		Ω
Open loop voltage gain		A_{VO}	80	106	dB
Rate of rise					
	LF 355 N: $A_V = 1$	$\frac{dvq}{dtr}$	5		V/ μs
	LF 356 N: $A_V = 1$		12		V/ μs
	LF 357 N: $A_V = 5$		50		V/ μs
Performance	LF 355 N	f_p	2.5		MHz
bandwidth	LF 356 N	f_p	5		MHz
	LF 357 N	f_p	20		MHz
Transient time (for 0.01%)					
	LF 355 N	t_r	4		μs
	LF 356 N, LF 357 N	t_r	1.5		μs
Input noise voltage					
$R_S = 100 \text{ } \Omega$; $f = 100 \text{ Hz}$:	LF 355 N	V_{IN}	25		nV/ $\sqrt{\text{Hz}}$
	LF 356 N, LF 357 N	V_{IN}	15		nV/ $\sqrt{\text{Hz}}$
$R_S = 100 \text{ } \Omega$; $f = 1000 \text{ Hz}$:	LF 355 N	V_{IN}	20		nV/ $\sqrt{\text{Hz}}$
	LF 356 N, LF 357 N	V_{IN}	12		nV/ $\sqrt{\text{Hz}}$
Input noise current					
$f = 100 \text{ Hz}$, or 1000 Hz		I_{IN}	0.01		pA/ $\sqrt{\text{Hz}}$
Input capacitance		C_I	3		pF

Karakteristik

$V_S = \pm 15 \text{ V}$; $T_{amb} = 0 \text{ to } 70 \text{ }^\circ\text{C}$,
unless otherwise specified

Input offset voltage $R_G = 50 \text{ } \Omega$		V_{IO}		14	mV	
Temperature coefficient of V_{IO} : $R_S = 50 \text{ } \Omega$		α_{VIO}	5		$\mu\text{V/K}$	
Change of α_{VIO} after a change of V_{IO} adjustment ¹⁾		$\Delta\alpha_{VIO}$	0.5		per mV	
Input offset current $T_j = 70 \text{ }^\circ\text{C}$		I_{IO}		2	nA	
Input current ²⁾ $T_j = 70 \text{ }^\circ\text{C}$		I_I		8	nA	
Open loop voltage gain						
$R_L = 2 \text{ k}\Omega$, $V_{Oapp} = \pm 10 \text{ V}$		A_{VO}	63		dB	
Output voltage $R_L = 10 \text{ k}\Omega$		V_{Oapp}	12	± 13	-12	V
$R_L = 2 \text{ k}\Omega$		V_{Oapp}	10	± 12	-10	V
Input common mode range		V_{IC}	+11	+12	-11	V
Common mode rejection		K_{CMR}	80	100		dB
Supply voltage rejection		K_{SVR}	80	100		dB

Catatan:

- 1) Kalau dibandingkan dengan harga asli yang tak dapat ditepatkan, koefisien suhu dari tegangan gelincir masukan yang telah ditepatkan hanya berubah sedikit (lumrahnya $0,5 \mu\text{V/K}$) untuk setiap mV dalam jangkah stelan. Penepatan tegangan gelincir tidaklah berpengaruh kepada tindasan ragam tunggal (*common mode rejection*) dan kepada penguatan ikal terbuka.
- 2) Arus masukan berlipat hampir dua-kali, kalau suhu pertemuan naik 10 K.

LF 355N, LF 356N, LF 357N

Penguat Operasi Masukan JFET (JFET Input Operational Amplifiers)

Penguat-penguat operasi ini memiliki transistor-transistor masukan JFET, dengan arus-arus gelincir dan arus-arus masukan sangat kecil. Keluarannya dirancang untuk beban bersifat kapasitas tinggi tanpa sesuatu persoalan stabilitas.

Sifat-sifat tambahan:

- Resistansi masukan sangat tinggi
- Sedikit hanyut oleh perubahan suhu
- Lebarjalur lebar
- Dibolehkan tegangan masukan tinggi sampai $+V_s$
- Kompensasi frekuensi intern

Tarif Maksimum

Tegangan catu	V_s	± 18	V
Tegangan masukan diferensial	V_{ID}	± 30	V
Lama hubungsingkat keluaran	t_{Osc}	∞	
Jangkah suhu simpan	T_s	$-55 - 125$	$^{\circ}C$
Suhu pertemuan	T_j	100	$^{\circ}C$
Resistansi termik antara sistem-udara lingkungan	$R_{th\text{samb}}$	175	K/W

Konfigurasi pena

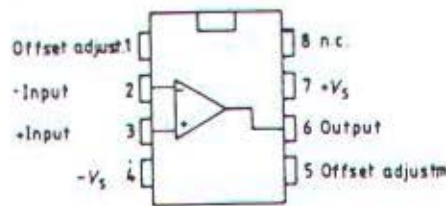
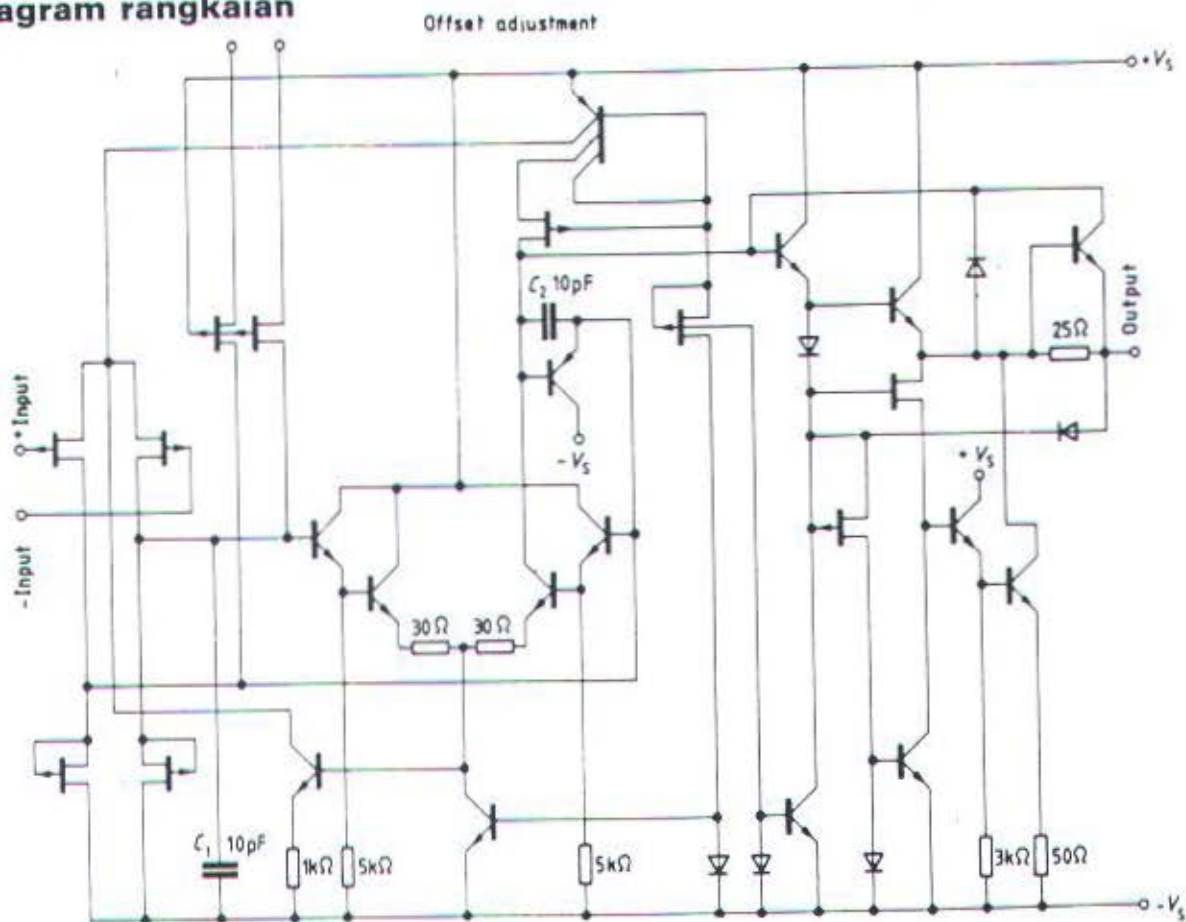


Diagram rangkaian



DAFTAR RIWAYAT HIDUP



Nama : Sri Damayanti
NRP : 2297030005
Tempat/Tanggal Lahir : Surabaya, 7 Desember 1977
Alamat : Jl. Kalikepiting No. 47/33
Surabaya, 60132
E-mail : dhamas97@yahoo.com

Riwayat Pendidikan :

1. (1985 – 1991) SDN Pacarkembang IV No. 195 Surabaya
2. (1991 – 1994) SMP Negeri IX Surabaya
3. (1994 – 1997) SMU Negeri 4 Surabaya
4. (1997 – 2001) Bidang Studi Komputer Kontrol, Program Studi Diploma III, Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS) Surabaya.

Pengalaman Organisasi :

1. Anggota Himpunan Mahasiswa Computer Control periode 1997–2001.
2. Instruktur BBCC dan BLCC 1998.

DAFTAR RIWAYAT HIDUP



Nama : M. Tamam Nafiri
NRP : 2297030073
Tempat/Tanggal Lahir : Lamongan, 21 Juli 1978
Alamat : Parengan RT 02 RW 03
Maduran - Lamongan
E-mail : m_tamam_n@yahoo.com

Riwayat Pendidikan :

1. (1984 – 1990) MIM I Pangkatrejo – Sekaran – Lamongan
2. (1990 – 1993) SMP Negeri 1 Sekaran – Lamongan
3. (1993 – 1996) SMU Negeri 1 Lamongan
4. (1997 – 2001) Bidang Studi Komputer Kontrol, Program Studi Diploma III, Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS) Surabaya.

Pengalaman Organisasi :

1. SKI Himpunan Mahasiswa Computer Control periode 1997–1998
2. OC BBCC dan BLCC 1998.