

21.418 / ITS / H / 05



1246

**RANCANG BANGUN SISTEM PENGENDALIAN  
LEVEL & LAJU ALIRAN PADA SUNGAI  
(AKSI KENDALI PADA PINTU AIR BENDUNGAN)  
UNTUK MENDETEKSI BANJIR BERBASIS  
MIKROKONTROLLER**

**TUGAS AKHIR**



RSf  
681.76  
Bud.  
F - 1  
2003,

Oleh :

**BUDI PORNOMO**

NRP 2400 109 504

PERPUSTAKAAN ITS	
Tgl. Terima	9-4-2003
Terima Dari	H1
No. Agenda Prp.	217116

**JURUSAN TEKNIK FISIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
2003**

**RANCANG BANGUN SISTEM PENGENDALIAN  
LEVEL & LAJU ALIRAN PADA SUNGAI  
(AKSI KENDALI PADA PINTU AIR BENDUNGAN)  
UNTUK MENDETEKSI BANJIR BERBASIS  
MIKROKONTROLLER**

**TUGAS AKHIR**

Oleh :

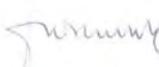
**BUDI PORNOMO**  
NRP : 2400 109 504

Surabaya, Februari 2003

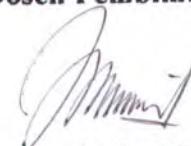
**Mengetahui/Menyetujui**

**Pembimbing**

**Dosen Pembimbing**

  
Ir. Ali Musyafa', M.Sc  
NIP 131 652 210

**Co. Dosen Pembimbing**

  
Imam Abadi, ST  
NIP 132 231 646

**Ketua Jurusan**



## **A B S T R A K**

## **ABSTRAK**

Kapasitas air dalam suatu tempat jika mendapat input yang berlebihan, misal debit, curah hujan yang lebih dari biasa akan menyebabkan tempat tersebut airnya berlimpah/banjir. Untuk mendeteksi secara dini dan mengetahui hal tersebut, maka perlu dikaji level air, kecepatan laju aliran air baik di tempat bendungan maupun di hulu sungai untuk mengantisipasi banjir yang berlebih.

Dengan pembuatan miniatur bendungan dan beberapa pintu air dengan mengetahui dan mengendalikan level air dan laju aliran dengan sistem otomatis maka telah dapat digunakan untuk mengetahui kondisi banjir.

Hasil pengambilan data pada ketinggian pintu air setelah dibandingkan dengan alat ukur standar ketinggian yaitu penggaris diperoleh ketidakpastian standart 0,03 cm untuk pintu I dan ketidakpastian standart 0,044 cm untuk pintu II.

## KATA PENGANTAR



## **KATA PENGANTAR**

Alhamdulillahirobbil'aalamin, segala puji dan syukur kami panjatkan kehadirat Allah SWT, atas berkat rahmat-Nya sehingga atas Ridho-Nya penulis bisa menyelesaikan Tugas Akhir yang berjudul :

### **RANCANG BANGUN SISTEM PENGENDALIAN LEVEL & LAJU ALIRAN PADA SUNGAI (AKSI KENDALI PADA PINTU AIR BENDUNGAN) UNTUK MENDETEKSI BANJIR BERBASIS MIKROKONTROLLER**

Tugas akhir ini merupakan salah satu syarat yang harus ditempuh bagi seorang mahasiswa Strata-1 untuk menyelesaikan masa studinya dan memperoleh gelar kesarjanaan di Teknik Fisika, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya.

Seiring dengan terselesainya penyusunan Tugas akhir ini, penyusun menyampaikan terima kasih dan penghargaan yang setinggi-tingginya atas bimbingan, petunjuk, saran serta kesempatan kepada semua pihak yang turut membantu dalam proses penyelesaian Tugas akhir ini kepada :

- Bapak Ir. Ya'umar, MT, selaku Ketua Jurusan Teknik Fisika FTI – ITS.
- Bapak Ir. Ali Musyafa', MSc, selaku Dosen Pembimbing Tugas akhir dan Dosen Wali.
- Bapak Imam Abadi, selaku Dosen Co Pembimbing Tugas akhir.
- Bapak dan Ibu Dosen serta karyawan Teknik Fisika ITS.
- Ayahanda Suhaimi dan Ibunda Siti Chamidah tercinta yang telah banyak memberikan dorongan dan mendo'akan diriku dalam menyelesaikan tugas akhir ini.

- Eny Agustin yang selalu mendampingi, mendo'akan dan membantu dalam penggerjaan tugas akhir.
- Teman tandingku seperjuangan tugas akhir Ahmad Firdaus, yang banyak bertukar fikiran dan ide dalam penggerjaan tugas akhir.
- Teman-teman kuliah Lintas Jalur angkatan 2000.
- Teman seperjuangan dari Politeknik Unibraw Malang Rimba Ary dan Yudi.
- Penghuni kost-an Klampis Ngasem 31 Conk Andy, Arif, A'an, Teguh, Madhan dan Bang Andy sekeluarga.
- Semua pihak yang turut membantu yang mungkin terlupakan untuk disebutkan satu persatu, hanya Tuhan yang mampu membalasnya.

Akhirnya betapapun kecilnya, semoga Tugas Akhir ini dapat memberi manfaat bagi pembaca yang memerlukan.

Surabaya, Februari 2003

Penyusun



## **DAFTAR ISI**

## DAFTAR ISI

	Halaman
HALAMAN JUDUL .....	i
LEMBAR PENGESAHAN .....	ii
ABSTRAK .....	iii
KATA PENGANTAR .....	iv
DAFTAR ISI .....	vi
DAFTAR GAMBAR .....	viii
DAFTAR TABEL .....	x
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Maksud dan Tujuan .....	2
1.3 Permasalahan .....	2
1.4 Batasan Masalah .....	3
1.5 Metodologi .....	3
1.6 Sistematika Pembahasan .....	4
BAB II DASAR TEORI .....	5
2.1 Bendungan .....	5
2.2 Pintu Air .....	11
2.3 Motor Stepper .....	15
2.4 Potensiometer .....	19
2.5 PPI 8255 .....	21
2.6 Analog To Digital Converter .....	24
BAB III PERANCANGAN DAN PEMBUATAN ALAT .....	29
3.1 Cara Kerja Rangkaian .....	29
3.2 Perancangan Perangkat Keras .....	31
3.3 Perancangan Sistem Mekanik .....	36
3.4 Perancangan Bentuk Miniplant Bendungan .....	37
BAB IV PENGUJIAN DAN PENGUKURAN .....	38
4.1 Pengujian Perangkat Keras .....	38

4.2 Pengujian Perangkat Lunak.....	41
4.3 Pengukuran Data Pintu Air .....	44
4.4 Analisa Data Pengukuran Ketinggian Pintu Air .....	45
BABA V KESIMPULAN.....	52
DAFTAR PUSTAKA	
LAMPIRAN	



## **DAFTAR GAMBAR**

## DAFTAR GAMBAR

	Halaman
2.1 Bendung pembagi banjir .....	6
2.2 Bendung penahan air pasang.....	6
2.3 Bendung penyadap .....	7
2.4 Denah bendung dan arah arus sungai.....	9
2.5 Limpahan sempurna .....	10
2.6 Limpahan melekat.....	10
2.7 Limpahan tidak sempurna .....	10
2.8 Limpahan tidak sempurna .....	10
2.9 Pintu geser roda.....	11
2.10 Pintu engsel dipasang pada permukaan lantai .....	12
2.11 Pintu Radial.....	13
2.12 Pintu rol.....	14
2.13 Penampang melintang dari motor stepper tipe variable reluctance .....	15
2.14 Ilustrasi sederhana dari motor stepper tipe permanent magnet .....	16
2.15 Penampang melintang dari motor stepper tipe hibrid .....	17
2.16 Motor stepper dengan lilitan unipolar.....	17
2.17 Motor stepper dengan lilitan bipolar.....	18
2.18 Representasi potensiometer .....	20
2.19 Blok diagram PPI 8255 .....	21
2.20 Register control word PPI 8255 .....	23
2.21 Blok Diagram SAC ADC .....	24
3.1 Diagram Blok 1 .....	29
3.2 Diagram blok 2 .....	30
3.3 Rangkaian driver motor stepper .....	32
3.4 Rangkaian ADC 0804 .....	33
3.5 Rangkaian potensiometer.....	35
3.6 Bentuk sistem mekanik bendungan (tampak dari depan) .....	36
3.7 Bentuk miniplant bendungan dan pintu air.....	37

4.1	Pengujian blok diagram ADC 0804 .....	38
4.2	Tampilan program.....	42
4.3	Tampilan menu setting pintu .....	43
4.4	Blok diagram pengujian pintu air I dan pintu air II .....	44
4.5	Grafik ketinggian pintu air I vs ketidakpastian standart .....	49
4.6	Grafik ketinggian pintu air II vs ketidakpastian standart.....	51

## **DAFTAR TABEL**

## DAFTAR TABEL

	Halaman
2.1 Mode operasi PPI 8255 .....	22
3.1 Control word PPI 8255.....	34
4.1 Hasil pengukuran ADC .....	39
4.2 Hasil pengujian PPI 8255.....	40
4.3 Data pintu air I .....	45
4.4 Data pintu air II .....	45
4.5 Data hasil perhitungan pada alat buatan untuk pintu I.....	47
4.6 Data hasil perhitungan pada alat buatan untuk pintu II .....	47
4.7 Data perhitungan ketidakpastian standart pintu I .....	49
4.8 Data perhitungan ketidakpastian standart pintu II .....	50

**BAB I**  
**PENDAHULUAN**

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Sebagian besar air hujan yang turun ke permukaan tanah, mengalir ke tempat-tempat yang lebih rendah dan akhirnya melimpah ke danau atau ke laut. Suatu alur yang panjang di atas permukaan bumi tempat mengalirnya air yang berasal dari hujan disebut alur sungai. Dan perpaduan antara alur sungai dan aliran air didalamnya disebut sungai.

Sungai-sungai sebagai saluran pembuang terbentuk secara alamiah dan berfungsi sebagai saluran penampung air hujan. Disaat terjadinya hujan yang lebat, sungai-sungai menampung volume air yang besar dan segera mengalirkannya menuju laut atau danau, bahkan kadang-kadang air banjir meluap keluar sungai, karena volume air melebihi daya tampung normalnya.

Pada kondisi level ketinggian yang kritis perlu ada kendali yang mengatur pembukaan pintu air secara otomatis. Hal ini digunakan untuk menghindari bahaya yang lebih besar, yaitu banjir di daerah hulu atau bahkan jebolnya bendungan air karena pembebanan yang berlebih. Pengaturan ketinggian otomatis bekerja terus sampai dicapai level ketinggian air yang aman.

Pemakaian kontrol utama dari bendungan air yang memiliki beberapa pintu air sangat penting untuk mendapatkan pengaturan optimal dan handal. Sehingga untuk mengatur level ketinggian pintu air tertentu dapat dikendalikan operator dari pusat kendali. Cara ini lebih handal daripada cara manual, karena

penggunaan motor untuk menggerakkan pintu air diperoleh hasil yang lebih presisi dan aman.

Kehadiran alat ini diharapkan dapat membantu operator bendungan dalam mengoptimalkan pengaturan air di bendungan yang memiliki beberapa pintu air. Mengamankan dari bahaya luapan air dan jebolnya bendungan dari beban berlebih. Dengan adanya proteksi terhadap gangguan maka pemakaian alat dapat efektif dan efisien. Sehingga didapatkan pengendali bendungan yang dapat dikendalikan secara terpadu yang handal dan aman.

## 1.2 Maksud dan Tujuan

Adapun tujuan dari pembuatan tugas akhir ini adalah untuk rancang bangun alat pengendali ketinggian pintu air bendungan yang digunakan untuk buka dan tutup pintu air dengan dikendalikan oleh komputer, sehingga didapatkan sistem pengendalian yang handal dan aman.

## 1.3 Permasalahan

Berdasarkan latar belakang diatas maka pengendalian pintu air yang konvensional itu mempunyai beberapa kelemahan, baik ditinjau dari unjuk kerja maupun keamanan. Oleh karena itu penulis berusaha untuk memecahkan permasalahan yang ada yaitu:

- Masih dijumpai penggunaan pintu air secara manual, dan pengoperasiannya membutuhkan tenaga yang besar dan waktu yang cukup lama.
- Membuat pengendali pintu air yang dapat bekerja secara simultan untuk beberapa pintu air.

#### 1.4 Batasan Masalah

Untuk memperjelas ruang lingkup permasalahan dan mempermudah pembahasan analisa masalah dalam perancangan alat maka diperlukan pembatasan masalah sebagai berikut :

- Pembuatan alat ini hanya dikhususkan untuk buka dan tutup pintu air pada darurat I, II, dan III yang dikendalikan oleh komputer dengan bentuk miniatur.
- Dalam pembuatan alat ini tidak membahas gangguan-gangguan pada aliran air ataupun level, misal sampah, gerakan sesaat.

#### 1.5 Metodologi

Adapun langkah-langkah yang diambil dalam penyusunan tugas akhir adalah sebagai berikut :

- Studi literatur tentang peralatan yang akan dibuat.
- Melakukan survay lapangan guna mengumpulkan data mengenai bendungan dan peralatan yang digunakan.
- Membuat perencanaan mekanik, hardware dan software untuk pembuatan pengendalian pintu air.
- Berdasarkan perencanaan yang telah disusun maka dibuat pembuatan pengendalian pintu air.
- Melakukan uji coba alat yang sudah di coba di laboratorium.
- Penguyusan tugas akhir.

## 1.6 Sistematika Pembahasan

Sistematika dari tugas akhir ini disusun sebagai berikut :

Bab I : pendahuluan pada bagian pendahuluan menerangkan secara garis besar latar belakang, maksud dan tujuan, permasalahan, batasan masalah, metodologi yang digunakan, sistematika pembahasan. Bab II : dasar teori pada bagian dasar teori akan dibahas mengenai bendungan, pintu air, motor stepper, potensiometer, PPI 8255, ADC 0804. Bab III : perancangan dan implementasi sistem bab ini berisi tentang perancangan seluruh rangkaian berisikan tentang cara kerja blok diagram, perancangan rangkaian dari alat yang dibuat, perlengkapan mekanik yang digunakan dalam perancangan. Bab IV : pengujian dan analisa berisi tentang analisa pengujian peralatan yang dibuat, apakah sesuai dengan perencanaan yang dilakukan sebelumnya. Bab V : penutup pada bab ini berisi kesimpulan dan saran dari seluruh perencanaan dan pembuatan alat untuk penyempurnaan sistem. Kesimpulan diambil dari hasil pengujian sistem, analisa masalah dan saran-saran untuk pengembangan berikutnya.

**BAB II**  
**DASAR TEORI**

## **BAB II**

### **DASAR TEORI**

Pada bab ini akan dibahas mengenai dasar teori dari modul dan peralatan yang akan direncanakan, dengan tujuan agar lebih mudah dipahami mengenai latar belakang dalam perencanaan alat. Materi yang akan dibahas meliputi pengertian dasar tentang bendungan, pintu air, motor stepper, potensiometer, sistem pengendalian, PPI 8255, ADC (Analog to Digital Converter).

#### **2.1 Bendungan**

Air merupakan kebutuhan utama manusia, Ketersediannya harus ada dalam setiap waktu. Namun kondisi alam menyebabkan pada masa aliran tinggi, air yang tersedia begitu melimpah sehingga menyebabkan bahaya banjir. Akan tetapi sebaliknya pada musim aliran rendah atau musim kemarau air yang tersedia kurang.

Oleh karena itu dibutuhkan suatu sarana untuk mengatasi kendala alam diatas. Disamping menampung air untuk pemanfaatan air dikemudian hari, hal ini akan mengurangi resiko akibat banjir di hilir bendungan. Berapapun ukuran bendungan dan bagaimanapun tujuan akhir dari pemanfaatan airnya, fungsi utama dari yang berubah-ubah pada suatu sungai alamiah, maupun dengan cara memenuhi kebutuhan konsumen yang berubah-ubah.

### 2.1.1 Jenis-jenis Bendungan

Berdasarkan fungsinya bendung dapat dibedakan dalam bendung pembagi banjir, bendung penahan air pasang dan bendung penyadap.

#### a. Bendung Pembagi Banjir.

Bendung semacam ini didirikan pada percabangan sungai untuk mengatur muka air, sehingga terjadi pemisahan antara debit banjir dan debit rendah sesuai dengan kapasitas yang telah ditetapkan sebelumnya.



Gambar 2.1 Bendung pembagi banjir <sup>[3]</sup>

#### b. Bendung Penahan Air Pasang.

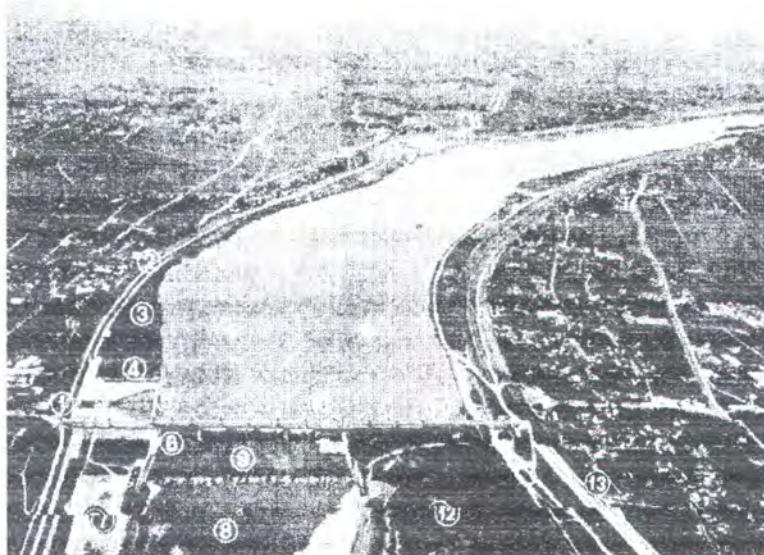
Bendung ini dibangun di bagian sungai yang dipengaruhi pasang-surut air laut untuk mencegah masuknya air asin dan untuk menjamin agar aliran air sungai senantiasa dalam keadaan normal.



Gambar 2.2 Bendung penahan air pasang <sup>[3]</sup>

### c. Bendung Penyadap.

Bendung ini digunakan untuk mengatur muka air di dalam sungai guna memudahkan penyadapan airnya untuk keperluan air minum, air perkotaan, irigasi dan pembangkit tenaga air.



Gambar 2.3 Bendung penyadap<sup>[3]</sup>

#### 2.1.2 Tempat kedudukan Bendungan

Pada hakekatnya antara kegiatan pengendalian banjir dan pemanfaatan potensi sungai, saling berkaitan secara erat. Dengan demikian merencanakan bendung pada sebuah sungai harus dengan sungguh-sungguh memperhatikan pula kaitannya dengan bangunan-bangunan persungai lainnya baik yang sudah didirikan maupun yang sedang direncanakan.

Jadi dalam penentuan tempat kedudukan, formasi dan dimensi bendung supaya diperhatikan hal-hal sebagai berikut :

- a. Lokasi bendung penyadap diusahakan sedapat mungkin lebih ke hulu, agar bendung tidak terlalu tinggi.

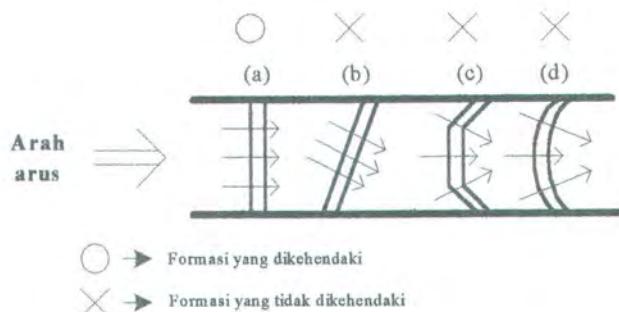
- b. Bendung pembagi banjir supaya ditempatkan dekat titik percabangan sungai.
- c. Diusahakan agar bendung ditempatkan pada ruas sungai yang lurus dengan penampang yang konstan, di mana perubahan kecepatan arus tidak terlalu drastis.
- d. Bendung supaya ditempatkan pada ruas sungai yang alurnya stabil dan perubahan dasar sungai tidak mencolok.
- e. Diusahakan agar pengaruh air balik akibat bendung tidak terlalu jauh ke hulu.
- f. Diusahakan agar bendung terletak pada tanah pondasi yang baik.

### **2.1.3 Formasi Bendung**

Untuk memudahkan penyadapan air sungai kadang-kadang digunakan bendung tipe diagonal atau tipe lengkung, tetapi pada prinsipnya arus horisontal bendung, sebaiknya tegak lurus arah aliran banjir.

Apabila lintasan arus pada alur sungai yang sedang banjir tidak sama dengan lintasan arus normalnya, maka alur sungai hendaknya diperbaiki. Apabila hal tersebut sulit dilaksanakan, maka perhatian khusus supaya diarahkan pada penempatan bendung sedemikian, sehingga di waktu banjir lintasan arus air tegak lurus bendung. Seandainya tempat kedudukan bendung harus ditempatkan pada ruas sungai yang berbelok-belok, maka dipandang dari segi banjir perlu diadakan pengamanan, misalnya dengan pembuatan perkuatan lereng tanggul sampai pada mercunya.

Penempatan denah dan formasi bendung terhadap arah arus sungai dapat dilihat dibawah ini :



Gambar 2.4 Denah bendung dan arah arus sungai [3]

#### 2.1.4 Limpasan Bendung

Perencanaan bendung pertama-tama dimulai dengan meneliti kondisi aliran sungai melalui bendung dan kemudian meneliti pengaruh aliran balik di sebelah hulunya.

##### a. Pelimpah

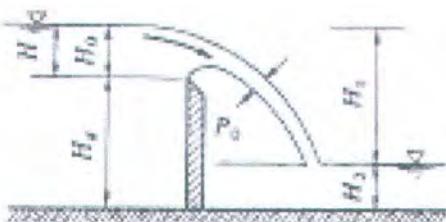
Bentuk pelimpah pada bendung sangat bervariasi, tergantung dari tinggi bendung, muka air di hilir dan di hulu bendung dan sebagainya.

- Apabila  $H_d > 2,5H$  : Akan terjadi bentuk limpahan seperti pada gambar 2.5 dan udara bergerak bebas di sisi hilir dinding peluap. Bentuk limpahan semacam ini disebut limpahan sempurna.
- Apabila  $H$  jauh lebih kecil dari  $0,4 H_d$  : Akan terbentuk limpahan seperti pada gambar 2.6 yang disebut limpahan melekat.

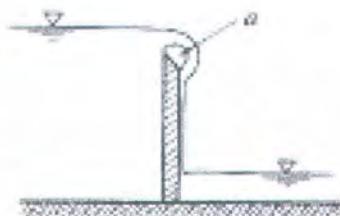
- Apabila  $H_d \leq 2,5H$  : Dalam hal ini bentuk limpahan tidak begitu jelas seperti yang terlihat pada gambar 2.7. Pelimpahan semacam ini disebut limpahan tidak sempurna.
- Apabila  $H_l \leq 0,75H_d$  : Jika perbedaan elevasi muka air di hulu dan di hilir bendung mengecil (gambar 2.8), maka semakin nyata terlihat bentuk limpahan tidak sempurna. Apabila  $H_l \leq 0,75H_d$ , olakan aliran air di sebelah hilir pelimpahan semakin intensif.

### b. Tipe bendung dan kondisi limpahan

Sesuai dengan tipe bendung, maka kondisi limpahan dapat dilihat pada skema berikut ini :



Gambar 2.5 Limpahan sempurna <sup>[3]</sup>



Gambar 2.6 Limpahan melekat <sup>[3]</sup>



Gambar 2.7 Limpahan tidak sempurna <sup>[3]</sup>



Gambar 2.8 Limpahan tidak sempurna <sup>[3]</sup>

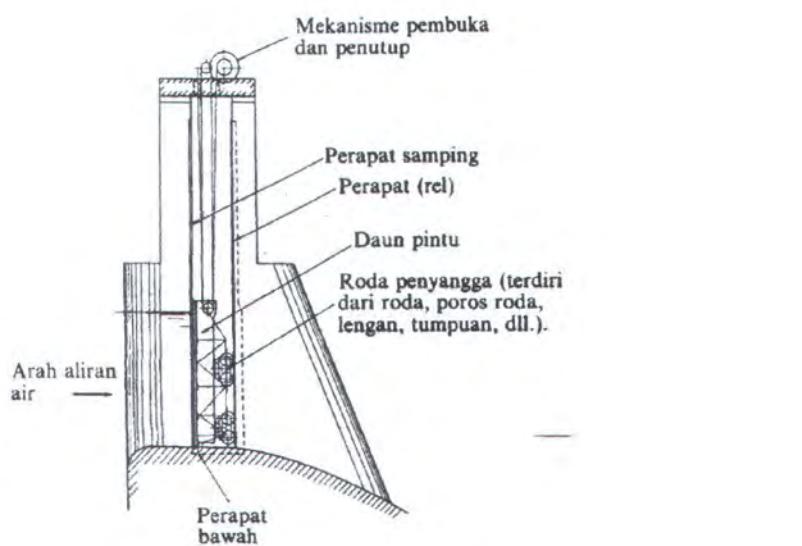
## 2.2 Pintu Air

Dalam perencanaan hampir setiap bendungan harus difikirkan cara untuk mengalirkan air ke hilir. Suatu pelimpahan (spillway) diperlukan untuk mengalirkan air dan mencegah rusaknya bendungan. Macam-macam pintu air :

### 2.2.1 Pintu Geser Roda

Pintu geser roda seperti yang terlihat digambar 2.9 adalah tipe pintu khusus yang memindahkan tekanan air dari pelat daun pintu ke balok horisontal utama pintu tersebut melalui balok-balok horisontal. Selanjutnya tekanan air dari balok-balok horisontal utama diteruskan ke balok-balok vertikal utama pada ke dua tepi pintu dan dari balok-balok diteruskan ke roda-roda penyangga.

Pada pintu yang dioperasikan dengan pemutaran, maka daun pintu digantung dengan kabel baja atau batang ulir. Konstruksi pintu tipe ini sangat sederhana dan gesekan yang terjadi pada saat pembukaan-penutupan dapat dikurangi dengan bantuan roda-roda, selain itu kerapatan airnya sangat tinggi. Oleh karenanya pintu tipe ini sangat banyak digunakan.

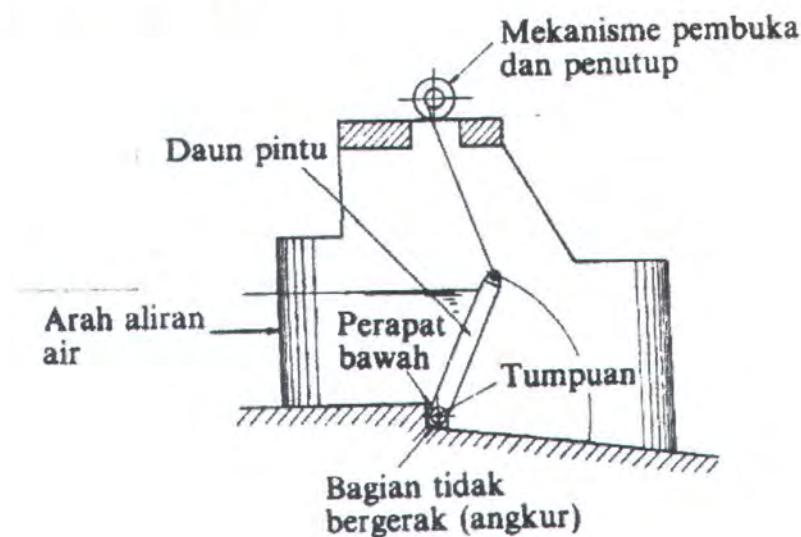


Gambar 2.9 Pintu geser roda <sup>[3]</sup>

### 2.2.2 Pintu Engsel

Pintu engsel, seperti yang tertera pada gambar 2.10 adalah salah satu tipe pintu air yang terbuka dan tertutup dengan gerakan berputar pada poros horizontal. Poros horizontal berbentuk engsel ditempatkan pada pinggir bawah daun pintunya dan dilekatkan di atas lantai atau mercu pelimpah. Akan tetapi kelemahan dari pintu tipe ini pada saat dilalui aliran air, timbul gerakan yang cukup kuat. Akibatnya bekerjanya engsel kurang sempurna atau mudah terjadi kerusakan pada daun pintu serta mekanisme operasi pintu.

Mengingat banyaknya kelemahan pada pintu engsel, maka pintu tipe ini dibuat tidak melebihi tinggi 3 meter.



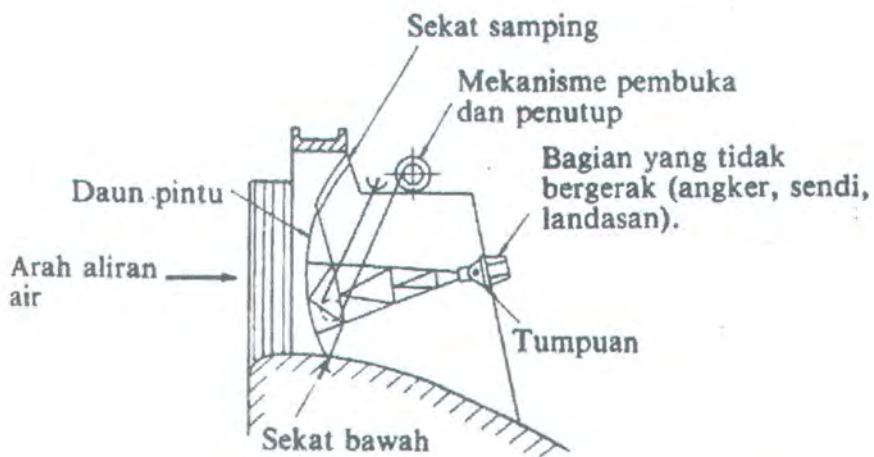
Gambar 2.10 Pintu engsel dipasang pada permukaan lantai [3]

### 2.2.3 Pintu Radial

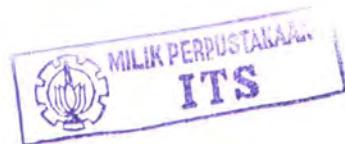
Pintu radial terdiri dari daun pintu berbentuk busur, balok utama dan kaki. Permukaan daun pintu dibuat dari pelat baja dan tekanan air disangga oleh sendi. Kelebihan dari pintu radial ini terutama adalah celah bukaannya tidak terlalu

tinggi, karena gerakannya berputar mengelilingi sendinya dan harganya lebih murah dibandingkan dengan tipe lainnya. Sebaliknya pembuatannya sangat sulit, karena merupakan konstruksi tiga dimensi, oleh sebab itu disain, pembuatan dan pemasangannya harus dilaksanakan dengan sangat hati-hati.

Selanjutnya bagian-bagian dari pintu jenis ini sangat ramping dan seluruh beban terpusat pada sendi. Akan tetapi pintu radial ini sangat lemah terhadap gaya-gaya limpasan.



Gambar 2.11 Pintu radial <sup>[3]</sup>



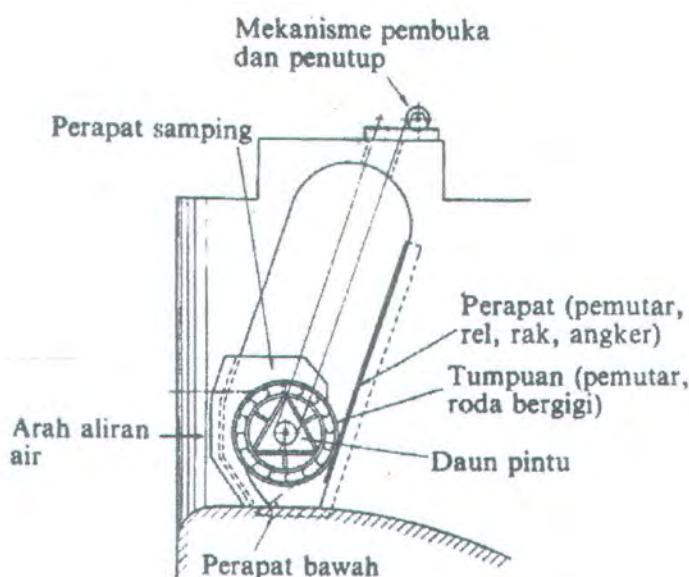
#### 2.2.4 Pintu rol

Tipe ini digunakan pada pintu air yang lebar, sehingga membutuhkan tubuh pintu yang panjang. Konstruksi tubuh pintunya terdiri dari beberapa balok horizontal yang arahnya terpusat pada satu titik dan dibungkus dengan pelat baja membentuk silinder.

Untuk memperkuat daun pintu, maka balok-balok horizontal tersebut ditambah dengan balok pembantu vertikal. Pada balok pembantu vertikal yang paling pinggir di kedua ujung tubuh pintu dipasang gigi dan dikaitkan dengan rel

gigi yang dipasang pada pilar bendung. Pada pintu yang bukaannya tinggi, guna mengurangi beratnya, maka dibagian bawah pintu dipasang semacam sayap. Pintu rol mempunyai kekakuan yang tinggi, oleh sebab itu cocok untuk sungai-sungai yang banyak membawa pasir dan krikil atau yang banyak menghanyutkan batang-batang pohon.

Pintu tipe ini mempunyai beberapa kekurangan yang terutama adalah stabilitasnya rendah, karena pada saat terjadi pelimpahan, pintu dalam posisi mengambang oleh gaya apung air yang menyebabkan terjadinya perbedaan yang besar pada beban operasinya. Selain itu bobot tubuh pintu sangat besar dan mekanisme pemutarnya cukup mahal.



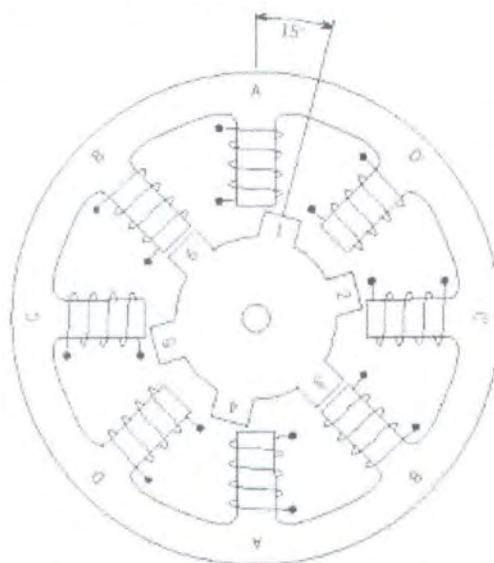
Gambar 2.12 Pintu rol <sup>[3]</sup>

### 2.3 Motor Stepper

Motor stepper adalah perangkat elektromekanis yang bekerja dengan mengubah pulsa elektronis menjadi gerakan mekanis diskrit. Motor stepper bergerak berdasarkan urutan pulsa yang diberikan kepada motor. Karena itu, untuk menggerakkan motor stepper diperlukan pengendali motor stepper yang membangkitkan pulsa-pulsa periodik. Pada dasarnya terdapat 3 tipe motor stepper yaitu:

#### a. Motor stepper tipe *Variable reluctance* (VR)

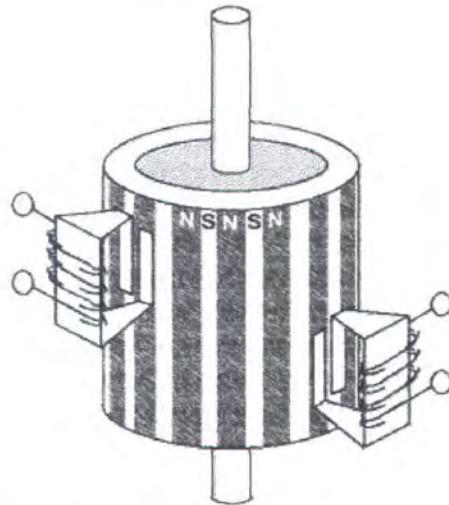
Motor stepper jenis ini telah lama ada dan merupakan jenis motor yang secara struktural paling mudah untuk dipahami. Motor ini terdiri atas sebuah rotor besi lunak dengan beberapa gerigi dan sebuah lilitan stator. Ketika lilitan stator diberi energi dengan arus DC, kutub-kutubnya menjadi termagnetasi. Perputaran terjadi ketika gigi-gigi rotor tertarik oleh kutub-kutub stator. Berikut ini adalah penampang melintang dari motor stepper tipe *variable reluctance* (VR):



Gambar 2.13 Penampang melintang dari motor stepper tipe *variable reluctance* [5]

### b. Motor stepper tipe *Permanent Magnet* (PM)

Motor stepper jenis ini memiliki rotor yang berbentuk seperti kaleng bundar (*tin can*) yang terdiri atas lapisan magnet permanen yang diselang-seling dengan kutub yang berlawanan (perhatikan gambar 2.14). Dengan adanya magnet permanen, maka intensitas fluks magnet dalam motor ini akan meningkat sehingga dapat menghasilkan torsi yang lebih besar. Motor jenis ini biasanya memiliki resolusi langkah (*step*) yang rendah yaitu antara  $7,5^0$  hingga  $15^0$  per langkah atau 48 hingga 24 langkah setiap putarannya. Berikut ini adalah ilustrasi sederhana dari motor stepper tipe *permanent magnet*:



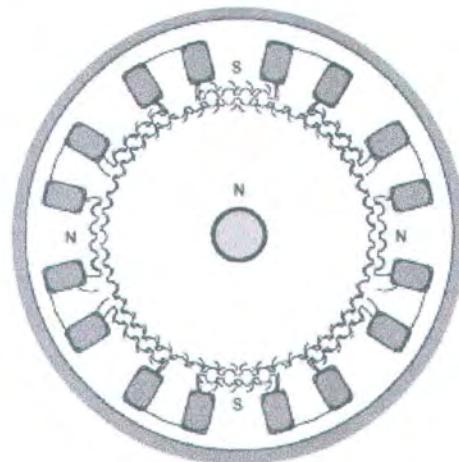
Gambar 2.14 Ilustrasi sederhana dari motor stepper

tipe *permanent magnet* (PM)<sup>[5]</sup>

### c. Motor stepper tipe *Hybrid* (HB)

Motor stepper tipe hibrid memiliki struktur yang merupakan kombinasi dari kedua tipe motor stepper sebelumnya. Motor stepper tipe hibrid memiliki gigi-gigi seperti pada motor tipe VR dan juga memiliki magnet permanen yang tersusun secara aksial pada batang porosnya seperti motor tipe PM. Motor tipe ini

paling banyak digunakan dalam berbagai aplikasi karena kinerja lebih baik. Motor tipe hibrid dapat menghasilkan resolusi langkah yang tinggi yaitu antara  $3,6^0$  hingga  $0,9^0$  per langkah atau 100-400 langkah setiap putarannya.

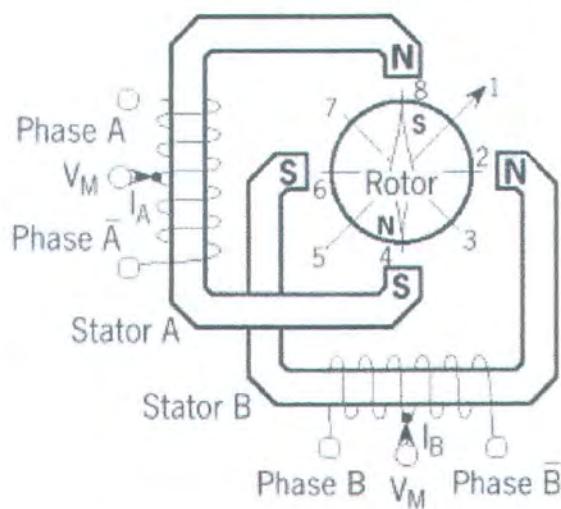


Gambar 2.15 Penampang melintang dari motor stepper tipe hibrid [5]

Berdasarkan metode perancangan rangkaian pengendalinya, motor stepper dapat dibagi menjadi jenis unipolar dan bipolar.

#### a. Unipolar motor stepper

Rangkaian pengendali motor stepper unipolar lebih mudah dirancang karena hanya memerlukan satu switch / transistor setiap lilitannya.

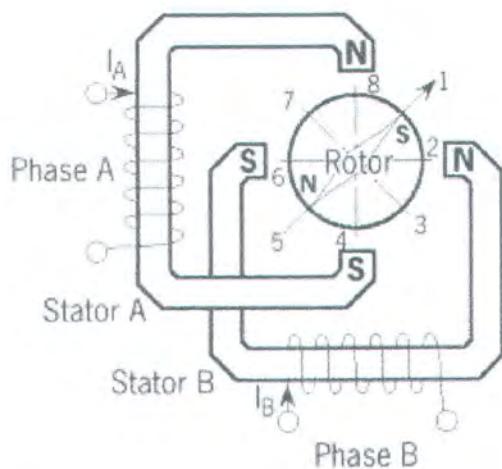


Gambar 2.16 Motor stepper dengan lilitan unipolar [5]

Untuk menjalankan dan menghentikan motor ini cukup dengan menerapkan pulsa digital yang hanya terdiri atas tegangan positif dan nol (*ground*) pada salah satu terminal lilitan (*wound*) motor sementara terminal lainnya dicatut dengan tegangan positif konstan ( $V_M$ ) pada bagian tengah (*center tap*) dari lilitan.

### b. Bipolar motor stepper

Untuk motor stepper dengan lilitan bipolar, diperlukan sinyal pulsa yang berubah-ubah dari positif ke negatif dan sebaliknya. Jadi pada setiap terminal lilitan (A & B) harus dihubungkan dengan sinyal yang mengayun dari positif ke negatif dan sebaliknya (perhatikan gambar 2.17). Karena itu dibutuhkan rangkaian pengendali yang agak lebih kompleks daripada rangkaian pengendali untuk motor unipolar. Motor stepper bipolar memiliki keunggulan dibandingkan dengan motor stepper unipolar dalam hal torsi yang lebih besar untuk ukuran yang sama.



Gambar 2.17 Motor stepper dengan lilitan bipolar [5]

## 2.4 Potensiometer

Potensiometer merupakan tranduser elektromekanik yang mengubah energi mekanik menjadi energi Listrik. Perubahan mekanik disini adalah berkaitan dengan displacement, yang dapat berupa gerakan putar atau lurus. Masukan potensiometer adalah perpindahan mekanik, baik linier atau putaran. Ketika suatu tegangan dipasang melintasi ujung tertentu dari potensiometer, tegangan keluaran, yang diukur melintasi ujung variable dan ground adalah sebanding dengan perpindahan masukan, baik bersifat linier ataupun menurut beberapa hubungan nonlinier.

Instrumen potensiometer yang direncanakan untuk mengukur tegangan yang tidak diketahui dengan cara membandingkannya terhadap tegangan yang diketahui. Tegangan yang diketahui dapat disuplai dari sebuah sel standar atau setiap sumber tegangan referensi yang diketahui. Pengukuran-pengukuran dengan menggunakan cara perbandingan mampu menghasilkan tingkat ketelitian yang sangat tinggi sebab hasil yang diperoleh tidak bergantung pada defleksi aktual jarum penunjuk sebagaimana halnya pada instrumen kumparan putar, tetapi hanya bergantung pada ketelitian tegangan standar yang diketahui terhadap mana perbandingan dilakukan.

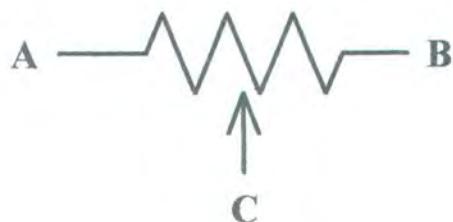
Resistor variabel atau potensiometer banyak macam dan bentuknya antara lain :

Berdasarkan bahannya potensiometer dibedakan menjadi :

- a. Potensiometer arang yang dapat diputar atau digeser
- b. Potensiometer kawat logam (bahan resistansi konduktif).

Berdasarkan pengemasannya potensiometer dibedakan menjadi :

- Potensiometer mono adalah potensiometer yang terdiri dari satu VR dalam satu kemasan
- Potensiometer stereo adalah potensiometer yang terdiri dari 2 buah VR yang seporos, artinya bila tuas diputar, maka dua-duanya ikut berubah.
- Potensiometer ber - CT (Center Tap = cabang tengah) adalah potensiometer yang mempunyai 1 kaki tambahan yang bila diukur terhadap kaki utama nilai resistansinya separuh nilai resistansi total.



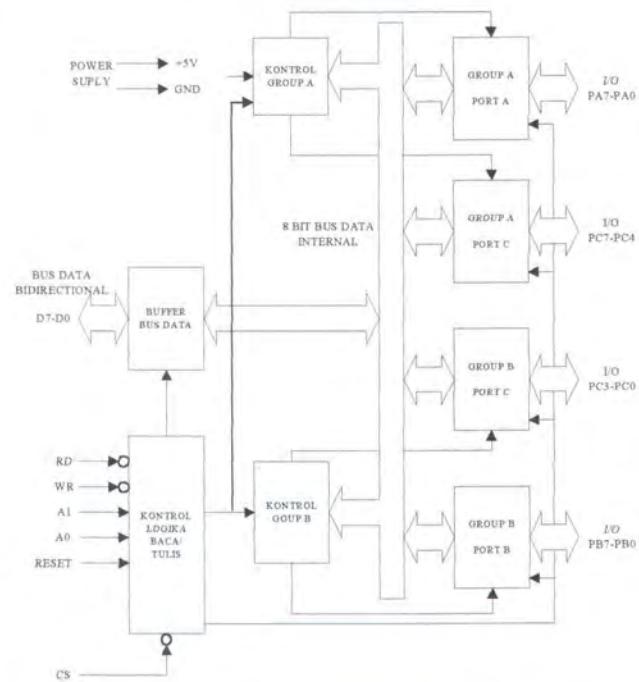
Gambar 2.18 Representasi Potensiometer

Potensiometer standar mempunyai tiga kaki (seperti gambar 2.18 diatas).

Kaki A dan B merupakan kaki utama yang mempunyai hambatan tetap sedangkan kaki C adalah kaki yang hambatannya dapat berubah dalam range hambatan kaki utama (kaki A dan B). besarnya hambatan kaki C dapat diukur relatif terhadap kaki A atau kaki B. misalnya hambatan di kaki utama adalah  $1K\Omega$  dan hambatan kaki C terhadap kaki A adalah  $250\Omega$ , maka besar nilai hambatan kaki C diukur relatif terhadap kaki B adalah  $750\Omega$ , yang merupakan hambatan diantara kaki C dan B yang sama dengan hasil pengurangan antara hambatan dikaki utama ( $1K\Omega$ ) dan hambatan antara kaki A dan C ( $250\Omega$ ).

## 2.5 Programmable Peripheral Interface (PPI) 8255

PPI 8255 adalah suatu media penghubung antarmuka antara komputer dengan dunia luar. Pada perancangan ini digunakan produksi dari Intel yang memiliki 3 buah port I/O yaitu port A, B, dan C.



Gambar 2.19 Blok Diagram PPI 8255 [6]

Ketiga port ini dibagi menjadi dua bagian yaitu bagian A yang terdiri atas port A dan port C *upper* (bit 4 sampai dengan 7) dan bagian B yang terdiri atas port B dan port C rendah (bit 0 sampai dengan bit 3). Di dalam PPI 8255 terdapat 8 bit penyangga bidirectional tiga keadaan (*Buffer 3-state*) untuk data yang terhubung ke CPU, yang berarti bahwa data dapat dikirim atau diterima oleh penyangga dari komputer, sedangkan keluaran masing-masing port memiliki penyangga (*buffer*) dan bersifat menahan (*latch*), sehingga data yang dikeluarkan ke *port* akan tetap keadaannya selama tidak diubah atau chip direset.

Operasi PPI 8255 yang berhubungan dengan pin-pin masukan dapat dilihat dalam table 2.1

Tabel 2.1 Mode Operasi PPI 8255

A1	A0	RD	WR	CS	Operasi yang terjadi
Operasi baca					
0	0	0	1	0	Port A ke bus data
0	1	0	1	0	Port B ke bus data
1	0	0	1	0	Port C ke bus data
Operasi tulis					
0	0	1	0	0	Bus data ke port A
0	1	1	0	0	Bus data ke port B
1	0	1	0	0	Bus data ke port C
1	1	1	0	0	Bus data ke control register

Terdapat 3 macam mode operasi PPI 8255 yang dapat diprogram ke masing-masing port, yaitu ;

- Mode 0 : *Basic I/O*

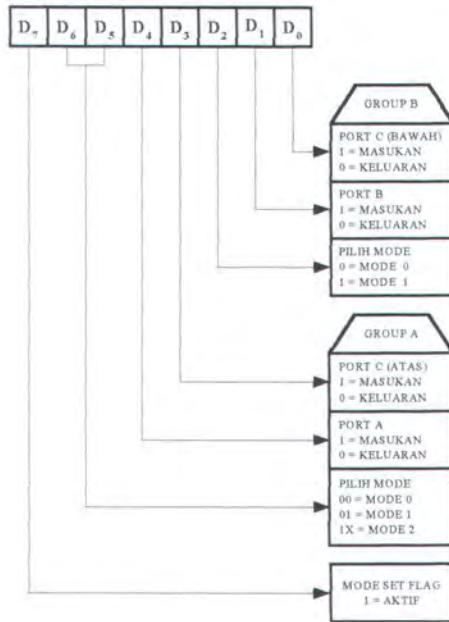
Pada mode 0, port A, B dan C dapat dioperasikan sebagai port masukan atau port keluaran. Untuk port C pada mode ini dibagi menjadi 2 saluran keluaran atau masukan, yang masing-masing saluran terdiri atas 4 bit. Saluran pertama pin PC<sub>0</sub>-PC<sub>3</sub> dan saluran kedua pin PC<sub>4</sub> - PC<sub>7</sub>.

- Mode 1 : *Strobe I/O*

Pada mode 1 port A dan port B bisa digunakan sebagai port masukan atau keluaran. Sedangkan port C berfungsi sebagai sinyal kontrol perpindahan data dari atau menuju port A dan port B.

- Mode 2 : Bidirectional Bus

Sedangkan untuk mode 2, port yang dapat digunakan untuk port masukan atau keluaran hanya port A, dan port C digunakan sebagai sinyal kontrol perpindahan data dari atau menuju port A.



Gambar 2.20 Register Control Word PPI 8255 [6]

Pemilihan mode tersebut dilakukan dengan cara mengisikan suatu *control word* pada *control word register*. Dalam *control word* tersebut berisi informasi tentang mode operasi yang diprogramkan untuk suatu port. Bila diprogram untuk operasi mode 1 atau mode 2, maka ada dua port yang bekerja sama untuk melaksanakan penerimaan dan pengeluaran data dan memberikan sinyal untuk kontrol. Dalam perencanaan perangkat keras dalam tugas akhir ini hanya membutuhkan port masukan dan keluaran tanpa sinyal kontrol, sehingga PPI 8255 dioperasikan pada mode 0.

Dalam mode 0, ketiga port PPI 8255 hanya berfungsi sebagai port masukan atau keluaran biasa, artinya data dapat diambil atau diletakkan dari dan

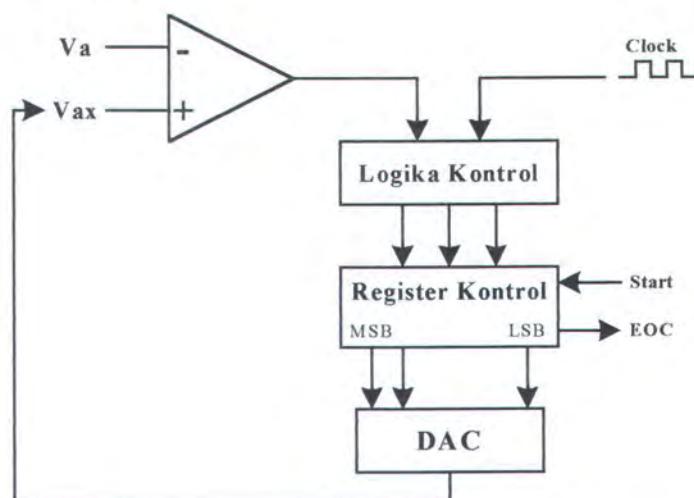
ke port secara langsung dengan instruksi IN/OUT ke port 8255 tanpa ada sinyal lain yang mengatur proses pengambilan atau peletakan data.

Semua port yang bekerja sebagai port keluaran akan bersifat menahan (*latch*), artinya data yang sudah diletakkan ke port keluaran masih tetap selama tidak diubah oleh data lain yang ditulis ke port yang sama atau akibat adanya sinyal RESET.

## 2.6 Analog To Digital Converter (ADC)

ADC merupakan pengkonversi sinyal analog ke digital. Diperlukan karena sinyal-sinyal listrik yang bekerja pada proses adalah sinyal analog sedangkan mikroprosesor bekerja dengan sinyal digital. Informasi digital adalah dalam bentuk bilangan biner. Pada saat digunakan pada komputer, bilangan biner ini disebut *binary word*.

ADC yang digunakan dalam perancangan tugas akhir ini adalah produk dari *National Semiconductor* dengan type ADC 0804 8 bit. Bekerja dengan cara *succesive approximation*.



Gambar 2.21 Blok diagram SAC ADC<sup>[6]</sup>

Blok diagram SAC ADC diperlihatkan pada gambar 2.21. Adapun cara kerjanya dapat diuraikan sebagai berikut. Sebuah SAC ADC tidak menggunakan pencacah (counter), tetapi menggunakan register kontrol (juga disebut Successive Approximation Register, SAR) yang isinya dapat diubah bit per bit oleh suatu logika kontrol.

Proses konversi dimulai dengan memberikan pulsa start. Akibat pulsa start ini, logika kontrol akan mereset semua bit dalam register kontrol, sehingga keluaran register semuanya sama dengan nol dan Vax sama dengan 0 volt. Karena itu  $V_{ax} < V_a$ , maka keluaran komparator akan tinggi (logika 1). Karena logika 1 ini, logika kontrol akan menset MSB (Most Significant Bit atau bit paling akhir) register kontrol menjadi logika 1. Dengan demikian sekarang Vax akan sama dengan bobot MSB dikalikan step size DAC. Kalau sekarang ternyata  $V_{ax} > V_a$ , maka keluaran komparator akan berubah menjadi logika 0. Akibatnya logika kontrol akan mereset MSB tadi menjadi logika 0. Kemudian logika kontrol akan menset bit berikutnya (MSB kedua) menjadi logika 1.

Sebaliknya jika  $V_{ax} \leq V_a$ , keluaran komparator akan berlogika 1, sehingga logika kontrol akan tetapi memberikan MSB berlogika 1. Logika kontrol kemudian akan menset bit berikutnya (MSB kedua) menjadi berlogika 1. Proses diatas diulang sampai semua bit dicoba, mulai dari MSB, kemudia MSB kedua dan seterusnya sampai LSB. Setelah LSB dicoba, maka proses konversi selesai dan logika kontrol akan mengeluarkan sinyal EOC (End Of Conversion). Setelah konversi selesai, register kontrol berisi bilangan biner yang ekivalen dengan sinyal analog  $V_a$ .

### 2.6.1 ADC 0804

IC ADC 0804 mempunyai 2 masukan analog, yaitu Vin(+) dan Vin(-), sehingga dapat menerima masukan deferensial (tegangan selisih). Jadi masukan analog sesungguhnya yaitu Vin, akan sama dengan selisih antara tegangan-tegangan yang dihubungkan kedua pena masukan ini :  $Vin = Vin(+) - Vin(-)$ . Jika masukan analog berupa tegangan tunggal, maka tegangan ini harus dihubungkan ke Vin(+), dan Vin(-) harus ke ground analog. Untuk operasi normal, ADC 0804 menggunakan Vcc (tegangan masukan) sebesar +5Volt sebagai tegangan referensi, dalam hal ini jangkauan masukan analog mulai dari 0 volt sampai +5 volt (skala penuh). Karena IC ini adalah ADC 8 bit, maka ukuran step adalah 19,6mV dan persentase resolusi 0,39 %.

ADC 0804 memiliki generator clock internal yang harus diaktifkan dengan menghubungkan sebuah resistor eksternal (R) antara pena CLK OUT dan CLK IN serta sebuah kapasitor eksternal (C) antara CLK IN dan ground digital.

Maka frekuensi clock yang digunakan akan sama dengan :

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

Sedangkan fungsi-fungsi pena lainnya akan dijelaskan secara singkat sebagai berikut :

- Masukan CS (Chip Select) aktif rendah.

Digunakan untuk mengaktifkan ADC 0804 jika CS berlogika 0 dan jika CS berlogika 1 ADC 0804 tidak aktif.

- Masukan RD (Read atau Output Enable).

Digunakan untuk mengaktifkan keluaran digital ADC 0804. Jika CS = 0, maka kondisi logika di pena D0 – D7 akan mewakili hasil konversi A/D terakhir dan dapat dibaca oleh microprosesor atau komputer.

- Masukan WR (Write atau Start Conversion).

Digunakan untuk memulai proses konversi. Untuk itu WR harus diberi pulsa logika 0.

- Keluaran INTR (Interrupt atau End of Conversion).

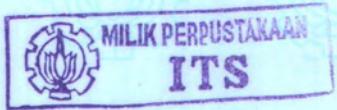
Adalah pena yang berfungsi untuk memberikan tanda bahwa konversi telah selesai. Pada saat konversi dimulai INTR akan berubah berlogika 1 dan diakhiri konversi INTR kembali ke logika 0.

- Masukan Vref/2.

Digunakan untuk mengurangi tegangan referensi internal, yang berarti mengubah jangkauan analog yang dapat ditangani oleh ADC 0804. Jika masukan Vref/2 tidak dihubungkan, maka jangkauan masukan analog adalah 0 volt sampai +5 Volt. Jika dihubungkan ke suatu tegangan eksternal sebesar Vx volt, maka jangkauan masukan analog akan menjadi  $0 - (2Vx)$  volt. Contoh jika Vref dihubungkan ke tegangan +2 Volt, maka jangkauan masukan analog adalah 0 volt sampai +4 Volt.

Secara umum algoritma pengoperasian ADC 0804 adalah sebagai berikut :

- Memberikan pulsa logika 0 ke CS.
- Memberikan pulsa logika 0 ke WR.



### **BAB III**

## **PERANCANGAN DAN PEMBUATAN ALAT**

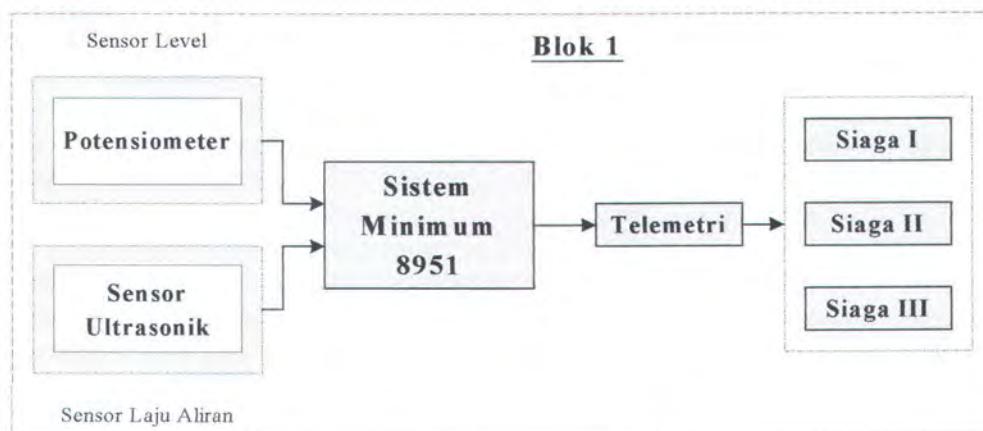
## **BAB III**

### **PERANCANGAN DAN PEMBUATAN ALAT**

Setelah dibahas mengenai teori-teori yang mendasari perancangan dalam tugas akhir ini, maka dalam bab ini akan dijelaskan cara kerja rangkaian dan bentuk perancangan beserta implementasi dari sistem.

#### **3.1 Cara Kerja Rangkaian**

Pada pembuatan tugas akhir dengan judul rancang bangun sistem pengendalian level dan laju aliran pada sungai (aksi kendali pada pintu air bendungan) untuk mendeteksi banjir berbasis mikrokontroller, merupakan kelanjutan tugas akhir seperti blok diagram pada gambar 3.1. Yang terdiri dari serangkaian alat yang mempunyai fungsinya masing-masing, seperti yang terlihat dibawah ini.



Gambar 3. 1 Diagram blok 1<sup>[4]</sup>

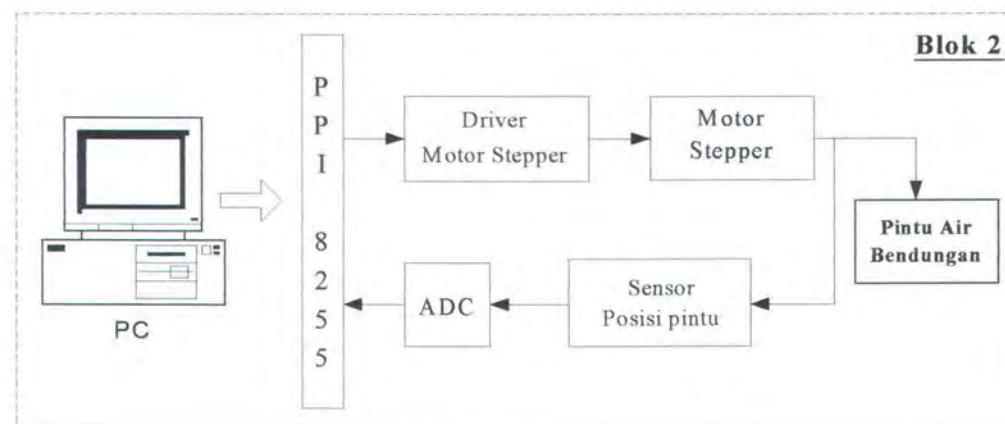
Dalam tugas akhir ini memakai dua deteksi banjir, yaitu variabel ketinggian air (level) dan laju aliran. Untuk sensor level air memakai metode

pelampung dan pemberat. Naik turunnya ketinggian air ini sebanding dengan naik turunnya resistansi pada sumbu tengah (poros) pulley pada pelampung, sehingga membuat tegangan analog inputan pada ADC berfluktuasi. Semakin tinggi tegangan pada  $V_{in}$  ADC ini sebanding dengan ketinggian air pada miniatur sungai.

Untuk sensor laju aliran memakai transduser Ultrasonik yang memanfaatkan metode efek doppler dimana frekuensi suara dari benda-benda bergerak yang kita dengar selalu lebih tinggi dari frekuensi sebenarnya bila arah geraknya mendekati kita. Sebaliknya bila arah geraknya menjauhi kita maka frekuensi suaranya terdengar lebih rendah.

Mikrokontroller merupakan pengolah data inputan dari sensor ketinggian air dan laju aliran. Dari data akuisisi kedua sensor ini kemudian ditelemetrikai ke komputer untuk diproses sebagai referensi aksi kendali buka atau tutup pintu air pada sungai.

Dari diagram blok 1 dan cara kerja rangkaian diatas, kemudian dilanjutkan diagram blok 2 seperti dibawah ini :



Gambar 3.2 Diagram blok 2

Pada gambar diatas terlihat bahwa komputer mendapat masukan yang berasal dari mikrokontroller melalui komunikasi serial RS-232, dengan inisialisasi ketinggian air dan laju aliran pada miniplant bendungan.

Antarmuka PPI 8255 digunakan sebagai jalur informasi antara sistem luar dengan PC, keluaran dari komputer dapat menggerakkan motor stepper dengan perantara driver motor stepper. Kemudian dengan bergeraknya motor stepper tersebut akan menggerakkan pintu air membuka atau menutup yang dihubungkan dengan roda gerigi. Aksi membuka atau menutup dari pintu air akan menggerakkan potensiometer yang nantinya akan mengetahui berapa bukaan pintu air tersebut. Data yang dihasilkan sensor potensiometer pintu bendungan adalah data analog, supaya data tersebut dapat diolah komputer maka data tersebut harus dikonversi ke bentuk digital. Untuk mengubah data analog ke data digital digunakan ADC 0804.

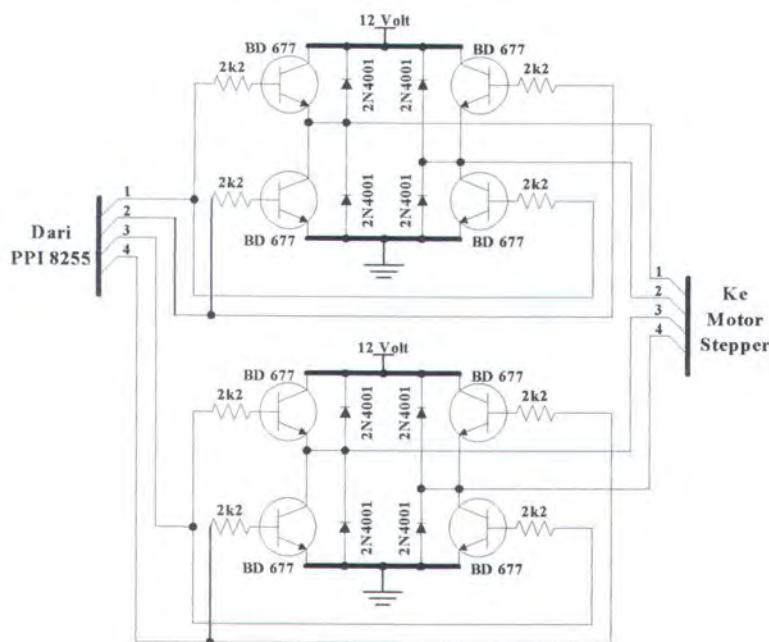
### **3.2 Perancangan Perangkat Keras**

Diagram blok rangkaian dari pengendali diatas terdiri dari beberapa rangkaian, yaitu rangkaian driver motor stepper, rangkaian ADC 0804, inisialisasi PPI 8255, rangkaian potensiometer. Karena antara rangkaian satu dengan rangkaian yang lainnya sangat berhubungan. Rangkaian tersebut akan dijelaskan satu per satu.

### 3.2.1 Driver Motor Stepper

Dalam perancangan rangkaian driver motor stepper ini menggunakan 8 transistor yaitu BD677. Rangkaian driver motor stepper seperti pada gambar rangkaian dibawah ini.

Motor stepper yang digunakan bertipe bipolar mempunyai karakteristik, yaitu memerlukan catu daya sebesar 12 volt dan  $1,8^\circ/\text{step}$



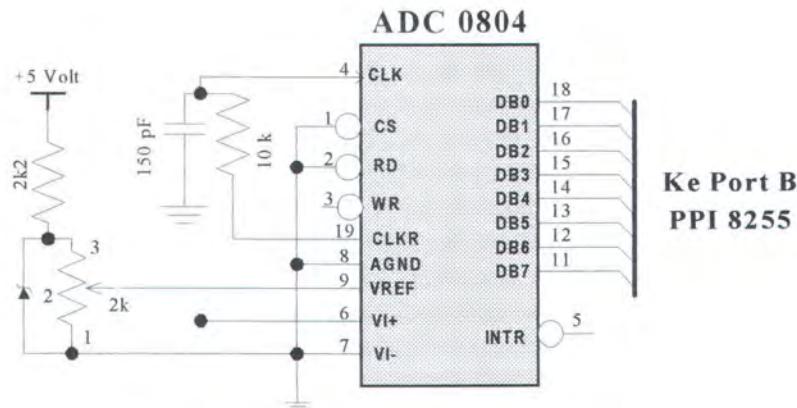
Gambar 3.3 Rangkaian driver motor stepper

### 3.2.2 Konverter Analog ke Digital

Data yang dihasilkan sensor posisi pintu bendungan adalah data analog, supaya data tersebut dapat diolah komputer maka data tersebut harus dikonversi ke bentuk digital. Untuk mengubah data analog ke data digital digunakan ADC, dalam tugas akhir ini digunakan ADC 0804 dari National Semiconductor.

ADC tipe ini memakai metode pendekatan berturut-turut (Successive Approximation) untuk mengkonversi masukan analog menjadi data digital 8 bit.

Berikut ini akan digambarkan hubungan pin-pin IC ADC 0804 yang dihubungkan dalam pemenuhan kebutuhan sinyal digital untuk masukan pada PPI 8255.



Gambar 3.4 Rangkaian ADC 0804

IC ini memiliki generator clock internal yang harus diaktifkan dengan menghubungkan sebuah resistor eksternal (R) antara pena CLK OUT dan CLK IN serta sebuah kapasitor eksternal (C) antara CLK IN dan ground digital. Maka frekuensi clock yang digunakan akan sama dengan :

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

dengan menggunakan  $R = 10 \text{ K}\Omega$  serta  $C = 150 \text{ pF}$  maka dapat diperoleh frekuensi sebesar :

$$f = \frac{1}{1,1 \times 10\text{K}\Omega \times 150\text{pF}}$$

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

Kaki 3  $\overline{\text{WR}}$  dan kaki 5  $\overline{\text{INTR}}$  digabung untuk menyamakan waktu konversi dan baca dari ADC dengan memberi logika 0. untuk menghubungkan kaki 3 dan 5 pada ground digunakan relay yang dikendalikan dari port C5 PPI 8255.

### 3.2.3 Peralatan Antar Muka PPI 8255

Antarmuka PPI 8255 digunakan sebagai jalur informasi antara sistem luar dengan PC. Pada komputer akan dipilih alamat 0380H-0383H, karena alamat tersebut tidak digunakan oleh I/O lain. Port PPI yang digunakan adalah port A, port B dan port C. Cara pengesetan yang demikian ini adalah dengan memasukkan control word pada PPI. Dalam hal ini yang diinginkan adalah :

- Port A : PA<sub>0</sub> – PA<sub>3</sub> digunakan sebagai keluaran untuk mengendalikan motor stepper I. PA<sub>4</sub> – PA<sub>7</sub> digunakan sebagai keluaran untuk mengendalikan motor stepper II.
- Port B : PB<sub>0</sub> – PB<sub>7</sub> digunakan untuk masukan digital 8 bit dari ADC 0804.
- Port C Upper : PC<sub>4</sub> digunakan sebagai keluaran untuk mengendalikan select multiplexing potensiometer. PC<sub>5</sub> sebagai keluaran untuk mengendalikan relay switch aktif low pada kaki 3 dan 5 ADC.

Maka control word yang digunakan adalah,

Tabel 3.1 Control Word PPI 8255

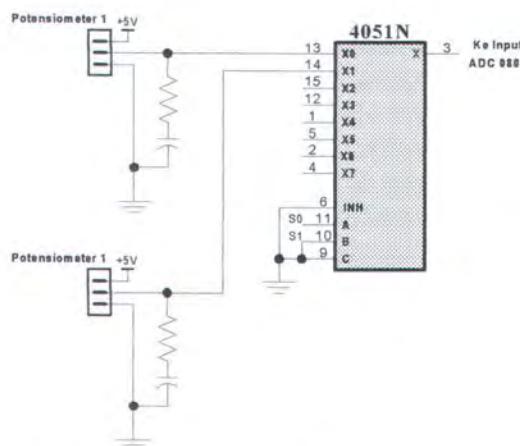
D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	0	0	1	1
8H				2H			



### 3.2.4 Potensiometer

Guna mendekripsi posisi gerakan pintu air maka dibutuhkan sensor posisi.

Sensor posisi yang digunakan adalah potensiometer jenis kawat  $50\text{K}\Omega$ .



Gambar 3.5 Rangkaian potensiometer

Potensiometer dipasang disetiap pintu dan penempatannya dipasang di atas motor stepper dikopel dengan ulir. Kopel yang digunakan adalah dari karet untuk memudahkan dalam melakukan gerakan memutar pada satu poros yang lurus.

Naik turunnya pintu air ini sebanding dengan naik turunnya resistansi, sehingga membuat tegangan analog inputan pada ADC *berfluktuasi*. Semakin tinggi tegangan pada  $V_{in}$  ADC ini sebanding dengan naik turunnya pintu air pada miniplant bendungan.

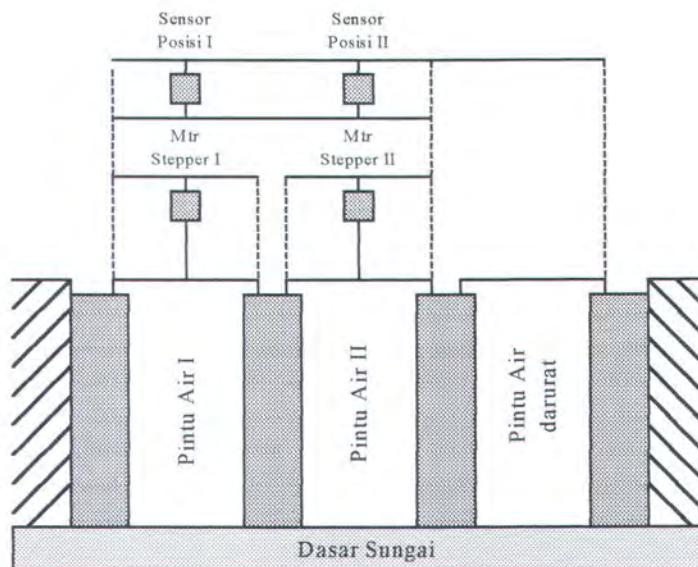
Pena  $S_0$  IC 4051 dihubungkan dengan pin C4 PPI 8255. Keluaran multiplexer dihubungkan ke ADC 0804 agar dapat dibaca oleh komputer dan dapat ditampilkan di PC.

### 3.3 Perancangan Sistem Mekanik

Pintu air yang direncanakan disini adalah pintu air yang memiliki gerakan naik dan turun sesuai arah yang diinginkan. Sebagai penggerak digunakan motor stepper.

Motor stepper dikopel dengan ulir, dan ulir inilah yang dihubungkan dengan pintu air. Kopel yang digunakan adalah dari tali bergerigi untuk memudahkan dalam melakukan gerakan memutar pada satu poros yang lurus. Keuntungan lainnya kopel ini memberikan toleransi terhadap gangguan yang ditimbulkan sebagai akibat kurang lurusnya poros. Jumlah pintu yang harus dikendalikan disini adalah dua buah bentuk dan ukuran masing-masing dibuat sama dan satu pintu air darurat untuk pelimpahan (spillway).

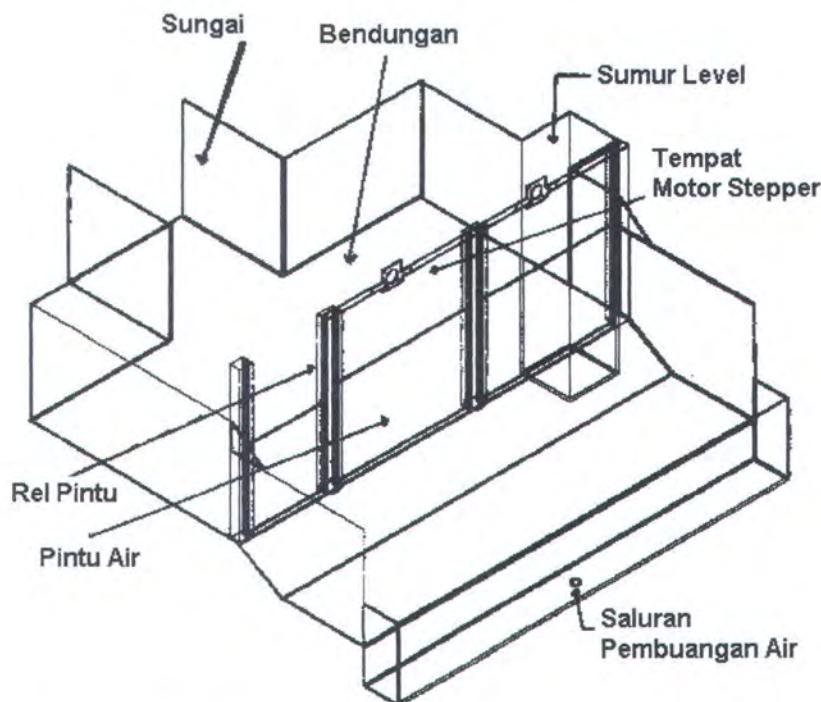
Penempatan sensor posisi pintu diletakkan diatas motor stepper dikopel dengan ulir. Kopel yang digunakan adalah dari karet untuk memudahkan dalam melakukan gerakan memutar.



Gambar 3.6 Bentuk Sistem mekanik bendungan (tampak dari depan).

### 3.4 Perancangan Bentuk Miniplant Bendungan

Bentuk miniplant bendungan pada tugas akhir ini dirancang untuk mendekati representasi dari plant bendungan sebenarnya. Seperti yang terlihat pada gambar 3.7



Gambar 3.7 Bentuk miniplant Bendungan dan Pintu air

Bahan yang digunakan dalam pembuatan bendungan terbuat dari mika dengan tebal 3 mm dan tiang (rel pintu) pintu air terbuat dari bahan aluminium.

## **BAB IV**

### **PENGUJIAN DAN PENGUKURAN**

## BAB IV

### PENGUJIAN DAN PENGUKURAN

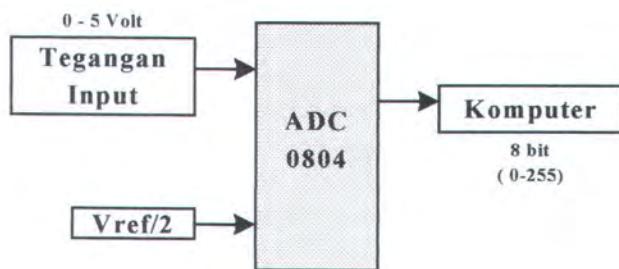
Pada bab ini dilakukan pengujian dan pengukuran dari miniplant yang dibuat. Sehingga diperoleh keterangan tentang kelebihan dan kekurangan dari sistem yang dirancang. Pengujian meliputi pengujian dan pengukuran masing-masing modul, pengujian sistem.

#### 4.1 Pengujian Perangkat Keras

Seperti yang telah disebut dalam bab sebelumnya, bahwa pengujian dan pengukuran terdiri atas rangkaian ADC 0804, dan peralatan antarmuka dengan komputer (PPI 8255).

##### 4.1.1 Pengujian ADC

Pengujian ADC dilakukan dengan memberikan sinyal inputan pada ADC berupa tegangan analog DC mulai 0 volt sampai dengan 5 volt, kemudian dilakukan pengamatan hasil konversi dengan bantuan perangkat lunak untuk mendapatkan informasi yang diperlukan. Blok pengujian dapat dilihat dibawah ini.



4.1 Pengujian blok diagram ADC 0804

Untuk pin  $V_{ref/2}$  diset tegangan 2,50 volt sehingga nilai Vref sama dengan 5,00 volt, maka perubahan sebesar  $\frac{1}{2}^8 \times 5,00$  volt = 19,53125 mV menyebabkan perubahan 1 bit pada keluaran ADC.

Selanjutnya hasil pengukuran dan pengkonversian dari perangkat lunak dibandingkan dengan harga sebenarnya (sinyal inputan). Data diambil 10 sempel dengan perubahan kenaikan sinyal tegangan inputan sebesar 0,5 volt.

Tabel 4.1 Hasil pengukuran ADC

No	Tegangan Input (Volt)	Hasil Teoritis	Hasil Pembacaan Komputer
1	0,5	25	25
2	1,0	50	50
3	1,5	75	76
4	2,0	100	100
5	2,5	125	125
6	3,0	150	149
7	3,5	175	175
8	4,0	200	200
9	4,5	225	225
10	5,0	250	249

Dari hasil yang diperoleh maka hasil konversi ADC masih terjadi perbedaan nilai, sesuai dengan data sheet dari ADC bahwa maksimum kesalahan pengkonversian yang diijinkan adalah  $\pm 1$  LSB, maka ADC yang dirancang mempunyai kesalahan konversi yang masih bisa ditoleransi.

#### 4.1.2 Pengujian PPI 8255

Pengujian PPI 8255 dilakukan untuk mengecek hubungan antar port yang digunakan sebagai antarmuka, yaitu pembacaan dilakukan port B dan pengiriman oleh port A dan port C. Disini juga diperlukan ADC sebagai sarana pembantu pengiriman sinyal pada PPI 8255 yang hanya dapat membaca sinyal digital. Untuk inputan tegangan diberikan sinyal tegangan analog 0-5 volt DC yang terlebih dulu

dikonversi menjadi data digital oleh ADC yang mana dalam bentuk biner 8 bit akan tampak dengan bantuan led kemudian baru dikirimkan menuju PPI 8255 melalui port B yang kemudian dengan perangkat lunak dilakukan pembacaan pada port B berapa nilai yang terukur dalam bentuk desimal (0-255).

Sedangkan untuk pengujian pengiriman data, pada perangkat lunak dengan memberikan data desimal yang akan dikirim keluaran PPI melalui port A dan port C dan nantinya penunjukan nilai terukur dapat diketahui melalui nyala led, dimana nyala led menunjukkan nilai biner dari tegangan terukur yang nantinya apabila dikonversikan dalam data desimal nilainya haruslah sama dengan data desimal yang terkirim, data desimal yang dikirimkan komputer melalui port A dan port C dengan penyalaan pada led artinya pengkonversian dari data biner ke data desimal ataupun sebaliknya adalah sama.

Tabel 4.2 Hasil pengujian PPI 8255

- Pengujian pembacaan port B

- Pengujian pengiriman port A dan port C

No	Data Desimal di Program	Nilai biner pada bit ke							
		7	6	5	4	3	2	1	0
1	0	0	0	0	0	0	0	0	0
2	15	0	0	0	0	1	1	1	1
3	30	0	0	0	1	1	1	1	0
4	45	0	0	1	0	1	1	0	1
5	60	0	0	1	1	1	1	0	0
6	75	0	1	0	0	1	0	1	1
7	90	0	1	0	1	1	0	1	0
8	105	0	1	1	0	1	0	0	1
9	120	0	1	1	1	1	0	0	0
10	135	1	0	0	0	0	1	1	1
11	150	1	0	0	1	0	1	1	0
12	165	1	0	1	0	0	1	0	1
13	180	1	0	1	1	0	1	0	1
14	195	1	1	0	0	0	0	1	1
15	210	1	1	0	1	0	0	1	0
16	225	1	1	1	0	0	0	0	1
17	240	1	1	1	1	0	0	0	0
18	255	1	1	1	1	1	1	1	1

#### 4.2 Pengujian Perangkat Lunak

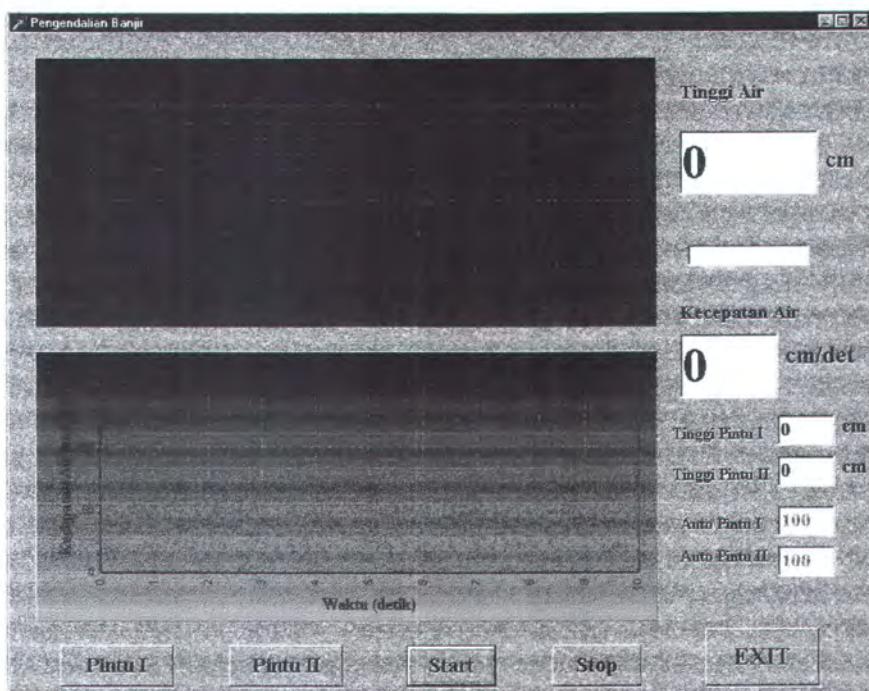
Pengoperasian miniplant menggunakan komputer sebagai pengontrol sedangkan spesifikasi yang digunakan prosesor Intel Pentium III 600 MHz, memori 128 MB dengan bahasa pemrograman yang digunakan adalah Borland Delphi 5.0. Pemilihan compiler Delphi dikarenakan bahasa pemrogramannya yang lebih mudah dicerna daripada bahasa tingkat rendah yang lebih dekat ke bahasa mesin.



#### 4.2.1 Tampilan Menu Utama

Program kontrol yang dibuat mempunyai tampilan visualisasi. Dengan menggunakan PPI 8255 sebagai antarmuka komputer dengan sistem akuisisi data, program ini melakukan pembacaan tinggi air dan kecepatan air yang terkirim dari sensor level dan sensor ultrasonic melalui komunikasi data serial RS-232.

Pada tampilan menu utama juga ditampilkan grafik tinggi air dan kecepatan air untuk tiap detik perubahan. Keterangan tiap fasilitas program adalah:



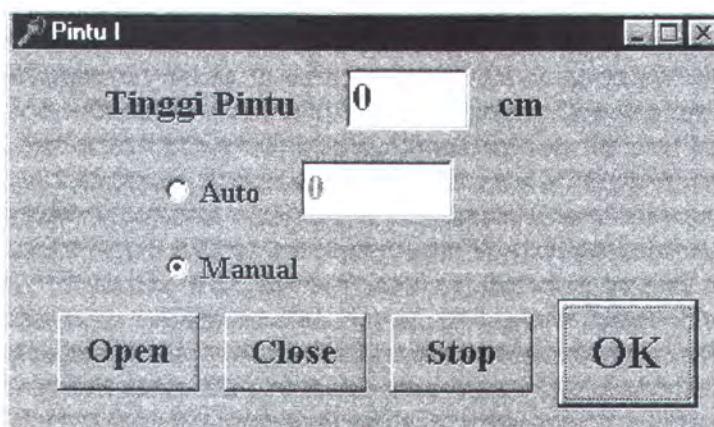
Gambar 4.2 Tampilan program.

- **Tinggi Air**, tampilan pembacaan ketinggian air dari sensor level.
- **Kecepatan Air**, tampilan pembacaan kecepatan air dari sensor ultrasonic.
- **Tinggi Pintu I**, tampilan pembacaan ketinggian pintu I setelah komputer memerintahkan kepada motor stepper untuk membuka atau menutup.

- **Tinggi Pintu II**, tampilan pembacaan ketinggian pintu II setelah komputer memerintahkan kepada motor stepper untuk membuka atau menutup.
- **Pintu I**, merupakan setting dari pintu I untuk membuka atau menutup.
- **Pintu II**, merupakan setting dari pintu II untuk membuka atau menutup.
- **Start**, perintah awal untuk melakukan proses.
- **Stop**, perintah selesai untuk melakukan proses.
- **Exit**, perintah keluar dari program pengendalian banjir.

#### 4.2.2 Tampilan Menu Setting Pintu

Pada tampilan menu utama program disediakan fasilitas setting pintu air untuk melakukan buka atau tutup pintu air dan tampilan setting pintu seperti gambar 4.3. Keterangan dari tiap-tiap fasilitas setting pintu adalah:



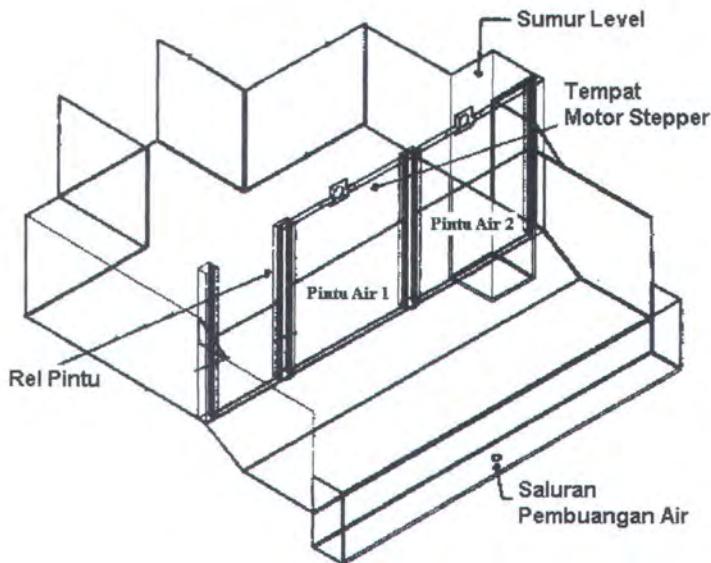
Gambar 4.3 Tampilan menu setting pintu

- **Tinggi pintu**, tampilan pembacaan ketinggian pintu dari sensor posisi pintu pada miniplant.
- **Auto**, perintah komputer untuk melakukan buka atau tutup pintu.
- **Manual**, perintah secara manual untuk membuka atau menutup pintu.

- **Open**, perintah untuk membuka pintu air per step.
- **Close**, perintah untuk menutup pintu air per step.
- **Stop**, perintah selesai dari proses.
- **Ok**, perintah untuk keluar dari tampilan setting pintu, menuju ke tampilan menu utama.

#### 4.3 Pengukuran Data Pintu Air

Pengambilan data posisi dari sensor posisi dilakukan dengan menggerakkan pintu ke posisi yang diinginkan. Dari hasil yang terbaca dibandingkan dengan pengukuran posisi secara pembacaan. Dalam proses ini posisi awal pintu pada titik 0 digerakkan menuju posisi yang diinginkan.



Gambar 4.4 Blok diagram pengujian pintu air I dan pintu air II

Pengambilan data dilakukan pada pintu air I dan pintu air II

#### Untuk pintu air I

Tabel 4.3 Data pintu air I

Ketinggian pintu air	0,2	0,4	0,7	0,9	1,2	1,4	1,7	1,9	2,1	2,4
No	Pembacaan di komputer									
	1	2	3	4	5	6	7	8	9	10
1	0,20	0,40	0,77	0,93	1,17	1,40	1,63	1,93	2,07	2,37
2	0,23	0,47	0,73	0,93	1,20	1,40	1,63	1,90	2,13	2,40
3	0,23	0,43	0,73	0,90	1,20	1,47	1,67	1,90	2,13	2,40
4	0,27	0,50	0,70	0,97	1,20	1,40	1,65	1,90	2,10	2,40
5	0,23	0,45	0,73	0,93	1,20	1,43	1,63	1,93	2,13	2,40
	<b>0,232</b>	<b>0,45</b>	<b>0,732</b>	<b>0,932</b>	<b>1,194</b>	<b>1,42</b>	<b>1,642</b>	<b>1,912</b>	<b>2,112</b>	<b>2,394</b>

#### Untuk pintu air II

Tabel 4.4 Data pintu air II

Ketinggian pintu air	0,2	0,4	0,7	0,9	1,2	1,4	1,7	1,9	2,1	2,4
No	Pembacaan di komputer									
	1	2	3	4	5	6	7	8	9	10
1	0,23	0,53	0,80	1,07	1,30	1,57	1,80	2,03	2,27	2,50
2	0,23	0,50	0,77	1,03	1,33	1,50	1,77	2,00	2,27	2,50
3	0,23	0,50	0,80	1,03	1,33	1,57	1,80	2,00	2,27	2,50
4	0,20	0,50	0,77	1,03	1,30	1,50	1,77	2,00	2,27	2,50
5	0,23	0,50	0,77	1,03	1,33	1,57	1,77	2,03	2,27	2,53
	<b>0,224</b>	<b>0,506</b>	<b>0,782</b>	<b>1,038</b>	<b>1,318</b>	<b>1,542</b>	<b>1,782</b>	<b>2,012</b>	<b>2,29</b>	<b>2,506</b>

#### **4.4 Analisa Data Pengukuran Ketinggian Pintu Air**

Pengambilan data hasil pengukuran ketinggian pintu air dapat dijadikan sebagai data untuk validitas alat dengan cara membandingkan dengan alat standart. Maka dilakukan pengukuran serta pengambilan data ketinggian pintu air dengan alat standart penggaris dan dibandingkan dengan keluaran data pada komputer melalui sensor posisi.

Dari data-data, seperti yang ditunjukkan pada tabel 4.3 dan tabel 4.4 maka akan dilakukan perhitungan-perhitungan untuk mendapatkan data-data kalibrasi *Type-A* dari alat yang dibuat, perhitungan tersebut meliputi :

- Nilai Rata-rata (*Mean*) :

$$\bar{X} = \frac{\sum x_i}{n}$$

$$\bar{X} = \frac{(x_1 + x_2 + x_3 + \dots + x_n)}{n}$$

- Derajat Kebebasan (*Degree of Freedom*) :

$$V = n - 1$$

- Standar Deviasi (*Standard Deviation*) :

Standar Deviasi (*Experimental Standard Deviation*) untuk sejumlah data terbatas adalah sebagai berikut :

$$S(X_i) = \sigma = \sqrt{\varphi} = \sqrt{\frac{\sum (x_i - \bar{x})^2}{n-1}}$$

- Ketidakpastian Standar (*Standard Uncertainty*) :

$$S(\bar{X}) = ESMD = \frac{S(X_i)}{\sqrt{n}}$$

$$U(X_i) = S(\bar{X})$$

$$U(X_i) = \frac{S(X_i)}{\sqrt{n}}$$

- Hasil Pengukuran :

$$X = \bar{x} \pm U(X_i)$$

Setelah mengetahui rumusan tersebut maka data-data yang telah diperoleh, selanjutnya disubtitusikan ke persamaan-persamaan tersebut diatas. Hasil dari perhitungan diperlihatkan seperti dibawah ini :

Hasil Pengukuran data :

$$X = \bar{x} \pm S(X_i)$$

Tabel 4.5 Data hasil perhitungan pada alat buatan untuk pintu I.

No	Ketinggian pintu air	$\bar{X}$ (m/s)	S(xi) (m/s)	U(xi) (m/s)	U(R) (%)	Presisi (%)	Akurasi (%)	X,alat (m/s)
1	0,2	0,232	0,025	0,011	4,74	98,9	95,26	$0,23 \pm 0,03$
2	0,4	0,450	0,038	0,017	3,78	98,3	96,22	$0,45 \pm 0,04$
3	0,7	0,732	0,025	0,011	1,50	98,9	98,50	$0,73 \pm 0,03$
4	0,9	0,932	0,025	0,011	1,18	98,9	98,82	$0,93 \pm 0,03$
5	1,2	1,194	0,013	0,006	0,01	99,4	99,99	$1,19 \pm 0,01$
6	1,4	1,420	0,031	0,014	0,99	98,6	99,01	$1,42 \pm 0,03$
7	1,7	1,642	0,018	0,008	0,49	99,2	99,51	$1,64 \pm 0,02$
8	1,9	1,912	0,016	0,007	0,37	99,3	99,63	$1,91 \pm 0,02$
9	2,1	2,112	0,027	0,012	0,57	98,8	99,43	$2,11 \pm 0,03$
10	2,4	2,394	0,013	0,006	0,25	99,4	99,75	$2,39 \pm 0,01$

Tabel 4.6 Data hasil perhitungan pada alat buatan untuk pintu II.

No	Ketinggian pintu air	$\bar{X}$ (m/s)	S(xi) (m/s)	U(xi) (m/s)	U(R) (%)	Presisi (%)	Akurasi (%)	X,alat (m/s)
1	0,2	0,224	0,013	0,006	2,68	99,4	97,32	$0,22 \pm 0,01$
2	0,4	0,506	0,013	0,006	0,12	99,4	97,20	$0,51 \pm 0,01$
3	0,7	0,782	0,016	0,007	0,89	99,3	99,11	$0,78 \pm 0,02$
4	0,9	1,038	0,018	0,008	0,77	99,2	99,23	$1,04 \pm 0,02$
5	1,2	1,318	0,016	0,007	0,53	99,3	99,47	$1,32 \pm 0,02$
6	1,4	1,542	0,038	0,017	1,10	98,3	98,90	$1,54 \pm 0,04$
7	1,7	1,782	0,016	0,007	0,39	99,3	99,61	$1,79 \pm 0,02$
8	1,9	2,012	0,016	0,007	0,35	99,3	99,65	$2,01 \pm 0,02$
9	2,1	2,290	0,045	0,020	0,87	98	99,13	$2,29 \pm 0,05$
10	2,4	2,506	0,013	0,006	0,24	99,4	99,76	$2,51 \pm 0,01$

Adanya pembulatan pada hasil pengukuran disebabkan karena pada alat yang dibuat memiliki kemampuan menampilkan dua angka dibelakang koma maka pada hasil pengukuran juga dibuat sesuai dengan kemampuan alat. Ketidakpastian mutlak ( $U[X_i]$ ) mempunyai hubungan dengan tingkat ketepatan/keakurasaan yaitu, yang berarti makin kecil harga ( $U[X_i]$ ) yang tercapai, makin tepat/akurat suatu pengukuran tersebut. Begitu pula hubungan antara ketidakpastian relatif ( $U_R$ ) dengan ketelitian/kepresision, yaitu makin kecil ketidakpastian relatif maka semakin tinggi tingkat ketelitian/kepresision pengukuran tersebut.

Perbedaan hasil pengukuran tersebut timbul karena disebabkan oleh sistem mekanik yang dibuat, memerlukan kopel yang sangat presisi antara motor stepper dengan sensor posisi atau sensor potensiometer.

#### **4.4.1 Analisa Penentuan Ketidakpastian Standar ( $S_q$ ) Ketinggian Pintu**

Ketidakpastian standar diperoleh dengan cara memakai rumus :

$$S_{(q)} = \sqrt{\frac{\sum SSR^2}{n - N}}$$

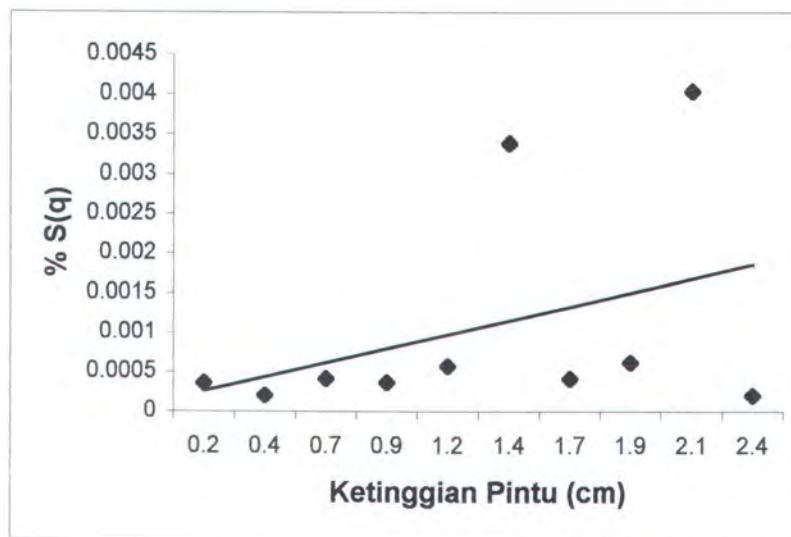
Dimana :

$$SSR = (\text{Residual})^2$$

$$N = 2 \text{ untuk regresi linear}$$

$$\text{Res} = Y_i - (a + b x_i); \text{ dimana nilai } a \text{ dan } b \text{ diperoleh dari regresi linear}$$

Sehingga didapat grafik hubungan antara ketinggian pintu air dengan Ketidakpastian Standar



Gambar 4.6 Grafik ketinggian pintu air II vs Ketidakpastian Standar

Dari tabel 4.4. diolah dengan memakai Microsoft Excel diperoleh harga regresi dan residu sehingga diperoleh harga Ketidakpastian standar :

$$S_{(q)} = \sqrt{\frac{\sum SSR^2}{n - N}}$$

$$= \sqrt{\frac{0.00568}{5 - 2}}$$

$$= 0,044 \text{ cm}$$

Kondisi yang digambarkan pada grafik 4.6 lebih menunjukkan keadaan hanya untuk ketinggian pintu air 1.4 cm dan ketinggian pintu air 2.1 cm ketidakpastian menunjukkan nilai yang besar. Tapi untuk ketinggian pintu air yang lain relatif bagus.

## **BAB V**

## **KESIMPULAN**



MILIK PERPUSTAKAAN  
ITS

## **BAB V**

### **KESIMPULAN**

Setelah melalui tahap perencanaan dan pembuatan alat yang dilanjutkan dengan tahap pengujian dan analisa data, maka dapat disimpulkan beberapa hal antara lain yaitu :

- Penggunaan komputer sebagai pengontrol dalam pemrosesan data-data dari sensor untuk menggerakkan pintu air membuka atau menutup memiliki kelebihan yaitu meningkatkan ketelitian dalam proses kontrol dan memberikan hasil yang cukup handal dan aman dalam pengoperasiannya.
- Dari alat yang dirancang dan dibuat memiliki harga ketidakpastian standart 0,03 cm untuk pintu 1 dan ketidakpastian standart 0,044 cm untuk pintu 2.
- Pemakaian motor stepper mempunyai keuntungan, kopel akan meningkatkan seiring peningkatan beban.
- Sensor yang digunakan merupakan faktor utama yang menentukan unjuk kerja dari sistem yang dibuat.

## **DAFTAR PUSTAKA**

## DAFTAR PUSTAKA

1. Curtis D. J, *Process Control Instrumentation Technology*, International Edition, 1995.
2. Gunterus F, *Falsafah Dasar Sistem Pengendalian Proses*, Elex Media Komputindo, Jakarta, 1994.
3. Yusuf Gayo M, *Perbaikan dan Pengaturan Sungai*, PT Pradnya Paramita, Jakarta, 1984.
4. Firdaus F, *Rancang Bangun Sistem Pengendalian Level dan Laju Aliran Pada Sungai (Deteksi dan Telemetri) Untuk Mendeteksi Banjir Berbasis Mikrokontroller*, Surabaya, 2003.
5. Koselan S W, *Dasar Motor Stepper*, Jakarta 2002.
6. National Semiconductor, *National Data Acquisition Databook*, 1995 Edition
7. Ogata. K, *Modern Control Engineering*, Prentice Hall, New York, 1970.
8. Copper W D, *Instrumentasi Elektronik dan Teknik Pengukuran*, Erlangga, Jakarta, 1994.
9. Pranata A, *Pemrograman Borland Delphi edisi 3*, Andi Offset, Yogyakarta, 2000.
10. Dandekar MM & Sharma KN, *Pembangkit Listrik Tenaga Air*, Universitas Indonesia, Jakarta, 1991.

## **LAMPIRAN**

### REGRESI KETINGGIAN PINTU AIR I RATA-RATA

Pintu A	Ketinggian Rata-rata (qo)	qi x qo	qi2	m	b	Res	(Res)2
1	0,23	0,232	1	0,23847	-0,0096	0,003127273	1E-05
2	0,45	0,9	4	0,23847	-0,0096	-0,017345455	0,0003
3	0,73	2,196	9	0,23847	-0,0096	0,026181818	0,0007
4	0,93	3,728	16	0,23847	-0,0096	-0,012290909	0,0002
5	1,19	5,97	25	0,23847	-0,0096	0,011236364	0,0001
6	1,42	8,52	36	0,23847	-0,0096	-0,001236364	2E-06
7	1,64	11,494	49	0,23847	-0,0096	-0,017709091	0,0003
8	1,91	15,296	64	0,23847	-0,0096	0,013818182	0,0002
9	2,11	19,008	81	0,23847	-0,0096	-0,024654545	0,0006
10	2,39	23,94	100	0,23847	-0,0096	0,018872727	0,0004
55	13,02	91,284	385				
25	= (Sigma qi) <sup>2</sup>						

$$SSR = 0,002744$$

$$S(q) = 0,00097$$

### REGRESI KETINGGIAN 0,2 CM

ke-	Ketinggian Rata-rata (qo)	qi x qo	qi2	m	b	Res	(Res)2
1	0,20	0,2	1	0,01	0,202	-0,012	0,0001
2	0,23	0,46	4	0,01	0,202	0,008	6E-05
3	0,23	0,69	9	0,01	0,202	-0,002	4E-06
4	0,27	1,08	16	0,01	0,202	0,028	0,0008
5	0,23	1,15	25	0,01	0,202	-0,022	0,0005
5	1,16	3,58	55				
25	= (Sigma qi) <sup>2</sup>						

$$SSR = 0,00148$$

$$S(q) = 0,000854$$

### REGRESI KETINGGIAN 0,4 CM

ke-	Ketinggian Rata-rata (qo)	qi x qo	qi2	m	b	Res	(Res)2
1	0,40	0,4	1	0,013	0,411	-0,024	0,0006
2	0,47	0,94	4	0,013	0,411	0,033	0,0011
3	0,43	1,29	9	0,013	0,411	-0,02	0,0004
4	0,50	2	16	0,013	0,411	0,037	0,0014
5	0,45	2,25	25	0,013	0,411	-0,026	0,0007
5	2,25	6,88	55				
25	= (Sigma qi) <sup>2</sup>						

$$SSR = 0,00411$$

$$S(q) = 0,002373$$

### REGRESI KETINGGIAN 0,7 CM

ke-	Ketinggian Rata-rata (qo)	qi x qo	qi2	m	b	Res	(Res)2
1	0,77	0,77	1	-0,011	0,765	0,016	0,0003
2	0,73	1,46	4	-0,011	0,765	-0,013	0,0002
3	0,73	2,19	9	-0,011	0,765	-0,002	4E-06
4	0,70	2,8	16	-0,011	0,765	-0,021	0,0004
5	0,73	3,65	25	-0,011	0,765	0,02	0,0004
5	3,66	10,87	55				
25	= (Sigma qi) <sup>2</sup>						

$$SSR = 0,00127$$

$$S(q) = 0,000733$$

### REGRESI KETINGGIAN 0,9 CM

ke-	Ketinggian Rata-rata (qo)	qi x qo	qi2	m	b	Res	(Res)2
	0,93	0,93	1	0,004	0,92	0,006	4E-05
2	0,93	1,86	4	0,004	0,92	0,002	4E-06

3	0,90	2,7	9	0,004	0,92	-0,032	0,001
4	0,97	3,88	16	0,004	0,92	0,034	0,0012
5	0,93	4,65	25	0,004	0,92	-0,01	0,0001
15	4,66	14,02	55				
25	= (Sigma q) <sup>2</sup>						

$$SSR = 0,00232$$

$$S(q) = 0,001339$$

### REGRESI KETINGGIAN 1,2 CM

ke-	Ketinggian Rata-rata (qo)	qi x qo	qi2	m	b	Res	(Res) <sup>2</sup>
1	1,17	1,17	1	0,006	1,176	-0,012	0,0001
2	1,20	2,4	4	0,006	1,176	0,012	0,0001
3	1,20	3,6	9	0,006	1,176	0,006	4E-05
4	1,20	4,8	16	0,006	1,176	0	0
5	1,20	6	25	0,006	1,176	-0,006	4E-05
15	5,97	17,97	55				
25	= (Sigma q) <sup>2</sup>						

$$SSR = 0,00036$$

$$S(q) = 0,000208$$

### REGRESI KETINGGIAN 1,4 CM

ke-	Ketinggian Rata-rata (qo)	qi x qo	qi2	m	b	Res	(Res) <sup>2</sup>
1	1,40	1,4	1	0,006	1,402	-0,008	6E-05
2	1,40	2,8	4	0,006	1,402	-0,014	0,0002
3	1,47	4,41	9	0,006	1,402	0,05	0,0025
4	1,40	5,6	16	0,006	1,402	-0,026	0,0007
5	1,43	7,15	25	0,006	1,402	-0,002	4E-06
15	7,10	21,36	55				
25	= (Sigma q) <sup>2</sup>						

$$SSR = 0,00344$$

$$S(q) = 0,001986$$

### REGRESI KETINGGIAN 1,7 CM

ke-	Ketinggian Rata-rata (qo)	qi x qo	qi2	m	b	Res	(Res) <sup>2</sup>
1	1,63	1,63	1	0,002	1,636	-0,008	6E-05
2	1,63	3,26	4	0,002	1,636	-0,01	0,0001
3	1,67	5,01	9	0,002	1,636	0,028	0,0008
4	1,65	6,6	16	0,002	1,636	0,006	4E-05
5	1,63	8,15	25	0,002	1,636	-0,016	0,0003
15	8,21	24,65	55				
25	= (Sigma q) <sup>2</sup>						

$$SSR = 0,00124$$

$$S(q) = 0,000716$$

### REGRESI KETINGGIAN 1,9 CM

ke-	Ketinggian Rata-rata (qo)	qi x qo	qi2	m	b	Res	(Res) <sup>2</sup>
1	1,93	1,93	1	0	1,912	0,018	0,0003
2	1,90	3,8	4	0	1,912	-0,012	0,0001
3	1,90	5,7	9	0	1,912	-0,012	0,0001
4	1,90	7,6	16	0	1,912	-0,012	0,0001
5	1,93	9,65	25	0	1,912	0,018	0,0003
15	9,56	28,68	55				
25	= (Sigma q) <sup>2</sup>						

$$SSR = 0,00108$$

$$S(q) = 0,000624$$

### REGRESI KETINGGIAN 2,1 CM

ke-	Ketinggian Rata-rata (qo)	qi x qo	qi2	m	b	Res	(Res) <sup>2</sup>
1	2,07	2,07	1	0,009	2,085	-0,024	0,0006

2	2,13	4,26	4	0,009	2,085	0,027	0,0007
3	2,13	6,39	9	0,009	2,085	0,018	0,0003
4	2,10	8,4	16	0,009	2,085	-0,021	0,0004
5	2,13	10,65	25	0,009	2,085	0	0
15	10,56	31,77	55				
25	= (Sigma q) <sup>2</sup>						

$$SSR = 0,00207$$

$$S(q) = 0,001195$$

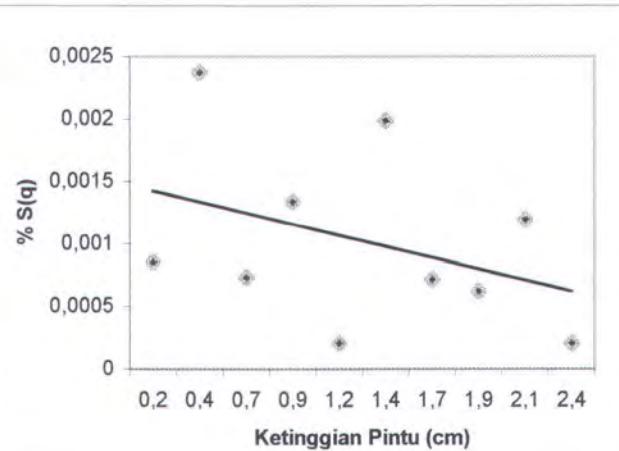
### REGRESI KETINGGIAN 2,4 CM

ke-	Ketinggian Rata-rata (qo)	qi x qo	qi <sup>2</sup>	m	b	Res	(Res) <sup>2</sup>
1	2,37	2,37	1	0,006	2,376	-0,012	0,0001
2	2,40	4,8	4	0,006	2,376	0,012	0,0001
3	2,40	7,2	9	0,006	2,376	0,006	4E-05
4	2,40	9,6	16	0,006	2,376	0	0
5	2,40	12	25	0,006	2,376	-0,006	4E-05
15	11,97	35,97	55				
25	= (Sigma q) <sup>2</sup>						

$$SSR = 0,01773$$

$$\text{total} = 0,006268502$$

el (cm)	% S(q)
0,2	0,000854478
0,4	0,00237291
0,7	0,000733235
0,9	0,001339453
1,2	0,000207846
1,4	0,001986085
1,7	0,000715914
1,9	0,000623538
2,1	0,001195115
2,4	0,000207846



## REGRESI KETINGGIAN PINTU AIR II RATA-RATA

Uji ke-	Ketinggian Pintu Air	Rata-rata	qi x qo	qi2	m	b	Res	(Res)2
1	0,22	0,224	1	0,25232	0,012267	-0,04058	0,00165	
2	0,51	1,012	4	0,25232	0,012267	-0,0109	0,00012	
3	0,78	2,346	9	0,25232	0,012267	0,01279	0,00016	
4	1,04	4,152	16	0,25232	0,012267	0,01647	0,00027	
5	1,32	6,59	25	0,25232	0,012267	0,04416	0,00195	
6	1,54	9,252	36	0,25232	0,012267	0,01584	0,00025	
7	1,78	12,474	49	0,25232	0,012267	0,00353	1,2E-05	
8	2,01	16,096	64	0,25232	0,012267	-0,01879	0,00035	
9	2,29	20,61	81	0,25232	0,012267	0,0069	4,8E-05	
10	2,51	25,06	100	0,25232	0,012267	-0,02942	0,00087	
55	14,00	97,816	385					
3025	= (Sigma qi) <sup>2</sup>							

$$SSR = 0,00568$$

$$S(q) = 0,002008$$

## REGRESI KETINGGIAN 0,2 CM

Uji ke-	Ketinggian Rata-rat	qi x qo	qi2	m	b	Res	(Res)2
1	0,23	0,23	1	-0,003	0,233	0	0
2	0,23	0,46	4	-0,003	0,233	0,003	9E-06
3	0,23	0,69	9	-0,003	0,233	0,006	3,6E-05
4	0,20	0,8	16	-0,003	0,233	-0,021	0,00044
5	0,23	1,15	25	-0,003	0,233	0,012	0,00014
15	1,12	3,33	55				
225	= (Sigma qi) <sup>2</sup>						

$$SSR = 0,00063$$

$$S(q) = 0,000364$$

## REGRESI KETINGGIAN 0,4 CM

Uji ke-	Ketinggian Rata-rat	qi x qo	qi2	m	b	Res	(Res)2
1	0,53	0,53	1	-0,006	0,524	0,012	0,00014
2	0,50	1	4	-0,006	0,524	-0,012	0,00014
3	0,50	1,5	9	-0,006	0,524	-0,006	3,6E-05
4	0,50	2	16	-0,006	0,524	0	0
5	0,50	2,5	25	-0,006	0,524	0,006	3,6E-05
15	2,53	7,53	55				
225	= (Sigma qi) <sup>2</sup>						

$$SSR = 0,00036$$

$$S(q) = 0,000208$$

## REGRESI KETINGGIAN 0,7 CM

Uji ke-	Ketinggian Rata-rat	qi x qo	qi2	m	b	Res	(Res)2
1	0,80	0,8	1	-0,006	0,8	0,006	3,6E-05
2	0,77	1,54	4	-0,006	0,8	-0,018	0,00032
3	0,80	2,4	9	-0,006	0,8	0,018	0,00032
4	0,77	3,08	16	-0,006	0,8	-0,006	3,6E-05
5	0,77	3,85	25	-0,006	0,8	0	0
15	3,91	11,67	55				
225	= (Sigma qi) <sup>2</sup>						

$$SSR = 0,00072$$

$$S(q) = 0,000416$$

## REGRESI KETINGGIAN 0,9 CM

Uji ke-	Ketinggian Rata-rat	qi x qo	qi2	m	b	Res	(Res)2
1	1,07	1,07	1	-0,008	1,062	0,016	0,00026
2	1,03	2,06	4	-0,008	1,062	-0,016	0,00026

3	1,03	3,09	9	-0,008	1,062	-0,008	6,4E-05
4	1,03	4,12	16	-0,008	1,062	0	0
5	1,03	5,15	25	-0,008	1,062	0,008	6,4E-05
15	5,19	15,49	55				
225	= (Sigma q) <sup>2</sup>						

$$\text{SSR} = \boxed{0,00064}$$

$$S(q) = \boxed{0,00037}$$

### REGRESI KETINGGIAN 1,2 CM

Uji ke-	ggian Rata-rat	qi x qo	qi <sup>2</sup>	m	b	Res	(Res) <sup>2</sup>
1	1,30	1,3	1	0,003	1,309	-0,012	0,00014
2	1,33	2,66	4	0,003	1,309	0,015	0,00023
3	1,33	3,99	9	0,003	1,309	0,012	0,00014
4	1,30	5,2	16	0,003	1,309	-0,021	0,00044
5	1,33	6,65	25	0,003	1,309	0,006	3,6E-05
15	6,59	19,8	55				
225	= (Sigma q) <sup>2</sup>						

$$\text{SSR} = \boxed{0,00099}$$

$$S(q) = \boxed{0,000572}$$

### REGRESI KETINGGIAN 1,4 CM

Uji ke-	ggian Rata-rat	qi x qo	qi <sup>2</sup>	m	b	Res	(Res) <sup>2</sup>
1	1,57	1,57	1	0	1,542	0,028	0,00078
2	1,50	3	4	0	1,542	-0,042	0,00176
3	1,57	4,71	9	0	1,542	0,028	0,00078
4	1,50	6	16	0	1,542	-0,042	0,00176
5	1,57	7,85	25	0	1,542	0,028	0,00078
15	7,71	23,13	55				
225	= (Sigma q) <sup>2</sup>						

$$\text{SSR} = \boxed{0,00588}$$

$$S(q) = \boxed{0,003395}$$

### REGRESI KETINGGIAN 1,7 CM

Uji ke-	ggian Rata-rat	qi x qo	qi <sup>2</sup>	m	b	Res	(Res) <sup>2</sup>
1	1,80	1,8	1	-0,006	1,8	0,006	3,6E-05
2	1,77	3,54	4	-0,006	1,8	-0,018	0,00032
3	1,80	5,4	9	-0,006	1,8	0,018	0,00032
4	1,77	7,08	16	-0,006	1,8	-0,006	3,6E-05
5	1,77	8,85	25	-0,006	1,8	-1,8E-15	3,2E-30
15	8,91	26,67	55				
225	= (Sigma q) <sup>2</sup>						

$$\text{SSR} = \boxed{0,00072}$$

$$S(q) = \boxed{0,000416}$$

### REGRESI KETINGGIAN 1,9 CM

Uji ke-	ggian Rata-rat	qi x qo	qi <sup>2</sup>	m	b	Res	(Res) <sup>2</sup>
1	2,03	2,03	1	5,7E-16	2,012	0,018	0,00032
2	2,00	4	4	5,7E-16	2,012	-0,012	0,00014
3	2,00	6	9	5,7E-16	2,012	-0,012	0,00014
4	2,00	8	16	5,7E-16	2,012	-0,012	0,00014
5	2,03	10,15	25	5,7E-16	2,012	0,018	0,00032
15	10,06	30,18	55				
225	= (Sigma q) <sup>2</sup>						

$$\text{SSR} = \boxed{0,00108}$$

$$S(q) = \boxed{0,000624}$$

### REGRESI KETINGGIAN 2,1 CM

Uji ke-	ggian Rata-rat	qi x qo	qi <sup>2</sup>	m	b	Res	(Res) <sup>2</sup>
1	2,27	2,27	1	0,01	2,26	0	0

2	2,27	4,54	4	0,01	2,26	-0,01	1E-04
3	2,27	6,81	9	0,01	2,26	-0,02	0,0004
4	2,37	9,48	16	0,01	2,26	0,07	0,0049
5	2,27	11,35	25	0,01	2,26	-0,04	0,0016
15	11,45	34,45	55				
225	= (Sigma q <sub>i</sub> ) <sup>2</sup>						

$$SSR = 0,007$$

$$S(q) = 0,004041$$

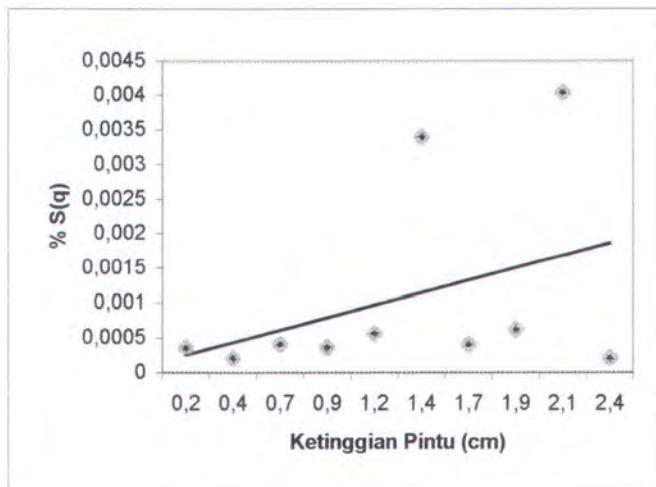
### REGRESI KETINGGIAN 2,4 CM

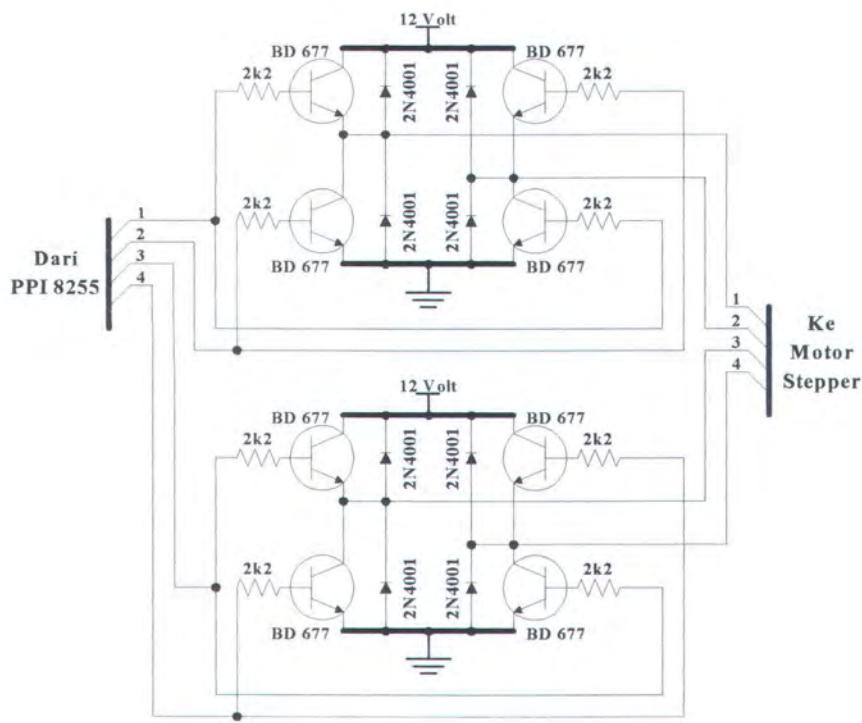
Uji ke-	Ketinggian Rata-rat	q <sub>i</sub> x q <sub>o</sub>	q <sub>i</sub> <sup>2</sup>	m	b	Res	(Res) <sup>2</sup>
1	2,50	2,5	1	0,006	2,488	0,006	3,6E-05
2	2,50	5	4	0,006	2,488	0	0
3	2,50	7,5	9	0,006	2,488	-0,006	3,6E-05
4	2,50	10	16	0,006	2,488	-0,012	0,00014
5	2,53	12,65	25	0,006	2,488	0,012	0,00014
15	12,53	37,65	55				
225	= (Sigma q <sub>i</sub> ) <sup>2</sup>						

$$\Sigma \text{q}_i = 0,01838$$

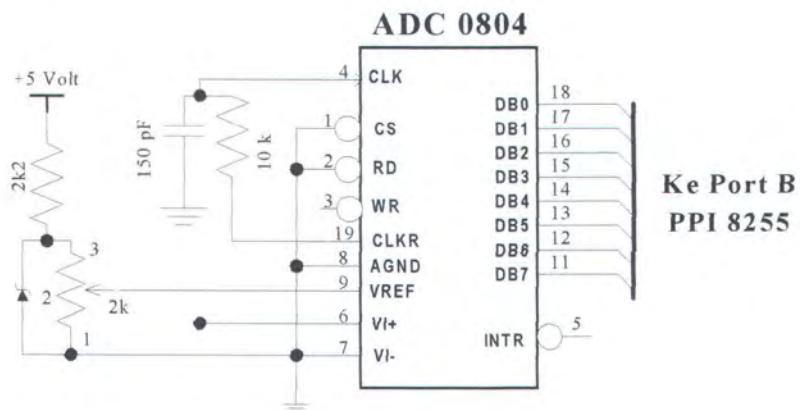
$$S(q)_{\text{total}} = 0,00649831$$

Level (cm)	% S(q)
0,2	0,00036373
0,4	0,00020785
0,7	0,00041569
0,9	0,0003695
1,2	0,00057158
1,4	0,00339482
1,7	0,00041569
1,9	0,00062354
2,1	0,00404145
2,4	0,00020785

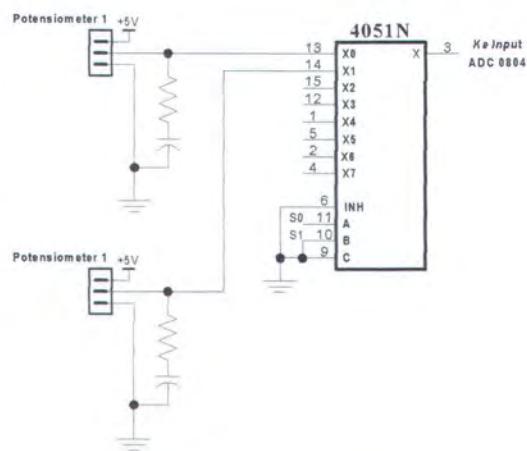




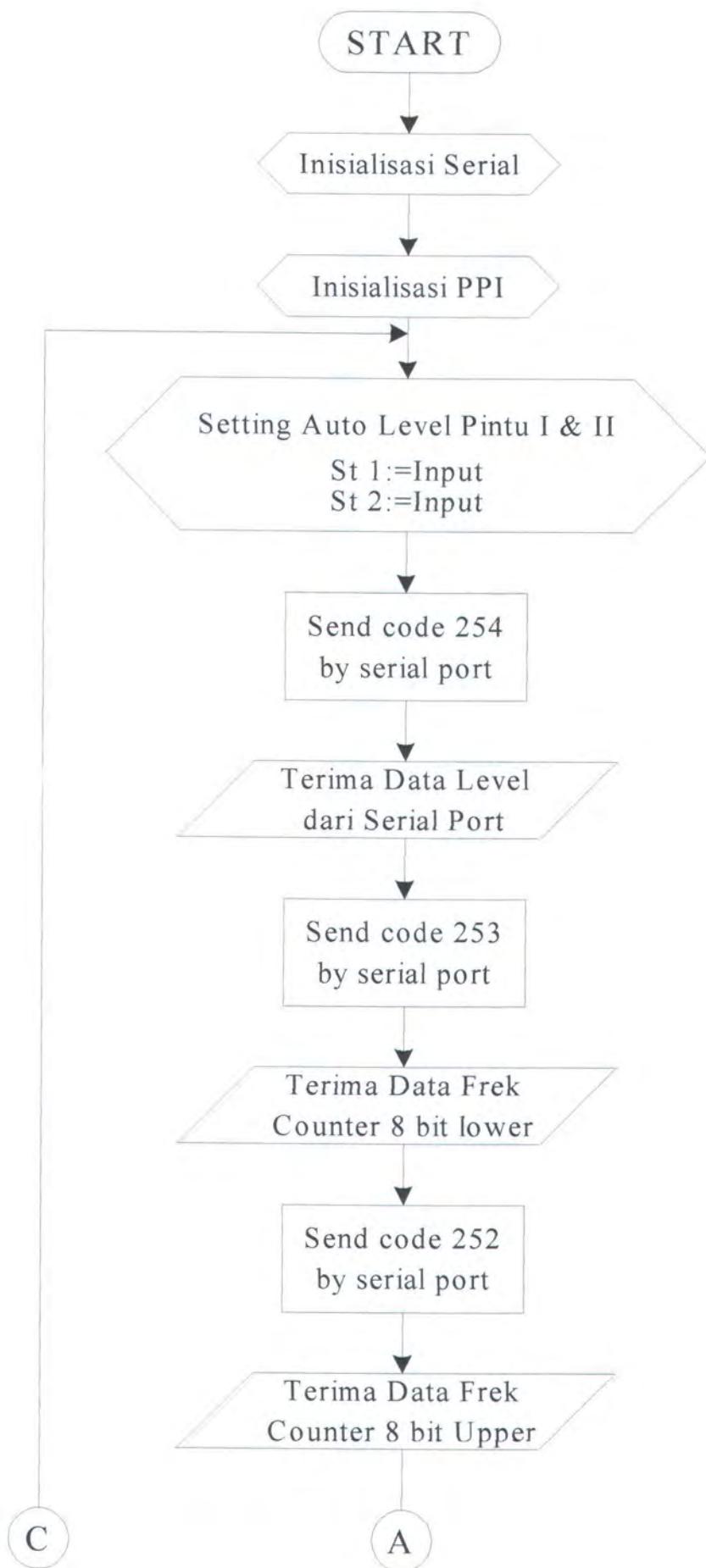
Rangkaian Driver motor stepper

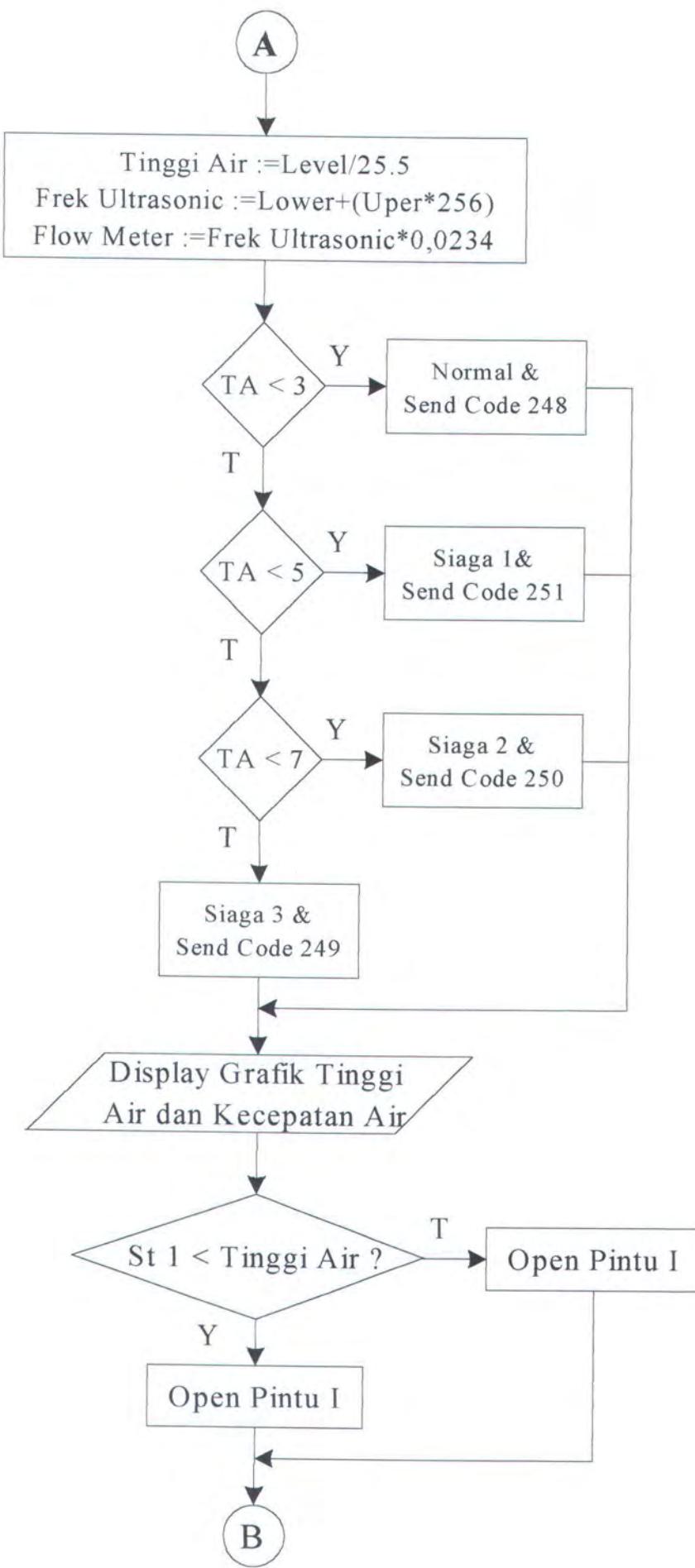


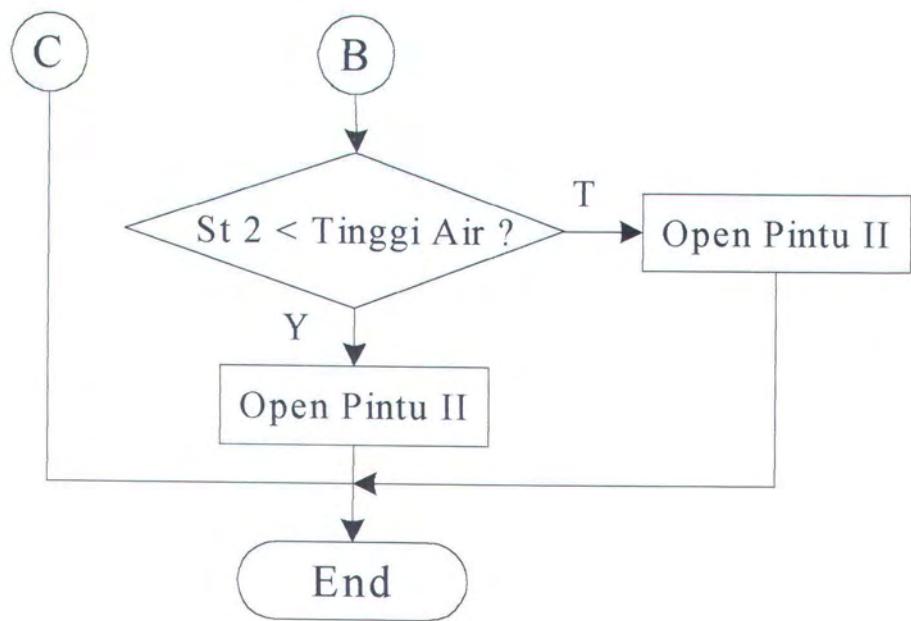
Rangkaian ADC 0804



Rangkaian potensiometer







```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  TeEngine, Series, ExtCtrls, TeeProcs, Chart, StdCtrls;

type
  TForm1 = class(TForm)
    Chart1: TChart;
    Start: TButton;
    Level: TEdit;
    Label1: TLabel;
    Stop: TButton;
    Chart2: TChart;
    Series2: TLineSeries;
    Flow: TEdit;
    Label2: TLabel;
    Edit2: TEdit;
    Edit3: TEdit;
    Label3: TLabel;
    Label4: TLabel;
    Timer1: TTimer;
    Label5: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    Label8: TLabel;
    Button1: TButton;
    Button2: TButton;
    Button3: TButton;
    Edit4: TEdit;
    Edit5: TEdit;
    Label9: TLabel;
    Label10: TLabel;
    Series1: TLineSeries;
    Edit1: TEdit;
    Label11: TLabel;
    Edit6: TEdit;
    Label12: TLabel;
    Procedure DrawGraph;
    procedure Receive;
    procedure BacaPintu;
    Procedure PortA;
    procedure SendD;
    procedure openPintuI;
    procedure closePintuI;
    procedure autopintuI;
    procedure openPintuII;
    procedure closePintuII;
    procedure autopintuII;
    procedure delay(lama:integer);
    procedure FormCreate(Sender: TObject);
    procedure StartClick(Sender: TObject);
    procedure StopClick(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    procedure SetStatusClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

uses Unit2, Unit3, Unit4;
```



```
{$R *.DFM}

var rbr,thr,ier,iir,lcr,mcr,lsr,msr,scr,dll,dlm,tampung:word;
P,Q,I,R,S,aver,Siaga1,Siaga2,Siaga3 :integer;
FloatLevel,x,TinggiI,TinggiIII,Tinggi,Aut1,Aut2,DataLevel,Counter :real;
Rage :array[1..200] of integer;
Data :array[1..200] of real;
Rata1,Rata2,RataI,RataII :array[1..25] of byte;
DataA,DataB,DataSend :byte;
Lower,Upper :integer;

procedure TForm1.PortA;
begin
  asm
    mov dx,380h //;port A
    mov al,DataA //;Data
    out dx,al //; ]
  end;
end;

Procedure Tform1.OpenPintuI;
begin
  If TinggiI<5 then
  begin
    DataA:=144;
    PortA;
    delay(5000);
    DataA:=80;
    PortA;
    delay(5000);
    DataA:=98;
    PortA;
    delay(5000);
    DataA:=160;
    PortA;
    delay(5000);
  end;
end;

Procedure Tform1.ClosePintuI;
begin
  If TinggiI>0 then
  begin
    DataA:=160;
    PortA;
    delay(5000);
    DataA:=98;
    PortA;
    delay(5000);
    DataA:=80;
    PortA;
    delay(5000);
    DataA:=144;
    PortA;
    delay(5000);
  end;
end;

Procedure Tform1.AutoPintuI;
begin
  if Edit4.Enabled=true then
  begin
    Aut1:=strtofloat(edit4.text);
    if DataLevel>Aut1 then OpenPintuI else
    ClosePintuI;
  end;
end;

Procedure Tform1.OpenPintuII;
begin
  If TinggiIII<5 then
  begin
    DataA:=5;
```

```
PortA;
delay(5000);
DataA:=6;
PortA;
delay(5000);
DataA:=10;
PortA;
delay(5000);
DataA:=9;
PortA;
delay(5000);
end;
end;

Procedure Tform1.ClosePintuII;
begin
  If TinggiII>0 then
  begin
    DataA:=9;
    PortA;
    delay(5000);
    DataA:=10;
    PortA;
    delay(5000);
    DataA:=6;
    PortA;
    delay(5000);
    DataA:=5;
    PortA;
    delay(5000);
  end;
end;

Procedure Tform1.AutoPintuII;
begin
  if Edit5.Enabled=true then
  begin
    Aut2:=strtofloat(edit5.text);
    if DataLevel>Aut2 then OpenPintuII else
      ClosePintuII;
  end;
end;

procedure TForm1.delay(lama:integer);
var I:Integer;
begin
  for I:=0 to lama do
    application.ProcessMessages;
end;

Procedure TForm1.BacaPintu;
begin
  inc(R);
  asm
    mov dx,382h
    mov al,00000000b //low-kan wr dan select A
    out dx,al
  end;
  delay(10000);
  delay(10000);
  asm
    mov dx,382h
    mov al,00100000b //high-kan wr dan select A
    out dx,al
  end;
  delay(1000);
  asm
    mov dx,381h
    in al,dx
    mov dataA,al
  end;
```

```
RataI[R]:=dataA;
//TinggiI:=((dataA/25.5));
asm
  mov dx,382h
  mov al,00010000b    //low-kan wr dan select B
  out dx,al
end;
delay(10000);
delay(10000);
asm
  mov dx,382h
  mov al,00110000b    //high-kan wr dan select B
  out dx,al
end;
delay(1000);
asm
  mov dx,381h
  in al,dx
  mov dataB,al
end;
RataII[R]:=dataB;
//TinggiII:=(((dataB)/10)-4.6)/2.4)-0.1;
if R=3 then
begin
  for S:=2 to 3 do
  begin
    if RataI[S]>RataI[S-1] then TinggiI:=RataI[S] else TinggiI:=RataI[S-1];
    if RataII[S]>RataII[S-1] then TinggiII:=RataII[S] else TinggiII:=RataII[S-1];
    //TinggiI:=TinggiI+RataI[S];
    //TinggiII:=TinggiII+RataII[S]
  end;
  TinggiI:=(TinggiI/30);
  TinggiII:=(TinggiII/30)-1.67;
  R:=0;
end;

If TinggiI<0 then TinggiI:=0;
Edit2.Text:=floattostrf(TinggiI,ffnumber,3,2);
with Form2 do
begin
  edit1.text:=Edit2.Text;
end;

If TinggiII<0 then TinggiII:=0;
Edit3.Text:=floattostrf(TinggiII,ffnumber,3,2);
with Form3 do
begin
  edit1.text:=Edit3.Text;
end;
//end;
end;

procedure TForm1.Receive;
label metu;
begin
asm
  mov dx,lsr
  in al,dx

  and al,01h
  cmp al,01h
  jne metu
  mov dx,rbr
  in al,dx
  mov tampung,al
metu:
  end;
end;

procedure TForm1.SendD;
label CEK;
begin
asm
```

```
CEK:  
    mov dx,3fdh{ ngecek transmit register}  
    in al,dx  
    and al,00100000b  
    jz cek  
    mov dx,3f8h  
    mov al,DataSend  
    out dx,al  
end;  
end;  
  
procedure TForm1.FormCreate(Sender: TObject);  
begin  
for S:=0 to 5 do  
begin  
    RataI[S]:=0;  
    RataII[S]:=0;  
end;  
Siagal:=0;  
Siaga2:=0;  
Siaga3:=0;  
aver:=0;  
DataSend:=0;  
Lower:=0;  
Counter:=0;  
Uper:=0;  
DataLevel:=0;  
rbr:=1016;  
thr:=1016;  
ier:=1017;  
iir:=1018;  
lcr:=1019;  
mcr:=1020;  
lsr:=1021;  
msr:=1022;  
scr:=1023;  
dll:=1016;  
dlm:=1017;  
x:=0;  
asm  
    xor ax,ax  
    mov dx,LCR  
    mov al,83h {;8 data bit,1 stop bit,no parity,dlab access}  
    out dx,al  
  
    mov dx,dll  
    mov al,0ch {;9600bps, xtal 1.843200}  
    out dx,al  
  
    inc dx {; dlm register}  
    mov al,00 {;9600bps }  
    out dx,al  
  
    mov dx,lcr  
    mov al,03h{; dlab disable }  
    out dx, al  
  
    mov dx,ier {; all interrupt disable}  
    mov al,0h  
    out dx,al  
  
    inc dx  
    inc dx  
    inc dx  
    mov al,1fh {; mcr register }  
    out dx,al  
    mov al,0  
    out dx,al  
  
    inc dx  
    inc dx  
  
    mov dx,383h      {init PPI}
```

```

mov al,10000010b {A: Output;B :Input; C :Output}
out dx,al
end;
chart1.Series[0].Clear;
chart2.Series[0].Clear;
end;

Procedure TForm1.DrawGraph;
var tempcount,FD:integer;
templevel:real;
SiagaLevel:integer;
begin
DataLevel:=floatlevel;
SiagaLevel:=round(DataLevel);
Level.Text:=floattostr(floatlevel,ffnumber,3,2);

Siaga1:=3;
Siaga2:=5;
Siaga3:=7;
If (DataLevel>Siaga1) and (DataLevel<Siaga2) then
begin
  DataSend:=251;
  SendD;
  Delay(5000);
  Edit1.Text:='Siaga 1';
end else
If (DataLevel>Siaga2) and (DataLevel<Siaga3) then
begin
  DataSend:=250;
  SendD;
  Delay(5000);
  Edit1.Text:='Siaga 2';
end else
If DataLevel>Siaga3 then
begin
  DataSend:=249;
  SendD;
  Delay(5000);
  Edit1.Text:='Siaga 3';
end else
begin
  DataSend:=248;
  SendD;
  Delay(5000);
  Edit1.Text:='Normal';
end;

DataSend:=253; //Ambil Data Lower
SendD;
Delay(5000);
Receive;
Lower:=Tampung;

DataSend:=252; //Ambil Data Uper
SendD;
Delay(5000);
Receive;
Upper:=Tampung;
tempcount:=round((Lower+(Upper*256))/1.0425);
tempcount:=(round(tempcount/1000)*100);
FD:=tempcount;
Edit6.Text:=inttostr(tempcount);
//FD:=40000-tempcount;
//edit1.Text:=inttostr(tempcount);
//Counter:=Round(tempcount/288);
//Counter:=Round(FD/288);
Counter:=((FD*0.00245486)-4.6);
//if counter<3.5 then Counter:=0;
Flow.Text:=floattostr(Counter,ffnumber,3,2);
chart1.Series[0].AddXY(x,Datalevel,'',clred);
chart1.Series[0].XLabel[p]:=TimetoStr(Time);;

```

```
chart2.Series[0].AddXY(x,Counter,'',clred);
chart2.Series[0].XLabel[p]:=TimetoStr(Time);
Label11.caption:=TimetoSTR(Time);
Uper:=0;
x := x + 1;
P:=round(x);

BacaPintu;
AutoPintuI;
AutoPintuII;
end;

procedure TForm1.StartClick(Sender: TObject);
var Rata
loop
begin
  If Start.Enabled=true then
begin
  floatlevel:=0;
  Aver:=0;
  I:=2;
  Data[I]:=0;
  Start.Enabled:=false;
  Stop.Enabled:=true;
  Timer1.Enabled:=true;
  Tinggi:=0;
  R:=0;
  S:=0;
  for loop:=1 to 5 do
  begin
    Rata[loop]:=0;
  end;
end;
if Start.Enabled=false then
begin
repeat
  //BacaPintu;
  inc(aver);
  DataSend:=254; //Ambil Data Level
  SendD;
  Delay(2000);
  Receive;

  //floatlevel:=tampung;
  Rage[aver]:=round(255-tampung);
  if aver=200 then
  begin
    for loop:=1 to 200 do
    begin
      floatlevel:=floatlevel+Rage[loop];
    end;
    floatlevel:=((floatlevel/200)/25.5)-0.25;
    if floatlevel<0 then floatlevel:=0;
    aver:=0;
  end;
  application.ProcessMessages;
until start.Enabled=true;
end;
end;

procedure TForm1.StopClick(Sender: TObject);
begin
  If Stop.Enabled=true then
begin
  Timer1.Enabled:=false;
  Stop.Enabled:=false;
  Start.Enabled:=true;
end;
end;

procedure TForm1.Timer1Timer(Sender: TObject);
```

```
begin
  if x>chart1.BottomAxis.Maximum then
begin
  chart1.BottomAxis.Maximum:=chart1.BottomAxis.Maximum+1;
  chart1.BottomAxis.Minimum:=chart1.BottomAxis.Minimum+1;
end;
if x>chart2.BottomAxis.Maximum then
begin
  chart2.BottomAxis.Maximum:=chart2.BottomAxis.Maximum+1;
  chart2.BottomAxis.Minimum:=chart2.BottomAxis.Minimum+1;
end;
DrawGraph;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  form1.Enabled:=false;
  form2.show;
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
  form1.Enabled:=false;
  form3.show;
end;

procedure TForm1.Button3Click(Sender: TObject);
begin
  halt;
  close;
end;

procedure TForm1.setStatusClick(Sender: TObject);
begin
  Form1.enabled:=false;
  Form4.Show;
end;
end.
```

```
1: unit Unit2;
2:
3: interface
4:
5: uses
6:   Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
7:   StdCtrls;
8:
9: type
10:   TForm2 = class(TForm)
11:     Close: TButton;
12:     Open: TButton;
13:     OK: TButton;
14:     Edit1: TEdit;
15:     Label1: TLabel;
16:     Label2: TLabel;
17:     RadioButton1: TRadioButton;
18:     RadioButton2: TRadioButton;
19:     Auto1: TEdit;
20:     Procedure delay(lama:integer);
21:     Procedure PortA;
22:     procedure OKClick(Sender: TObject);
23:     procedure CloseClick(Sender: TObject);
24:     procedure OpenClick(Sender: TObject);
25:     procedure RadioButton1Click(Sender: TObject);
26:     procedure RadioButton2Click(Sender: TObject);
27:     procedure FormCreate(Sender: TObject);
28:   private
29:     { Private declarations }
30:   public
31:     { Public declarations }
32:   end;
33:
34: var
35:   Form2: TForm2;
36:
37: implementation
38:
39: uses Unit1;
40:
41: {$R *.DFM}
42: var DataA:byte;
43:   Tinggi:real;
44:   cek:boolean;
45:
46: procedure TForm2.delay(lama:integer);
47: var I:Integer;
48: begin
49:   for I:=0 to lama do
50:     application.ProcessMessages;
51: end;
52:
53:
54:
55: procedure TForm2.PortA;
56: begin
57:   asm
58:     mov dx,380h          //;port A
59:     mov al,DataA          //;Data
60:     out dx,al            //; ]
61:   end;
62: end;
63:
64: procedure TForm2.OKClick(Sender: TObject);
65: begin
66:   form2.Hide;
67:   form1.show;
68:   form1.Enabled:=true;
69:   DataA:=0;
70:   PortA;
71:   if cek=true then
72:   begin
73:     with Form1 do
74:       begin
75:         Edit4.Enabled:=true;
```

```
76:     end;
77:   end;
78:   with Form1 do
79:     begin
80:       Edit4.Text:=Autol.Text;
81:     end;
82:   end;
83:
84: procedure TForm2.CloseClick(Sender: TObject);
85: begin
86:   Close.Enabled:=false;
87:   //Putar Kiri
88:   Tinggi:=strtofloat(edit1.text);
89:   If tinggi>0 then
90:   begin
91:     DataA:=160;
92:     PortA;
93:     delay(5000);
94:     DataA:=98;
95:     PortA;
96:     delay(5000);
97:     DataA:=80;
98:     PortA;
99:     delay(5000);
100:    DataA:=144;
101:    PortA;
102:    delay(5000);
103:  end;
104: Close.Enabled:=true;
105: end;
106:
107: procedure TForm2.OpenClick(Sender: TObject);
108: begin
109:   Open.Enabled:=false;
110:   //Putar Kanan
111:   If tinggi<5 then
112:   begin
113:     DataA:=144;
114:     PortA;
115:     delay(5000);
116:     DataA:=80;
117:     PortA;
118:     delay(5000);
119:     DataA:=98;
120:     PortA;
121:     delay(5000);
122:     DataA:=160;
123:     PortA;
124:     delay(5000);
125:   end;
126: Open.Enabled:=true;
127: end;
128:
129: procedure TForm2.RadioButton1Click(Sender: TObject);
130: begin
131:   cek:=true;
132:   open.Enabled:=false;
133:   close.Enabled:=false;
134:   Autol.Enabled:=true;
135: end;
136:
137: procedure TForm2.RadioButton2Click(Sender: TObject);
138: begin
139:   cek:=false;
140:   open.Enabled:=true;
141:   close.Enabled:=true;
142:   Autol.Enabled:=false;
143:   with Form1 do
144:   begin
145:     edit4.Enabled:=false;
146:   end;
147: end;
148:
149: procedure TForm2.FormCreate(Sender: TObject);
150: begin
```

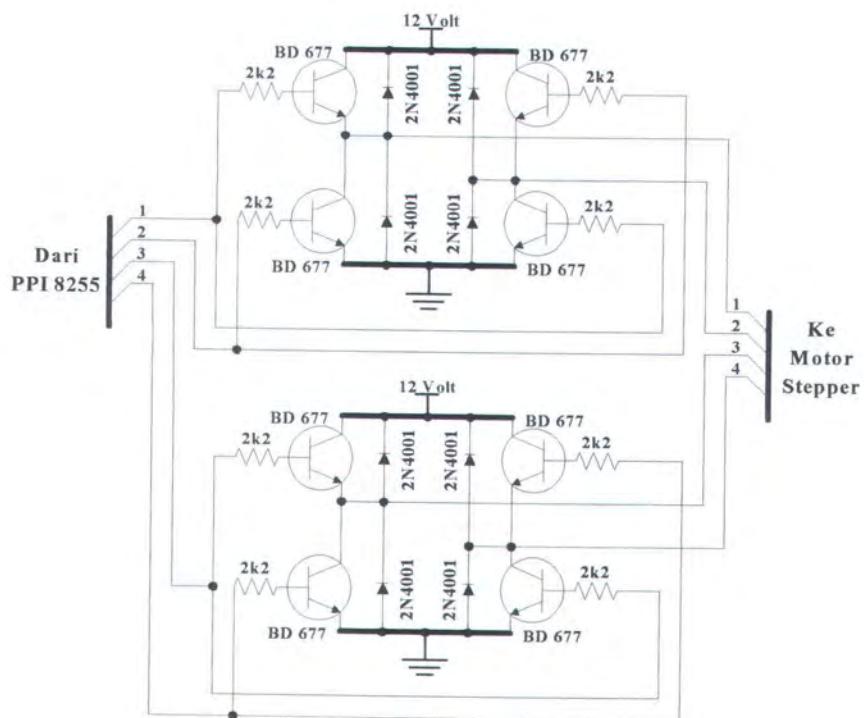
```
151: //Tinggi:=0;  
152: open.Enabled:=true;  
153: close.Enabled:=true;  
154: Auto1.Enabled:=false;  
155: end;  
156:  
157: end.
```

```
1: unit Unit3;
2:
3: interface
4:
5: uses
6:   Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
7:   StdCtrls;
8:
9: type
10:  TForm3 = class(TForm)
11:    Close: TButton;
12:    Open: TButton;
13:    OK: TButton;
14:    Label1: TLabel;
15:    Edit1: TEdit;
16:    Label2: TLabel;
17:    RadioButton1: TRadioButton;
18:    RadioButton2: TRadioButton;
19:    Auto2: TEdit;
20:    procedure delay(lama:integer);
21:    procedure PortA;
22:    procedure OKClick(Sender: TObject);
23:    procedure CloseClick(Sender: TObject);
24:    procedure OpenClick(Sender: TObject);
25:    procedure FormCreate(Sender: TObject);
26:    procedure RadioButton1Click(Sender: TObject);
27:    procedure RadioButton2Click(Sender: TObject);
28:    procedure StopClick(Sender: TObject);
29:  private
30:    { Private declarations }
31:  public
32:    { Public declarations }
33:  end;
34:
35: var
36:  Form3: TForm3;
37:
38: implementation
39:
40: uses Unit1;
41:
42: {$R *.DFM}
43:
44: var DataA:byte;
45:   Tinggi:real;
46:   cek:boolean;
47:
48: procedure TForm3.PortA;
49: begin
50:   asm
51:     mov dx,380h          //;port A
52:     mov al,DataA         //;Data           //;kirim instruksi
53:     out dx,al            //; ]
54:   end;
55: end;
56:
57: procedure TForm3.delay(lama:integer);
58: var I:Integer;
59: begin
60:   for I:=0 to lama do
61:     application.ProcessMessages;
62: end;
63:
64: procedure TForm3.OKClick(Sender: TObject);
65: begin
66:   form3.hide;
67:   form1.show;
68:   form1.Enabled:=true;
69:   DataA:=0;
70:   PortA;
71:   if cek=true then
72:   begin
73:     with Form1 do
74:     begin
75:       Edit5.Enabled:=true;
```

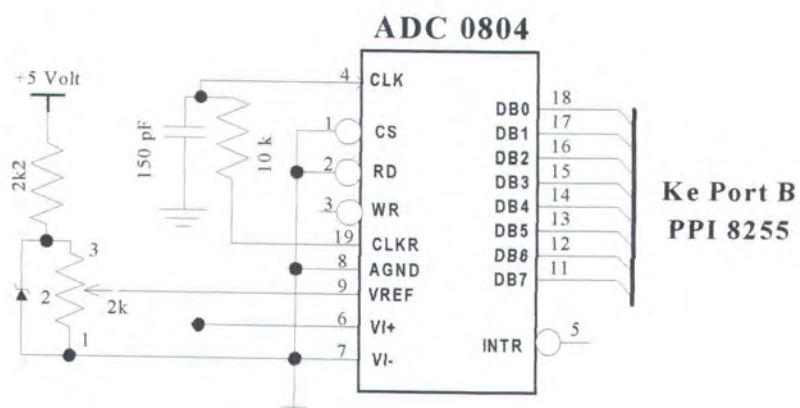
```
76:     end;
77:   end;
78:   with Form1 do
79:     begin
80:       edit5.text:=Auto2.Text;
81:     end;
82:   end;
83:
84: procedure TForm3.CloseClick(Sender: TObject);
85: begin
86:   Close.Enabled:=false;
87: //Putar Kiri
88:   Tinggi:=strtofloat(edit1.text);
89:   If tinggi>0 then
90:     begin
91:       DataA:=9;
92:       PortA;
93:       delay(5000);
94:       DataA:=10;
95:       PortA;
96:       delay(5000);
97:       DataA:=6;
98:       PortA;
99:       delay(5000);
100:      DataA:=5;
101:      PortA;
102:      delay(5000);
103:    end;
104:   Close.Enabled:=true;
105: end;
106:
107: procedure TForm3.OpenClick(Sender: TObject);
108: begin
109:   Open.Enabled:=false;
110: //Putar Kanan
111:   If tinggi<5 then
112:     begin
113:       DataA:=5;
114:       PortA;
115:       delay(5000);
116:       DataA:=6;
117:       PortA;
118:       delay(5000);
119:       DataA:=10;
120:       PortA;
121:       delay(5000);
122:       DataA:=9;
123:       PortA;
124:       delay(5000);
125:     end;
126:   Open.Enabled:=true;
127: end;
128:
129: procedure TForm3.FormCreate(Sender: TObject);
130: begin
131:   Tinggi:=0;
132:   open.Enabled:=true;
133:   close.Enabled:=true;
134:   Auto2.Enabled:=false;
135: end;
136:
137: procedure TForm3.RadioButton1Click(Sender: TObject);
138: begin
139:   cek:=true;
140:   open.Enabled:=false;
141:   close.Enabled:=false;
142:   Auto2.Enabled:=true;
143: end;
144:
145: procedure TForm3.RadioButton2Click(Sender: TObject);
146: begin
147:   cek:=false;
148:   open.Enabled:=true;
149:   close.Enabled:=true;
150:   Auto2.Enabled:=false;
```

```
151:   with Form1 do
152:     begin
153:       edit5.Enabled:=false;
154:     end;
155:   end;
156:
157: procedure TForm3.StopClick(Sender: TObject);
158: begin
159:   Open.Enabled:=true;
160:   Close.Enabled:=true;
161: end;
162:
163: end.
```

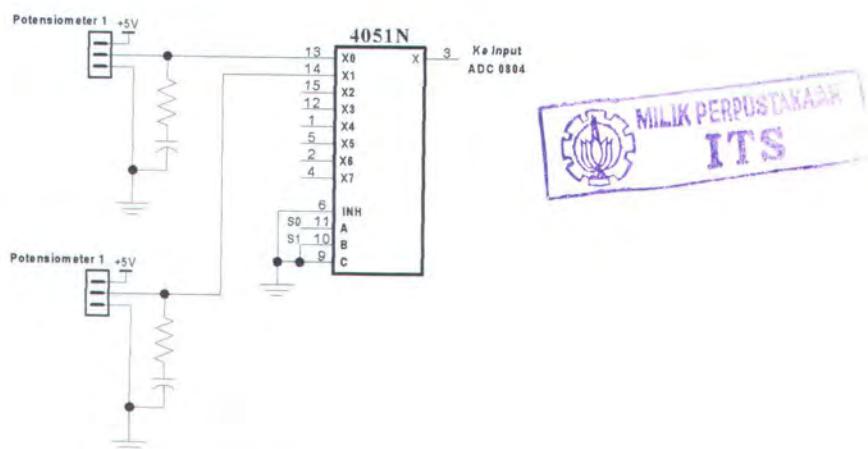
```
1: unit Unit4;
2:
3: interface
4:
5: uses
6:   Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
7:   StdCtrls, Spin;
8:
9: type
10:  TForm4 = class(TForm)
11:    OK: TButton;
12:    Label1: TLabel;
13:    Label2: TLabel;
14:    Label3: TLabel;
15:    Label4: TLabel;
16:    Label5: TLabel;
17:    Label6: TLabel;
18:    SpinEdit1: TSpinEdit;
19:    SpinEdit2: TSpinEdit;
20:    SpinEdit3: TSpinEdit;
21:    procedure OKClick(Sender: TObject);
22: private
23:   { Private declarations }
24: public
25:   { Public declarations }
26: end;
27:
28: var
29:  Form4: TForm4;
30:
31: implementation
32:
33: uses Unit1;
34:
35: {$R *.DFM}
36:
37: procedure TForm4.OKClick(Sender: TObject);
38: begin
39:  Form4.Hide;
40:  Form1.show;
41:  form1.Enabled:=true;
42:  //with Form1 do
43:  // begin
44:  //  Siaga1:=Spinedit1.Value;
45:  //  Siaga2:=Spinedit2.Value;
46:  //  Siaga3:=Spinedit3.Value;
47:  //end;
48: end;
49:
50: end.
```



Rangkaian Driver motor stepper



Rangkaian ADC 0804



Rangkaian potensiometer

# PETA PENGAMATAN SIAGA BANJIR KALI BRANTAS

## TAHUN 2001/2002

SIAGA	ELEVASI SHVP (m)	FREE BOARD (m)	DEBIT (m³/detik)
	≤ 3.55		≥ 1000
H	≥ 29.80	≤ 2.60	≥ 1250
K	≥ 30.25	≤ 2.20	> 1500
M	≥ 30.85	≤ 2.20	> 1500

**18. PLOSO**

SIAGA	ELEVASI SHVP (m)	DEBIT OUTFLOW (m³/detik)
	≥ 109.00	
H	≥ 108.10	> 100
K	≥ 109.10	> 200
M	≥ 109.20	> 400

**17. BENDUNG BENING**

SIAGA	ELEVASI SHVP (m)	DEBIT OUTFLOW (m³/detik)
	≥ 100	
H	≥ 109.00	> 100
K	≥ 108.10	> 200
M	≥ 109.20	> 400

**16. LENGKONG WIDAS**

SIAGA	ELEVASI SHVP (m)	FREE BOARD (m)	DEBIT (m³/detik)
	≤ 1.50		
H	≥ 40.70	≤ 1.50	≥ 400
K	≥ 41.05	≤ 1.15	≥ 460
M	≥ 41.30	≤ 0.80	≥ 500

**15. KERTOSONO**

SIAGA	ELEVASI SHVP (m)	FREE BOARD (m)	DEBIT (m³/detik)
	≤ 2.30		
H	≥ 41.45	≤ 2.30	≥ 800
K	≥ 41.75	≤ 2.00	≥ 860
M	≥ 42.05	≤ 1.70	≥ 1400

**14. BENDUNG MRICAN**

SIAGA	ELEVASI SHVP (m)	DEBIT OUTFLOW (m³/detik)
	≤ 800	
H	≥ 57.50	> 800
K	≥ 57.75	> 900
M	≥ 58.00	> 950

**13. KEDIRI**

SIAGA	ELEVASI SHVP (m)	FREE BOARD (m)	DEBIT (m³/detik)
	≤ 2.80		
H	≥ 61.20	≤ 2.80	≥ 750
K	≥ 61.45	≤ 2.65	≥ 850
M	≥ 61.80	≤ 2.30	≥ 950

**12. BEND. SEGawe**

SIAGA	ELEVASI SHVP (m)	DEBIT OUTFLOW (m³/detik)
	≤ 150	
H	≥ 248.20	> 150
K	≥ 248.70	> 200
M	≥ 249.10	> 280

**11. BEND. WONOREJO**

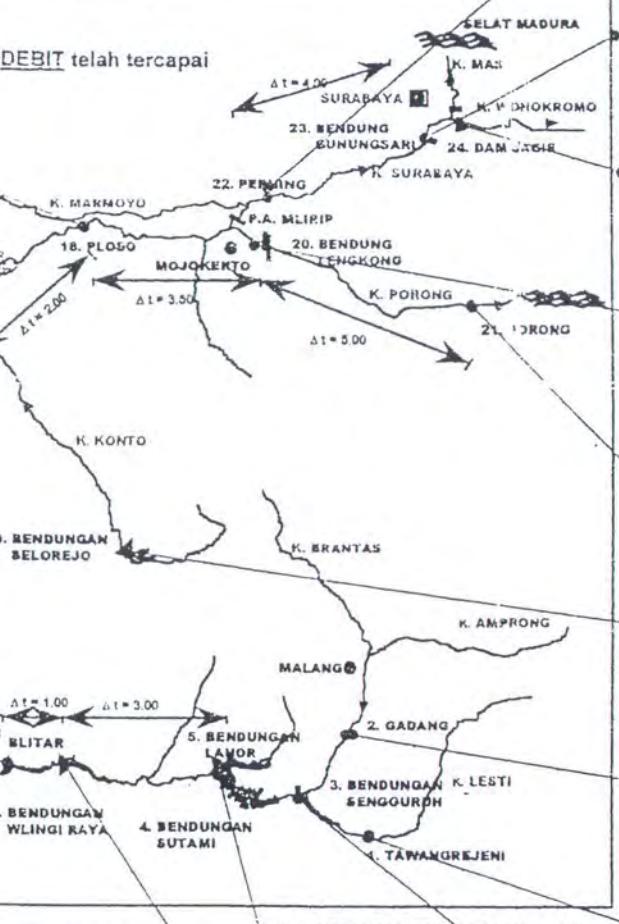
SIAGA	ELEVASI SHVP (m)	DEBIT OUTFLOW (m³/detik)
	≤ 100	
H	≥ 183.50	> 100
K	≥ 184.00	> 250
M	≥ 184.50	> 450

**KETERANGAN :**

SIAGA BANJIR DIMULAI apabila ELEVASI atau DEBIT telah tercapai

H = Hijau  
K = Kuning  
M = Merah

Δt = Waktu perjalanan air banjir (jam)


**22. PERNING**

SIAGA	ELEVASI SHVP (m)	FREE BOARD (m)	DEBIT (m³/detik)
H	≥ 10.10	≤ 5.50	≥ 200
K	≥ 10.30	≤ 5.30	≥ 220
M	≥ 10.65	≤ 4.95	≥ 260

**23. BENDUNG GUNUNGSAARI**

SIAGA	ELEVASI SHVP (m)	DEBIT OUTFLOW (m³/detik)
H	≥ 4.73	≥ 230
K	≥ 4.75	≥ 260
M	≥ 4.80	≥ 300

**24. PA JAGIR**

SIAGA	ELEVASI SHVP (m)	DEBIT OUTFLOW (m³/detik)
H	≥ 3.10	≥ 200
K	≥ 3.15	≥ 250
M	≥ 3.20	≥ 300

**20. BENDUNG LENGKONG**

SIAGA	ELEVASI SHVP (m)	DEBIT OUTFLOW (m³/detik)
H	≥ 18.00	≥ 1000
K	≥ 18.30	≥ 1250
M	≥ 18.50	≥ 1450

**21. PORONG**

SIAGA	ELEVASI SHVP (m)	FREE BOARD (m)	DEBIT (m³/detik)
H	≥ 6.10	≤ 2.30	≥ 1250
K	≥ 6.55	≤ 1.90	≥ 1350
M	≥ 7.20	≤ 1.20	≥ 1500

**19. BENDUNG SELOREJO**

SIAGA	ELEVASI SHVP (m)	DEBIT OUTFLOW (m³/detik)
H	≥ 622.20	≥ 200
K	≥ 622.60	≥ 260
M	≥ 623.00	≥ 360

**2. GADANG**

SIAGA	ELEVASI SHVP (m)	DEBIT (m³/detik)
H	≥ 390.00	≥ 200
K	≥ 392.60	≥ 400
M	≥ 394.30	≥ 570

**3. BENDUNG SENGGURU**

SIAGA	ELEVASI SHVP (m)	REFL-I	P	REFL-II	OUTPUT ENABLE
H	≥ 318.05	≥ 350			
K	≥ 319.00	≥ 500			
M	≥ 320.10	≥ 700			

TL/H/5672-1

Keterangan : Pengendalian banjir meliputi satu kesatuan Daerah Pengaliran Sungai ( Hulu, Tengah, Hilir )

please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Supply Voltage ( $V_{CC}$ ) (Note 3)	6.5V
Voltage at Any Pin Except Control Inputs	-0.3V to ( $V_{CC} + 0.3V$ )
Voltage at Control Inputs (START, OE, CLOCK, ALE, ADD A, ADD B, ADD C)	-0.3V to +15V
Storage Temperature Range	-65°C to +150°C
Package Dissipation at $T_A = 25^\circ C$	875 mW
Lead Temp. (Soldering, 10 seconds) Dual-In-Line Package (plastic)	260°C
Dual-In-Line Package (ceramic)	300°C
Molded Chip Carrier Package Vapor Phase (60 seconds)	215°C
Infrared (15 seconds)	220°C
ESD Susceptibility (Note 11)	400V

## Electrical Characteristics

Converter Specifications:  $V_{CC} = 5 V_{DC} = V_{REF(+)}$ ,  $V_{REF(-)} = GND$ ,  $T_{MIN} \leq T_A \leq T_{MAX}$  and  $f_{CLK} = 640$  kHz unless otherwise stated.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
	ADC0808					
	Total Unadjusted Error (Note 5)	25°C $T_{MIN}$ to $T_{MAX}$			$\pm \frac{1}{2}$ $\pm \frac{3}{4}$	LSB LSB
	ADC0809					
	Total Unadjusted Error (Note 5)	0°C to 70°C $T_{MIN}$ to $T_{MAX}$			$\pm 1$ $\pm 1\frac{1}{4}$	LSB LSB
	Input Resistance	From Ref(+) to Ref(-)	1.0	2.5		k $\Omega$
	Analog Input Voltage Range (Note 4) $V(+)$ or $V(-)$	$GND - 0.10$		$V_{CC} + 0.10$	$V_{DC}$	
$V_{REF(+)}$	Voltage, Top of Ladder	Measured at Ref(+)		$V_{CC}$	$V_{CC} + 0.1$	V
$V_{REF(+)} + V_{REF(-)}$ 2	Voltage, Center of Ladder		$V_{CC}/2 - 0.1$	$V_{CC}/2$	$V_{CC}/2 + 0.1$	V
$V_{REF(-)}$	Voltage, Bottom of Ladder	Measured at Ref(-)	-0.1	0		V
$I_{IN}$	Comparator Input Current	$f_c = 640$ kHz, (Note 6)	-2	$\pm 0.5$	2	$\mu A$

## Electrical Characteristics

Digital Levels and DC Specifications: ADC0808CJ,  $4.5V \leq V_{CC} \leq 5.5V$ ,  $-55^\circ C \leq T_A \leq +125^\circ C$  unless otherwise noted  
ADC0808CCJ, ADC0808CCN, ADC0808CCV, ADC0809CCN and ADC0809CCV,  $4.75 \leq V_{CC} \leq 5.25V$ ,  $-40^\circ C \leq T_A \leq +85^\circ C$  unless otherwise noted

ADC0808CJ	$-55^\circ C \leq T_A \leq +125^\circ C$
ADC0808CCJ, ADC0808CCN,	$-40^\circ C \leq T_A \leq +85^\circ C$
ADC0809CCN	$-40^\circ C \leq T_A \leq +85^\circ C$
ADC0808CCV, ADC0809CCV	

Range of  $V_{CC}$  (Note 1)

4.5 V<sub>DC</sub> to 6.0 V<sub>DC</sub>

ADC0808CCJ, ADC0808CCN, ADC0808CCV, ADC0809CCN and ADC0809CCV,  $4.75 \leq V_{CC} \leq 5.25V$ ,  $-40^\circ C \leq T_A \leq +85^\circ C$  unless otherwise noted

Symbol	Parameter	Conditions	Min	Typ	Max	Units
<b>CONTROL INPUTS</b>						
$V_{IN(1)}$	Logical "1" Input Voltage			$V_{CC} - 1.5$		V
$V_{IN(0)}$	Logical "0" Input Voltage				$1.5$	V
$I_{IN(1)}$	Logical "1" Input Current (The Control Inputs)	$V_{IN} = 15V$			$1.0$	$\mu A$
$I_{IN(0)}$	Logical "0" Input Current (The Control Inputs)	$V_{IN} = 0$	-1.0			$\mu A$
$I_{CC}$	Supply Current	$f_{CLK} = 640$ kHz		0.3	3.0	mA
<b>DATA OUTPUTS AND EOC (INTERRUPT)</b>						
$V_{OUT(1)}$	Logical "1" Output Voltage	$I_O = -360 \mu A$	$V_{CC} - 0.4$			V
$V_{OUT(0)}$	Logical "0" Output Voltage	$I_O = 1.6 mA$			0.45	V
$V_{OUT(0)}$	Logical "0" Output Voltage EOC	$I_O = 1.2 mA$			0.45	V
$I_{OUT}$	TRI-STATE Output Current	$V_O = 5V$ $V_O = 0$	-3		3	$\mu A$

## Electrical Characteristics

Timing Specifications  $V_{CC} = V_{REF(+)} = 5V$ ,  $V_{REF(-)} = GND$ ,  $t_r = t_f = 20$  ns and  $T_A = 25^\circ C$  unless otherwise noted.

Symbol	Parameter	Conditions	Min	Typ	Max	Units
$t_{WS}$	Minimum Start Pulse Width	(Figure 5)		100	200	ns
$t_{WALE}$	Minimum ALE Pulse Width	(Figure 5)		100	200	ns
$t_s$	Minimum Address Set-Up Time	(Figure 5)		25	50	ns
$t_H$	Minimum Address Hold Time	(Figure 5)		25	50	ns
$t_D$	Analog MUX Delay Time From ALE	$R_S = 0\Omega$ (Figure 5)		1	2.5	$\mu s$
$t_{H1}, t_{H0}$	OE Control to Q Logic State	$C_L = 50 pF, R_L = 10k$ (Figure 8)		125	250	ns
$t_{LH}, t_{OH}$	OE Control to Hi-Z	$C_L = 10 pF, R_L = 10k$ (Figure 8)		125	250	ns
$t_c$	Conversion Time	$f_c = 640$ kHz, (Figure 5) (Note 7)	90	100	116	$\mu s$
$f_c$	Clock Frequency		10	640	1280	kHz
$t_{EOC}$	EOC Delay Time	(Figure 5)	0		$8 + 2 \mu s$	Clock Periods
$C_{IN}$	Input Capacitance	At Control Inputs		10	15	pF
$C_{OUT}$	TRI-STATE Output Capacitance	At TRI-STATE Outputs, (Note 12)		10	15	pF

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.

Note 3: A zener diode exists, internally, from  $V_{CC}$  to GND and has a typical breakdown voltage of 7 V<sub>DC</sub>.

Note 4: Two on-chip diodes are tied to each analog input which will forward conduct for analog input voltages one diode drop below ground or one diode drop greater than the  $V_{CC}$  supply. The spec allows 100 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 100 mV, the output code will be correct. To achieve an absolute 0V<sub>DC</sub> to 5V<sub>DC</sub> input voltage range will therefore require a minimum supply voltage of 4.900 V<sub>DC</sub> over temperature variations, initial tolerance and loading.

Note 5: Total unadjusted error includes offset, full-scale, linearity, and multiplexer errors. See Figure 3. None of these A/Ds requires a zero or full-scale adjust. However, if an all zero code is desired for an analog input other than 0.0V, or if a narrow full-scale span exists (for example: 0.5V to 4.5V full-scale) the reference voltages can be adjusted to achieve this. See Figure 13.

Note 6: Comparator input current is a bias current into or out of the chopper stabilized comparator. The bias current varies directly with clock frequency and has little temperature dependence (Figure 6). See paragraph 4.

Note 7: The outputs of the data register are updated one clock cycle before the rising edge of EOC.

Note 8: Human body model, 100 pF discharged through a 1.5 k $\Omega$  resistor.

ed analog signal multiplexer. A particular input channel is selected by using the address decoder. Table I shows the input states for the address lines to select any channel. The address is latched into the decoder on the low-to-high transition of the address latch enable signal.

TABLE I

SELECTED ANALOG CHANNEL	ADDRESS LINE		
	C	B	A
IN0	L	L	L
IN1	L	L	H
IN2	L	H	L
IN3	L	H	H
IN4	H	L	L
IN5	H	L	H
IN6	H	H	L
IN7	H	H	H

#### CONVERTER CHARACTERISTICS

##### The Converter

The heart of this single chip data acquisition system is its 8-bit analog-to-digital converter. The converter is designed

wide range of temperatures. The converter is partitioned into 3 major sections: the 256R ladder network, the successive approximation register, and the comparator. The converter's digital outputs are positive true.

The 256R ladder network approach (Figure 1) was chosen over the conventional R/2R ladder because of its inherent monotonicity, which guarantees no missing digital codes. Monotonicity is particularly important in closed loop feedback control systems. A non-monotonic relationship can cause oscillations that will be catastrophic for the system. Additionally, the 256R network does not cause load variations on the reference voltage.

The bottom resistor and the top resistor of the ladder network in Figure 1 are not the same value as the remainder of the network. The difference in these resistors causes the output characteristic to be symmetrical with the zero and full-scale points of the transfer curve. The first output transition occurs when the analog signal has reached  $+1/2$  LSB and succeeding output transitions occur every 1 LSB later up to full-scale.

The successive approximation register (SAR) performs 8 iterations to approximate the input voltage. For any SAR type converter,  $n$ -iterations are required for an  $n$ -bit converter. Figure 2 shows a typical example of a 3-bit converter. In the ADC0808, ADC0809, the approximation technique is extended to 8 bits using the 256R network.

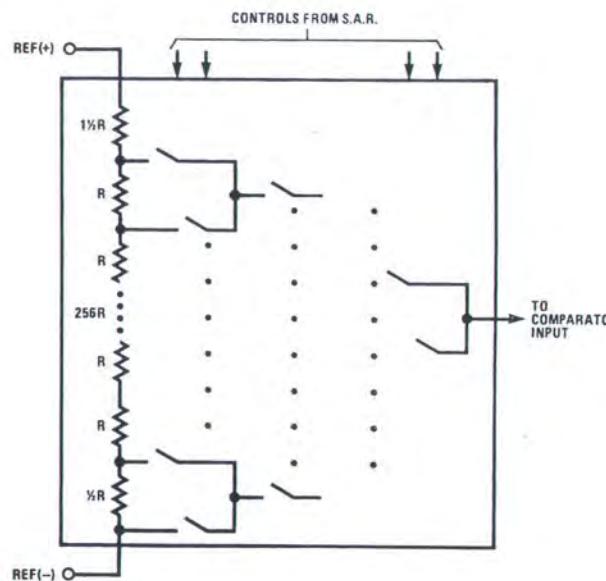


FIGURE 1. Resistor Ladder and Switch Tree

TL/H/5672-2

(SAR) is reset on the positive edge of the start conversion (SC) pulse. The conversion is begun on the falling edge of the start conversion pulse. A conversion in process will be interrupted by receipt of a new start conversion pulse. Continuous conversion may be accomplished by tying the end-of-conversion (EOC) output to the SC input. If used in this mode, an external start conversion pulse should be applied after power up. End-of-conversion will go low between 0 and 8 clock pulses after the rising edge of start conversion. The most important section of the A/D converter is the comparator. It is this section which is responsible for the ultimate accuracy of the entire converter. It is also the

repeatability of the device. A chopper-stabilized comparator provides the most effective method of satisfying all the converter requirements.

The chopper-stabilized comparator converts the DC input signal into an AC signal. This signal is then fed through a high gain AC amplifier and has the DC level restored. This technique limits the drift component of the amplifier since the drift is a DC component which is not passed by the AC amplifier. This makes the entire A/D converter extremely insensitive to temperature, long term drift and input offset errors.

Figure 4 shows a typical error curve for the ADC0808 as measured using the procedures outlined in AN-179.

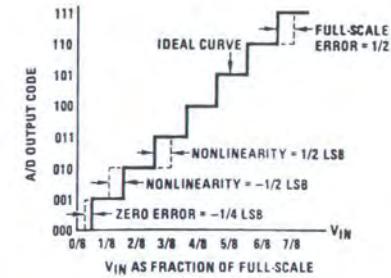


FIGURE 2. 3-Bit A/D Transfer Curve

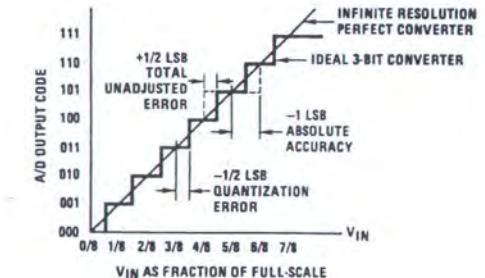


FIGURE 3. 3-Bit A/D Absolute Accuracy Curve

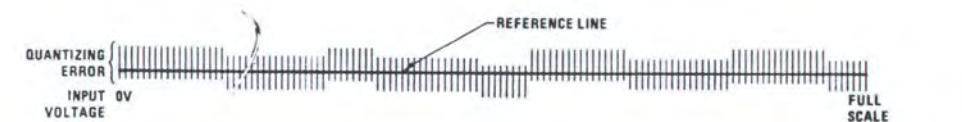
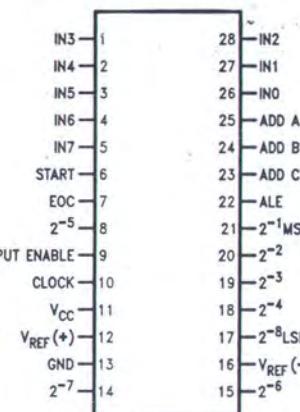


FIGURE 4. Typical Error Curve

TL/H/5672-3

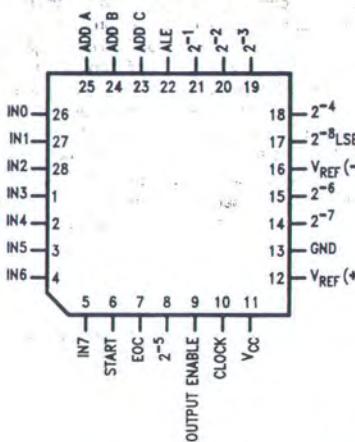
2-55

## Dual-In-Line Package



Order Number ADC0808CCN, ADC0809CCN,  
ADC0808CCJ or ADC0808CJ  
See NS Package J28A or N28A

## Molded Chip Carrier Package



Order Number ADC0808CCV or ADC0809CCV  
See NS Package V28A

## Timing Diagram

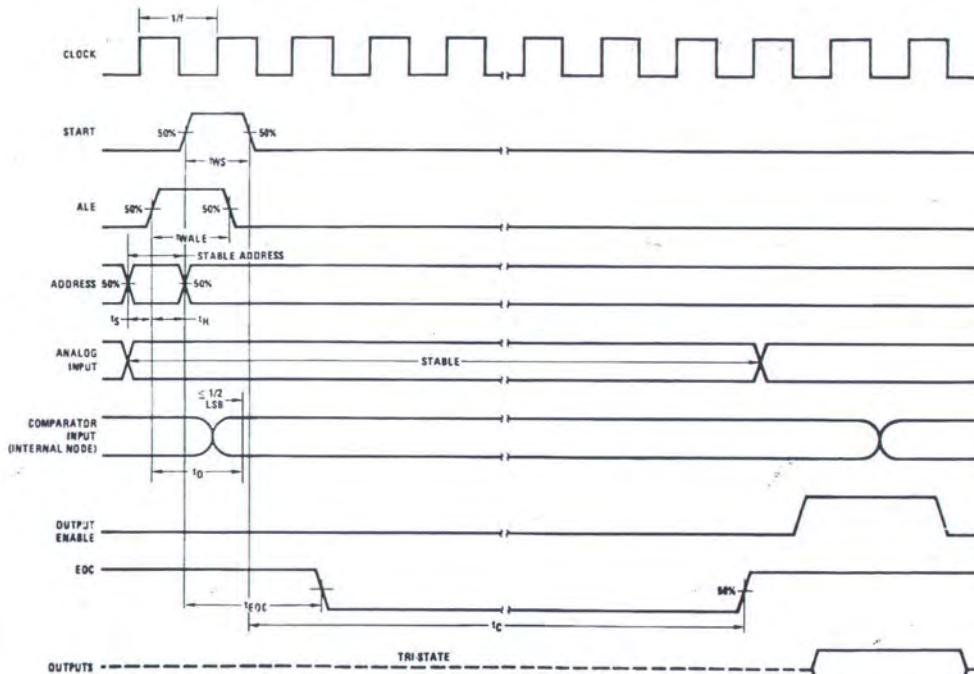
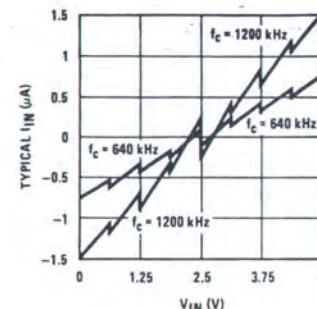
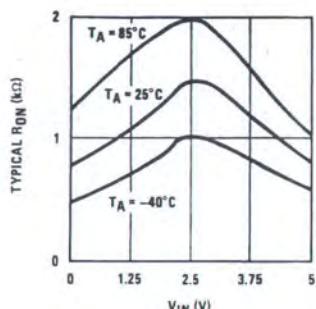


FIGURE 5

FIGURE 6. Comparator I<sub>IN</sub> vs V<sub>IN</sub>  
(V<sub>CC</sub> = V<sub>REF</sub> = 5V)FIGURE 7. Multiplexer R<sub>ON</sub> vs V<sub>IN</sub>  
(V<sub>CC</sub> = V<sub>REF</sub> = 5V)

## TRI-STATE Test Circuits and Timing Diagrams

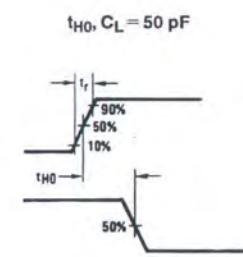
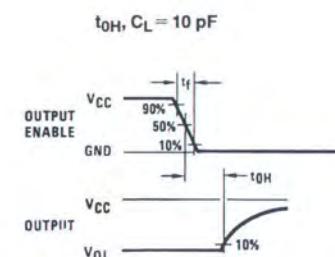
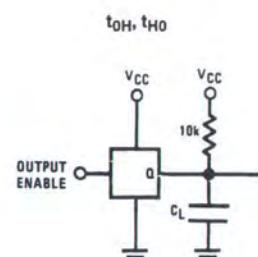
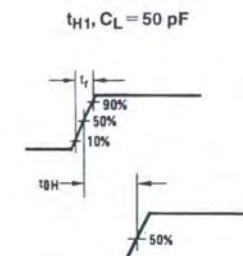
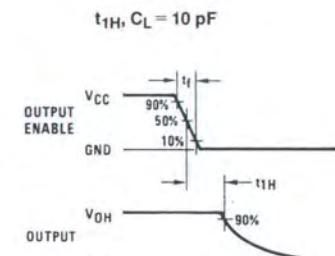
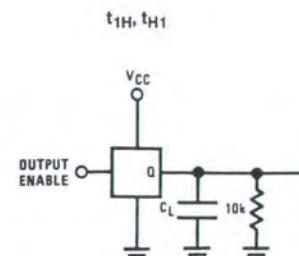


FIGURE 8

## 1.0 RATIO METRIC CONVERSION

The ADC0808, ADC0809 is designed as a complete Data Acquisition System (DAS) for ratiometric conversion systems. In ratiometric systems, the physical variable being measured is expressed as a percentage of full-scale which is not necessarily related to an absolute standard. The voltage input to the ADC0808 is expressed by the equation

$$\frac{V_{IN}}{V_{fs} - V_Z} = \frac{D_X}{D_{MAX} - D_{MIN}} \quad (1)$$

$V_{IN}$  = Input voltage into the ADC0808

$V_{fs}$  = Full-scale voltage

$V_Z$  = Zero voltage

$D_X$  = Data point being measured

$D_{MAX}$  = Maximum data limit

$D_{MIN}$  = Minimum data limit

A good example of a ratiometric transducer is a potentiometer used as a position sensor. The position of the wiper is directly proportional to the output voltage which is a ratio of the full-scale voltage across it. Since the data is represented as a proportion of full-scale, reference requirements are greatly reduced, eliminating a large source of error and cost for many applications. A major advantage of the ADC0808, ADC0809 is that the input voltage range is equal to the supply range so the transducers can be connected directly across the supply and their outputs connected directly into the multiplexer inputs, (Figure 9).

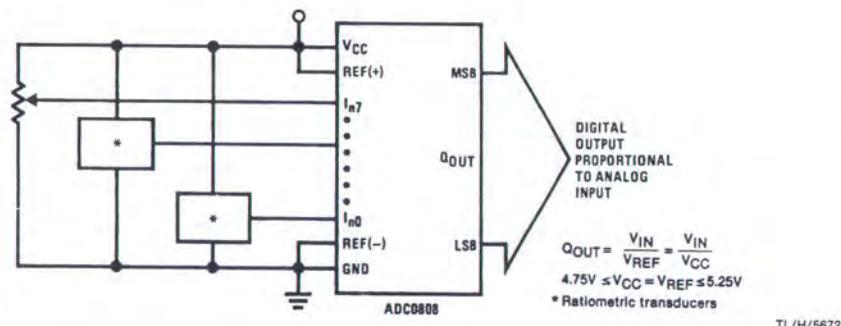


FIGURE 9. Ratiometric Conversion System

TL/H/5672-7

gauges, thermistor bridges, pressure transducers, etc., are suitable for measuring proportional relationships; however, many types of measurements must be referred to an absolute standard such as voltage or current. This means a system reference must be used which relates the full-scale voltage to the standard volt. For example, if  $V_{CC} = V_{REF} = 5.12V$ , then the full-scale range is divided into 256 standard steps. The smallest standard step is 1 LSB which is then 20 mV.

## 2.0 RESISTOR LADDER LIMITATIONS

The voltages from the resistor ladder are compared to the selected into 8 times in a conversion. These voltages are coupled to the comparator via an analog switch tree which is referenced to the supply. The voltages at the top, center and bottom of the ladder must be controlled to maintain proper operation.

The top of the ladder, Ref(+), should not be more positive than the supply, and the bottom of the ladder, Ref(-), should not be more negative than ground. The center of the ladder voltage must also be near the center of the supply because the analog switch tree changes from N-channel switches to P-channel switches. These limitations are automatically satisfied in ratiometric systems and can be easily met in ground referenced systems.

Figure 10 shows a ground referenced system with a separate supply and reference. In this system, the supply must be trimmed to match the reference voltage. For instance, if a 5.12V is used, the supply should be adjusted to the same voltage within 0.1V.

so developing the supply from the reference is readily as accomplished. In Figure 11 a ground referenced system is shown which generates the supply from the reference. The buffer shown can be an op amp of sufficient drive to supply the milliamper of supply current and the desired bus drive, or if a capacitive bus is driven by the outputs a large capacitor will supply the transient supply current as seen in Figure 12. The LM301 is overcompensated to insure stability when loaded by the 10  $\mu F$  output capacitor.

than  $V_{CC}$  and greater than ground. The center of the ladder voltage should always be near the center of the supply. The sensitivity of the converter can be increased, (i.e., size of the LSB steps decreased) by using a symmetrical reference system. In Figure 13, a 2.5V reference is symmetrically centered about  $V_{CC}/2$  since the same current flows in identical resistors. This system with a 2.5V reference allows the LSB bit to be half the size of a 5V reference system.

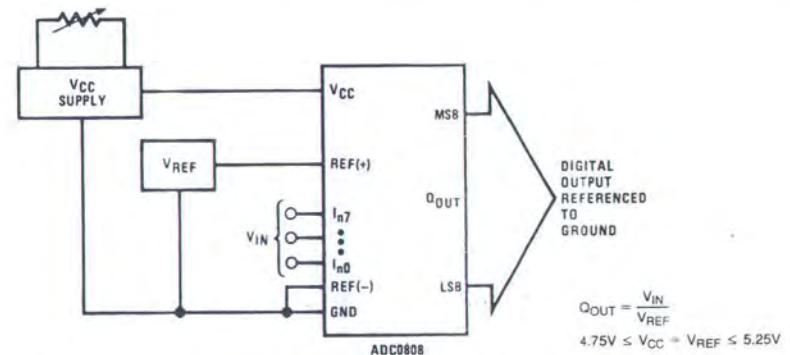


FIGURE 10. Ground Referenced Conversion System Using Trimmed Supply

2

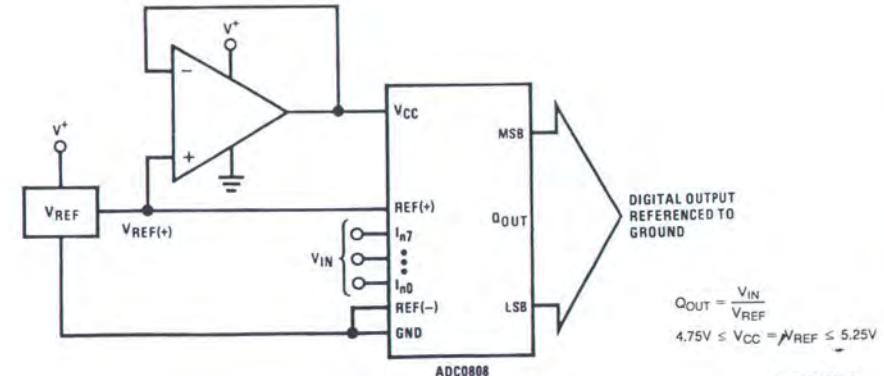


FIGURE 11: Ground Referenced Conversion System with Reference Generating  $V_{CC}$  Supply

TL/H/5672-8

2-59

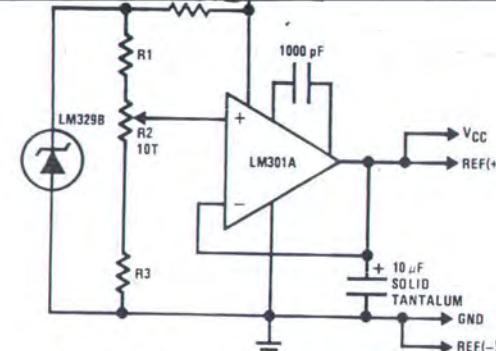


FIGURE 12. Typical Reference and Supply Circuit

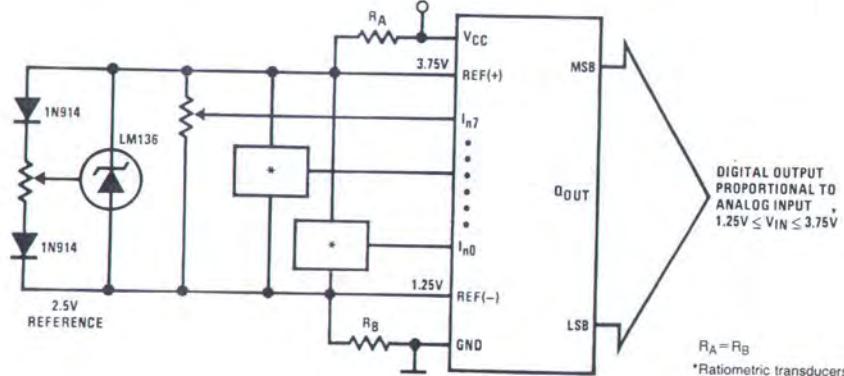


FIGURE 13. Symmetrically Centered Reference

### 3.0 CONVERTER EQUATIONS

The transition between adjacent codes  $N$  and  $N + 1$  is given by:

$$V_{IN} = \left\{ (V_{REF}(+) - V_{REF}(-)) \left[ \frac{N}{256} + \frac{1}{512} \right] \pm V_{TUE} \right\} + V_{REF}(-) \quad (2)$$

The center of an output code  $N$  is given by:

$$V_{IN} \left\{ (V_{REF}(+) - V_{REF}(-)) \left[ \frac{N}{256} \right] \pm V_{TUE} \right\} + V_{REF}(-) \quad (3)$$

The output code  $N$  for an arbitrary input are the integers within the range:

$$N = \frac{V_{IN} - V_{REF}(-)}{V_{REF}(+) - V_{REF}(-)} \times 256 \pm \text{Absolute Accuracy} \quad (4)$$

where:  $V_{IN}$  = Voltage at comparator input

$V_{REF}(+)$  = Voltage at Ref(+)

$V_{REF}(-)$  = Voltage at Ref(-)

$V_{TUE}$  = Total unadjusted error voltage (typically

$V_{REF}(+) \div 512$ )

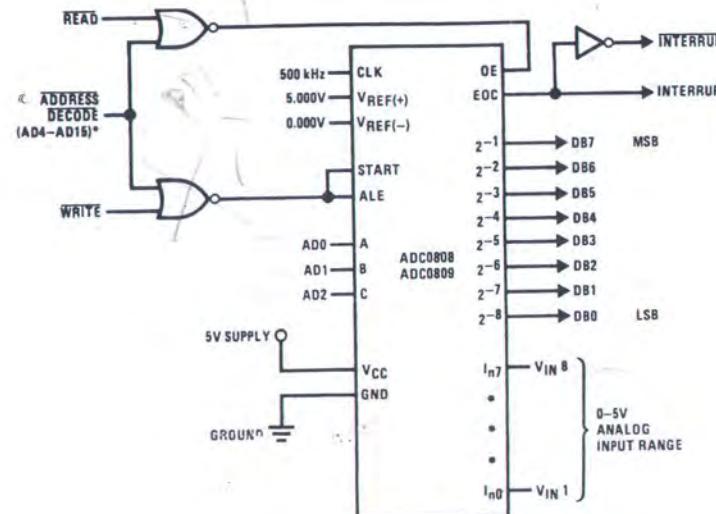
### 4.0 ANALOG COMPARATOR INPUTS

The dynamic comparator input current is caused by the periodic switching of on-chip stray capacitances. These are connected alternately to the output of the resistor ladder/splitter tree network and to the comparator input as part of the operation of the chopper stabilized comparator.

The average value of the comparator input current varies directly with clock frequency and with  $V_{IN}$  as shown in Figure 6.

If no filter capacitors are used at the analog inputs and the signal source impedances are low, the comparator input current should not introduce converter errors, as the transient created by the capacitance discharge will die out before the comparator output is strobed.

If input filter capacitors are desired for noise reduction and signal conditioning they will tend to average out the dynamic comparator input current. It will then take on the characteristics of a DC bias current whose effect can be predicted conventionally.



\*Address latches needed for 8085 and SC/MP interfacing the ADC0808 to a microprocessor

TL/H/5672-10

### MICROPROCESSOR INTERFACE TABLE

PROCESSOR	READ	WRITE	INTERRUPT (COMMENT)
8080	MEMR	MEMW	INTR (Thru RST Circuit)
8085	RD	WR	INTR (Thru RST Circuit)
Z-80	RD	WR	INT (Thru RST Circuit, Mode 0)
SC/MP	NRDS	NWDS	SA (Thru Sense A)
6800	VMA•φ2•R/W	VMA•φ•R/W	IRQA or IRQB (Thru PIA)

### Ordering Information

Error	TEMPERATURE RANGE			
	±½ LSB Unadjusted	ADC0808CCN	ADC0808CCV	ADC0808CCJ
	±1 LSB Unadjusted	ADC0809CCN	ADC0809CCV	
	Package Outline	N28A Molded DIP	V28A Molded Chip Carrier	J28A Ceramic DIP
				J28A Ceramic DIP