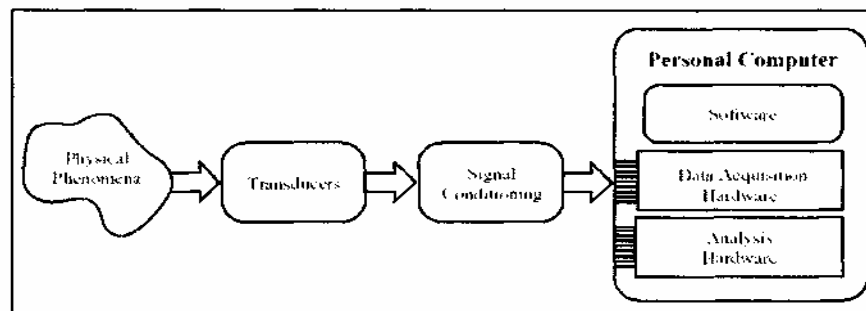


# KONSEP AKUISISI DATA DAN KONVERSI

## 5.1. DASAR-DASAR AKUISISI DATA

Elemen-elemen dasar dari sistem akuisisi data berbasis komputer (PC), sebagaimana ditunjukkan pada gambar 5.1, antara lain :

- Sebuah komputer PC;
- Transduser;
- Pengkondisi sinyal (*signal conditioning*);
- Perangkat keras akuisisi data;
- Perangkat keras analisa; dan
- Perangkat lunak yang terkait.



Gambar 5.1. Elemen-elemen sistem akuisisi data berbasis PC

### 5.1.1. Komputer Personal (PC)

Komputer yang digunakan dapat mempengaruhi kecepatan akuisisi data. Tipe-tipe transfer data yang tersedia pada komputer yang bersangkutan juga, secara signifikan, mempengaruhi unjuk-kerja dari sistem akuisisi data secara keseluruhan. Penggunaan DMA mampu meningkatkan unjuk-kerja melalui penggunaan perangkat keras terdedikasi (khusus) untuk mentransfer data langsung ke memori, sehingga prosesor bisa bebas mengerjakan tugas lain.

Faktor yang mempengaruhi jumlah data yang dapat disimpan dan kecepatan penyimpanan adalah kapasitas dan waktu akses *hard disk*. Dengan demikian, untuk sistem akuisisi data kontinyu dengan frekuensi sinyal yang diamati cukup tinggi akan dibutuhkan *hard disk* dengan waktu akses yang cepat dan kapasitas yang cukup besar.

*Hard disk* yang mengalami fragmentasi akan mengurangi laju akuisisi data.

Aplikasi-aplikasi akuisisi data secara *real-time* (waktu-nyata) membutuhkan prosesor yang cepat (dan tentunya akurat) atau menggunakan suatu prosesor terdedikasi seperti prosesor khusus untuk pemrosesan sinyal digital (DSP -*Digital Signal Processor*).

### **5.1.2. Transduser**

Transduser mendeteksi fenomena fisik (suhu, tekanan dan lain-lain) kemudian mengubahnya menjadi sinyal-sinyal listrik. Misalnya termokopel, RTD (*Resistive Temperature Detectors*), termistor, *flow-meter* dan lain-lain. Pada masing-masing kasus, sinyal listrik yang dihasilkan sebanding dengan parameter fisik yang diamati.

### **5.1.3. Pengkondisi Sinyal**

Sinyal-sinyal listrik yang dihasilkan oleh transduser harus dikonversi ke dalam bentuk yang dikenali oleh papan akuisisi data yang dipakai. Tugas pengkondisi sinyal yang sering dilakukan adalah penguatan (*amplification*). Misalnya sinyal-sinyal lemah yang berasal dari termokopel, sebaiknya dikuatkan untuk meningkatkan resolusi pengukuran. Dengan menempatkan penguat cukup dekat dengan transduser, maka interferensi atau gangguan yang timbul pada kabel penghubung antara transduser dengan komputer dapat diminimal-kan. Minimisasi terjadi karena sinyal telah dikuatkan sebelum menempuh perjalanan melalui kabel tersebut.

Tugas lain dari pengkondisi sinyal adalah melakukan linearisasi. Beberapa alat pengkondisi sinyal dapat melakukan penguatan sekaligus linearisasi untuk berbagai macam tipe transduser sedangkan jenis alat pengkondisi sinyal lainnya hanya bisa melakukan penguatan, linearisasinya menggunakan perangkat lunak (program) yang digunakan.

Aplikasi umum dari pengkondisi sinyal lainnya adalah melakukan isolasi sinyal dari transduser terhadap komputer untuk ke-amanan. Sistem yang diamati bisa mengandung perubahan-perubahan tegangan-tinggi yang dapat merusak komputer atau bahkan melukai operatornya.

Selain itu pengkondisi sinyal bisa juga melakukan penapisan sinyal yang diamati. Misalnya pengkondisi sinyal dengan penapis lo-los-rendah digunakan untuk meloloskan

sinyal-sinyal dengan frekuensi rendah dan menahan sinyal-sinyal dengan frekuensi tinggi.

## 5.2. PERANGKAT KERAS AKUISISI DATA (DAQ)

### 5.2.1. Masukan Analog

Spesifikasi papan perangkat keras akuisisi data meliputi jumlah kanal, laju pencuplikan, resolusi, jangkauan, ketepatan (akurasi), derau dan ketidak-linearitas, yang semuanya berpengaruh pada kualitas sinyal yang terdigitisasi (terakuisisi secara digital). Jumlah kanal masukan analog telah ditentukan, baik untuk masukan diferensial maupun ujung-tunggal (*single-ended*) pada papan akuisisi data yang memiliki kedua macam masukan tersebut. Masukan ujung-tunggal merupakan masukan dengan referensi titik pentanahan (*ground*) yang sama. Masukan-masukan ini digunakan untuk sinyal masukan yang memiliki aras tegangan yang cukup tinggi (lebih besar dari 1 volt), kabel penghubungnya juga cukup pendek (kurang dari 4,5 meter) dan semua sinyal masukan memiliki referensi *ground* yang sama. Jika sinyal-sinyal masukan tersebut tidak memenuhi kriteria ini, maka digunakan masukan diferensial, dengan tipe masukan diferensial ini, masing-masing masukan memiliki referensi *ground-nya* sendiri-sendiri. Ralat derau, dalam hal ini, dapat dikurangi karena derau *common-mode* (karena menggunakan referensi *ground* yang sama pada masukan ujung-tunggal) pada kabel sudah tidak ada.

Laju pencuplikan menentukan seberapa sering konversi data dilakukan. Laju pencuplikan yang cepat akan menghasilkan data yang lebih banyak dan akan menghasilkan penyajian-ulang sinyal asli yang lebih baik. Misal-nya, sinyal suara (*audio*) yang diubah ke sinyal listrik melalui mikrofon memiliki komponen frekuensi hingga mencapai 20 KHz. Untuk mendigitasi sinyal ini secara benar digunakan teorema Pencuplikan Nyquist yang mengatakan bahwa kita harus melakukan pencuplikan dengan laju atau frekuensi pencuplikan lebih besar dari dua kali komponen frekuensi maksimum yang ingin dideteksi (diakuisisi). Dengan demikian untuk sinyal audio tersebut diperlukan perangkat keras akuisisi data dengan frekuensi pencuplikan lebih dari 40 kHz (40.000 cuplikan tiap detik).

Sinyal-sinyal yang dihasilkan oleh transduser suhu biasanya tidak membutuhkan laju pencuplikan yang tinggi karena suhu tidak akan berubah secara cepat (pada

kebanyakan aplikasi). Dengan demikian, perangkat keras akuisisi data dengan laju pencuplikan rendah sudah mencukupi untuk digunakan pada akuisisi data suhu/temperatur.

Pemultipleksan merupakan cara yang sering digunakan untuk menambah jumlah kanal masukan ke ADC (papan akuisisi data). ADC yang bersangkutan mencuplik sebuah kanal, kemudian berganti ke kanal berikutnya, kemudian mencuplik kanal tersebut, berganti lagi ke kanal berikutnya dan seterusnya. Karena menggunakan sebuah ADC untuk mencuplik beberapa kanal, maka laju efektif pencuplikan pada masing-masing kanal berbanding terbalik dengan jumlah kanal yang dicuplik. Misalnya sebuah papan akuisisi data mampu mencuplik dengan laju 100 Kcuplik/detik pada 10 kanal, maka masing-masing kanal secara efektif memiliki laju pencuplikan :

$$\frac{100 \text{ kcuplik / detik}}{10 \text{ kanal}} = 10 \text{ kcuplik / detik}$$

Dengan kata lain laju pencuplikan menurun seiring dengan bertambahnya kanal yang dimultipleks.

Resolusi adalah istilah untuk jumlah atau lebar bit yang digunakan oleh ADC dalam penyajian-ulang sinyal analog. Semakin besar resolusinya, semakin besar pembagi jangkauan tegangan masukan sehingga semakin kecil perubahan tegangan yang bisa dideteksi. Pada gambar 5.2 ditunjukkan sebuah grafik gelombang sinus serta grafik digital yang diperoleh menggunakan ADC 3-bit.

Konverter 3-bit tersebut digunakan untuk membagi jangkauan sinyal analog menjadi  $2^3$  atau 8 bagian. Masing-masing bagian disajikan dalam kode-kode biner antara 000 hingga 111. Penyajian-ulang digital bukan merupakan penyajian-ulang yang baik dari sinyal analog asli karena ada informasi yang hilang selama proses konversi. Dengan meningkatkan resolusi hingga 16 bit, misalnya, maka jumlah kode-kode bilangan ADC meningkat dari 8 menjadi 65.536. Dengan demikian, penyajian-ulang digitalnya lebih akurat dibanding 3-bit.

Jangkauan berkaitan dengan tegangan minimum dan maksimum yang bisa ditangani oleh ADC yang bersangkutan. Papan akuisisi data ragam fungsi memiliki jangkauan yang bisa dipilih sedemikian rupa hingga mampu dikonfigurasi untuk menangani berbagai macam jangkauan tegangan yang berbeda-beda. Dengan fleksibilitas ini, anda

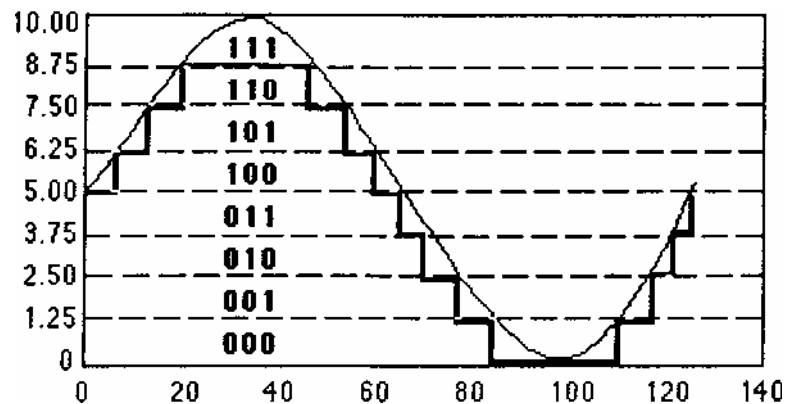
dapat menyesuaikan jangkauan sinyal masukan dengan jangkauan papan akuisisi data agar diperoleh resolusi yang akurat dan tepat untuk pengukuran sinyal yang bersangkutan.

Spesifikasi jangkauan, resolusi dan penguatan (*gain*) pada papan akuisisi data menentukan seberapa kecil perubahan tegangan yang mampu dideteksi. Perubahan tegangan ini menyatakan 1 LSB (*Least Signifincant Bit*) pada nilai digital dan sering dinamakan sebagai Lebar Kode (*code width*). Lebar kode yang ideal ditentukan menggunakan persamaan berikut :

$$\text{Lebar kode ideal} = \frac{\text{Jangkauan _ tegangan}}{\text{Penguatan} \left( 2^{\text{resolusi}} \right)}$$

Jika diketahui jangkauan tegangannya antara 0 sampai dengan 5 V dan penguatan 500 dan resolusi 16 bit, maka diperoleh :

$$\text{Lebar\_kode\_ideal} = 5 / (500 \times 2^{16}) = 153 \text{ nanovolt}$$



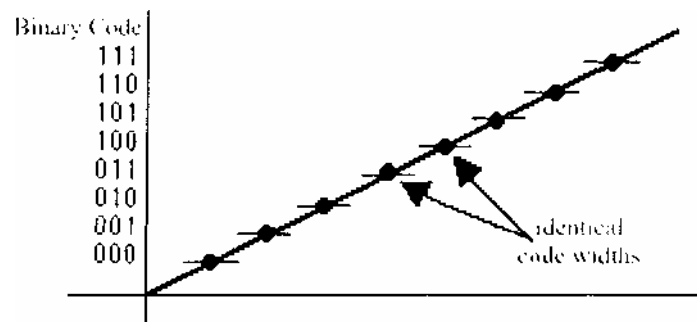
Gambar 5.2 Gelombang Sinus Terdigitasi dengan resolusi 3 bit

Ralat atau kesalahan lain yang mempengaruhi masukan analog adalah derau (*noise*). Derau ini bisa menurunkan resolusi ADC karena seiring dengan aras derau mencapai 1 LSB, ADC tidak mampu lagi membedakan antara kenaikan sinyal satu lebar kode dengan aras derau yang lebarnya sama. Ralat yang terkait dengan derau dapat dikurangi dengan mencuplik data pada laju yang tinggi serta melakukan rerata data terakuisisi tersebut.

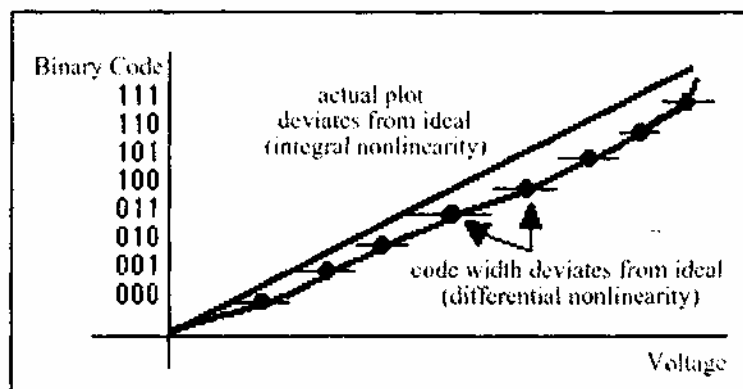
Idealnya, lebar kode pada masing-masing bagian tegangan adalah sama sebagaimana grafiknya ditunjukkan pada gambar 5.3. Non-linearitas integral pada suatu ADC menunjukkan seberapa jauh simpangan terhadap garis ideal (garis lurus). Sedangkan non-linearitas diferensial menunjukkan seberapa sama lebar kode

pada masing-masing bagian tegangan, perhatikan gambar 5.4.

Linearitas integral yang baik, adalah penting karena terjemahan akurat dari kode biner ke tegangan merupakan penskalaan yang sederhana. Non-linearitas diferensial yang ideal memastikan bahwa pembacaan tegangan yang diterjemahkan ada dalam  $\pm 0,5$  LSB dari tegangan masukan yang sebenarnya.



Gambar 5.3. Kode biner ideal vs Plot Tegangan



Gambar 5.4 Plot Tegangan Kode Biner dan ketidak linearan

### 5.2.2. Keluaran Analog

Rangkaian keluaran analog dibutuhkan untuk menstimulus suatu proses atau unit yang diuji pada sistem akuisisi data. Beberapa spesifikasi DAC yang menentukan kualitas sinyal keluaran yang dihasilkan adalah *settling time*, *slew rate* dan resolusi. *Settling time* dan *slew rate* bersama-sama menentukan seberapa cepat DAC dapat mengubah aras sinyal keluaran. *Settling time* adalah waktu yang dibutuhkan oleh keluaran agar stabil dalam durasi tertentu. *Slew rate* adalah laju perubahan maksimum agar DAC bisa menghasilkan keluaran. Dengan demikian, *settling time* yang kecil dan *slew rate* yang besar dapat menghasilkan sinyal-sinyal dengan frekuensi tinggi karena

hanya dibutuhkan waktu sebentar untuk mengubah keluaran ke aras tegangan baru secara akurat.

Suatu contoh aplikasi yang membutuhkan unjuk kerja tinggi dengan parameter-parameter tersebut adalah pembangkit sinyal-sinyal audio. DAC membutuhkan *slew rate* yang tinggi dan *settling time* yang kecil agar menghasilkan frekuensi pencuplikan tinggi yang cukup untuk mencakup jangkauan audio. Sebaliknya, suatu contoh aplikasi yang tidak membutuhkan konversi D/A yang cepat adalah aplikasi sumber tegangan yang digunakan untuk mengontrol pemanas (*heater*). Karena pemanas tidak mampu merespon secara cepat perubahan tegangan, maka tidak diperlukan waktu konversi D/A yang cepat.

Resolusi keluaran mirip dengan resolusi masukan. Yaitu jumlah bit kode digital yang (nantinya) akan menghasilkan keluaran analog. Semakin banyak jumlah bit resolusinya semakin berkurang besar kenaikan tegangan nya (semakin kecil perubahan tegangan yang mampu dideteksi), sehingga dimungkinkan untuk menghasilkan perubahan sinyal yang halus. Aplikasi yang membutuhkan jangkauan dinamis yang lebar dengan perubahan kenaikan tegangan yang kecil pada keluaran sinyal analog membutuhkan keluaran tegangan dengan resolusi tinggi.

### **5.2.3. Pemicuan**

Banyak aplikasi akuisisi data yang membutuhkan pemicuan eksternal yang digunakan untuk memulai dan menghentikan operasi akuisisi data. Pemicuan digital mensinkronkan antara akuisisi dan pembangkit tegangan ke suatu pulsa digital eksternal. Pemicu analog, yang banyak digunakan pada operasi masukan analog, akan memulai atau menghentikan operasi akuisisi data saat suatu sinyal masukan mencapai suatu aras dan *slope* suatu tegangan analog.

### **5.2.4. Digital I/O**

Antarmuka digital I/O sering digunakan pada sistem akuisisi data PC untuk mengontrol proses-proses, membangkitkan pola-pola pengujian dan untuk berkomunikasi dengan perangkat lain. Pada tiap-tiap kasus, parameter-parameter yang penting mencakup jumlah jalur digital yang tersedia, laju pemasukan dan pengeluaran data digital pada jalur-jalur tersebut dan kemampuan penggerakannya. Jika suatu jalur

digital digunakan untuk mengontrol suatu kejadian seperti menghidupkan dan mematikan pemanas, motor atau lampu, maka tidak dibutuhkan laju data yang tinggi karena peralatan-peralatan tersebut tidak dapat merespon dengan cepat. Pada contoh tersebut, jumlah arus yang dibutuhkan untuk menghidupkan dan mematikan alat harus lebih kecil dari arus penggerak yang disediakan oleh papan akuisisi data yang bersangkutan.

Suatu aplikasi umum lainnya adalah memindah data antara satu komputer dengan peralatan lain seperti *data logger*, pemroses data dan printer. Karena alat-alat ini biasanya menstransfer data dalam satuan byte atau 8 bit maka masing-masing jalur digital pada papan digital I/O dibentuk dalam kelompok 8. Selain itu beberapa papan memiliki rangkaian *handsaking* untuk tujuan sinkronisasi komunikasi. Jumlah kanal data dan kebutuhan *handsaking* harus sesuai (d disesuaikan) dengan aplikasi yang dibutuhkan.

#### **5.2.5. Pewaktuan I/O**

Rangkaian *pencacah/timer* berguna untuk berbagai macam aplikasi, termasuk menghitung jumlah kejadian-kejadian (*event*), mengukur pewaktu pulsa digital serta membangkitkan gelombang kotak. Semua hal tersebut dapat diimplementasikan menggunakan 3 sinyal pencacah/timer yaitu gerbang, sumber dan keluaran. Gerbang adalah suatu masukan digital yang digunakan untuk mengaktifkan dan mematikan fungsi pencacah. Sumber adalah masukan digital yang menyediakan pulsa-pulsa untuk menaikkan isi pencacah. Keluaran dari pencacah dapat berupa gelombang kotak atau pulsa-pulsa digital. Spesifikasi yang terkait dalam operasi *pencacah/timer* adalah resolusi dan frekuensi detak. Resolusi adalah jumlah bit pada pencacah. Semakin besar resolusinya mengakibatkan jumlah pencacahan semakin banyak. Sedangkan frekuensi detak menentukan seberapa cepat kerja dari pencacah/mer, artinya semakin tinggi frekuensinya semakin cepat pencacah itu bekerja sehingga mampu mendeteksi sinyal-sinyal masukan serta mampu menghasilkan pulsa dan gelombang kotak dengan frekuensi tinggi.

### **5.3. PERANGKAT KERAS PENGANALISA (*ANALYZER HARDWARE*)**



Kemampuan pemrosesan komputer pada saat ini telah mengalami peningkatan sedemikian rupa sehingga mencapai suatu tingkat kemampuan untuk melakukan akuisisi dan pemrosesan (analisa) data yang kompleks. Namun untuk aplikasi-aplikasi yang membutuhkan unjuk-kerja yang tinggi, seringkali komputer sudah tidak mampu lagi untuk melakukan pemrosesan data dengan cukup cepat untuk merespon sinyal-sinyal waktunya (*real-time*). Dengan demikian dibutuhkan perangkat keras tambahan yang harus dipasang pada komputer yang bersangkutan.

Prosesor sinyal digital dapat melakukan komputasi atau pemrosesan data lebih cepat dibandingkan dengan mikroprosesor pada umumnya, karena prosesor khusus tersebut mampu melakukan proses akumulasi dan multiplikasi data hanya dalam satu siklus detak, sedangkan mikroprosesor kebanyakan tidak dapat melakukan hal tersebut (dibutuhkan lebih dari satu siklus detak).

Saat ini prosesor sinyal digital telah tersedia dalam berbagai macam format dan tingkat akurasi. Misalnya prosesor sinyal digital 32-bit dengan format penyimpanan data *floating-point* (bilangan pecahan), memiliki jangkauan dinamis yang lebih tinggi dibandingkan dengan prosesor dengan format *fixed-point* (bilangan bulat). Sehingga aplikasi-aplikasi yang dikembangkan menggunakan prosesor *floating-point* ini tidak memerlukan pemrograman yang kompleks (*dibanding fixed-point*) untuk menangani data-data pecahan.

Kemampuan komputasi atau kalkulasi dari prosesor sinyal digital ini dinyatakan dalam jumlah operasi (komputasi) *floatingpoint* yang dapat dikerjakan dalam satu detik. Misalnya prosesor TMS320C30 dan Texas Instrument, mampu melakukan 33 juta operasi *floating-point* dalam satu detik (*Million Floating-point Operations Per Second = MFLOPS*).

#### **5.4. PERANGKAT LUNAK AKUISISI DATA (DAQ)**

Suatu perangkat lunak dan perangkat keras akuisisi data dapat merubah komputer PC menjadi suatu sistem akuisisi, pemroses (analisa) dan penampil data yang terpadu (*Data Acquisition System*).

Melakukan pemrograman langsung pada tingkat register pada papan akuisisi

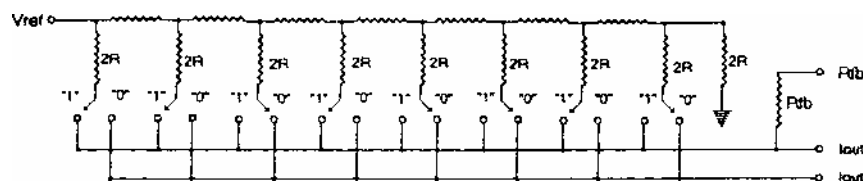
data merupakan tingkat pemrograman yang paling sulit dalam pengembangan perangkat lunak akuisisi data. Dalam hal ini, Anda harus menentukan nilai biner yang tepat dan benar yang harus dituliskan pada register-register tersebut. Selain itu, bahasa pemrograman yang digunakan harus mampu melakukan pembacaan dan penulisan data dari atau ke papan akuisisi data yang terpasang pada komputer.

Perangkat lunak akuisisi data dibagi menjadi dua macam: (1) Perangkat lunak aras-penggerak (*driver-level*) dan (2) Perangkat lunak aras-aplikasi (*application-level*). Perangkat lunak aras-penggerak menyederhanakan pemrograman akuisisi data dengan cara menangani secara langsung pemrograman aras-rendah (*low-level programming*) dan memberikan Anda berbagai fungsi aras-tinggi (*high-level functions*) yang dapat dipanggil dalam bahasa pemrograman yang Anda gunakan.

Perangkat lunak tingkat-aplikasi adalah perangkat lunak akuisisi data yang langsung bisa Anda gunakan, seperti Lab View, LabWindows dan lain-lain.

### 5.5. DAC (DIGITAL TO ANALOG CONVERTER)

Rangkaian pada gambar 5.5, diambil dari *data sheet* DAC 0832 yang merupakan suatu pendekatan dengan melakukan konversi dari data-data digital menjadi analog (tegangan) menggunakan rangkaian *tangga R 2R* (*R 2R ladder*). Nilai dari R dan Rfb sekitar 15 K ohm sehingga 2R-nya sekitar 30 Kohm. Nilai-nilai yang sebenarnya tidak terlalu penting karena kenyataannya nilai-nilai resistor tersebut masing-masing sangat dekat (sama) antara satu dengan yang lain.

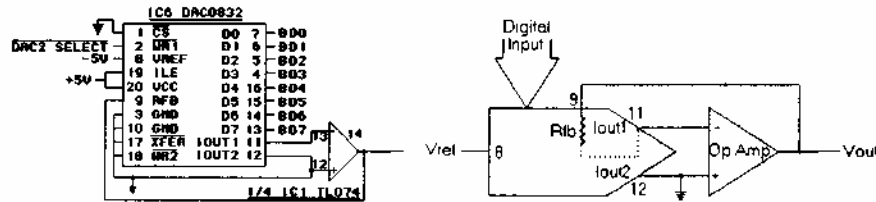


Gambar 5.5. Rangkaian tangga R2R

Logika "1" dan "0" mengindikasikan posisi-posisi saklar MOSFET yang ada dalam konverter. Saklar-saklar tersebut akan terhubung pada "1" jika bit yang terkait dalam kondisi ON dan akan terhubung "0" jika OFF. Suatu saklar yang terhubung ke posisi "1" akan meneruskan arus dari Vref ke Iout1, sedangkan saklar yang terhubung ke posisi "0" akan meneruskan arus dari Vref ke Iout2, masing-masing melalui resistor-resistor yang terkait.

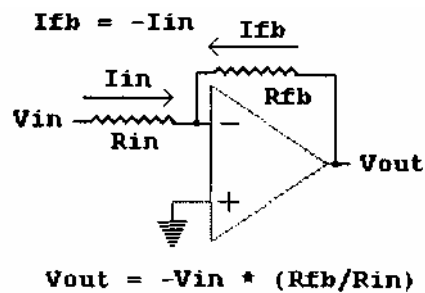
Untuk melihat bagaimana rangkaian tangga R 2R bekerja, perhatikan rangkaian

pada gambar 5.6. Gambar 5.6(a) merupakan rangkaian aplikasi DAC 0832 yang sederhana dan (b) merupakan diagram blok yang disederhanakan, diambil dari *data sheet*. Pada gambar 5.6(b) Rfb digambarkan sedemikian rupa sehingga Rfb ada didalam DAC, tapi dapat diakses dari luar dan dapat dihubungkan ke Op-amp.



Gambar 5.6. (a) Rangkaian aplikasi DAC 0832 (b) Blok diagram DAC 0832

Iout1 pada tangga R2R terhubung pada masukan terinversi pada Op-amp, sedangkan Iout2 terhubung pada masukan tak-terinversi dan ground. Resistor Rfb digunakan sebagai resistor umpan-balik ke Op-amp (dari keluaran ke masukan terinversi Op-amp). Dalam hal ini, agar Op-amp mampu menghilangkan arus yang melalui masukan terinversi maka arus melalui resistor umpan-balik, Rfb, harus sama dengan arus yang melalui resistor masukan terinversi namun dengan polaritas terbalik, perhatikan gambar 5.7.



Gambar 5.7

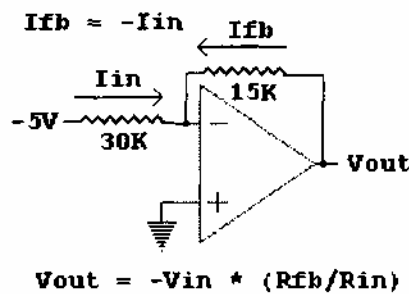
Karena arusnya sama tetapi daya polaritas yang berbeda maka hasilnya adalah tegangan 0 volt. Dengan demikian, masukan terinversi memiliki beda potensial yang sama dengan *ground*. Hal ini mengakibatkan terjadinya kesamaan pertanahan atau dinamakan pentanahan semu. Hal-hal yang bisa diketahui :

- $I_{fb} = -I_{in}$
- Tegangan dikiri Rfb adalah 0 berkaitan dengan pertanahan semu, sehingga

tegangan disebelah kanan adalah tegangan  $R_{fb}$   $\rightarrow V_{out} = R_{fb} \times I_{fb}$ .

- Karena  $I_{fb} = -I_{in}$  maka  $V_{out} = R_{fb} \times -I_{in}$  ;
- Arus yang dihilangkan pada masukan terinversi adalah  $I_{out1}$  maka  $V_{out} = R_{fb} \times -I_{out1}$ . (sebagaimana juga pada *data sheet*)

Jika digunakan tegangan referensi -5 volt, maka satu-satunya resistor yang terhubung pada  $I_{out1}$  adalah  $2R$  yang ada di paling kiri diagram tangga R2R jika hanya bit MSB saja yang ON (gambar 5.5), maka ( $2R = 30K\Omega$  dan  $R_{fb} = 15K\Omega$  serta  $V_{ref} = -5V$ ) diperoleh rangkaian gambar 5.8.



Gambar 5.8

$$\text{Sekarang } V_{out} = -(-5V) \times \frac{15K}{30K} = 2.5 \text{ volt} .$$

Ingat bahwa bit MSB memiliki bobot 128. Pada *data sheet* juga dikatakan bahwa :

$$V_{out} = \frac{-V_{ref} \times (\text{Bobot Digital})_{10}}{256}$$

angka "10" artinya bilangan basis 10 (desimal). Dengan demikian untuk bit MSB yang aktif diperoleh :

$$V_{out} = \frac{-(-5 \times 128)}{256} = 2.5 \text{ volt}$$

sesuai dengan hasil perhitungan kita.

Dengan menghilangkan tanda negatif ganda dan memasukan nilai sebagai tegangan acuan maka persamaan yang kita peroleh :

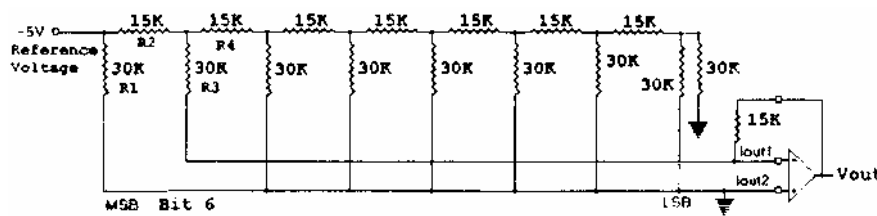
$$V_{out} = \frac{(5 \times \text{Nilai\_Digital}_{10})}{256}$$

Untuk nilai\_digital = 1 kita peroleh tegangan tiap langkah (step) :

$$V_{out} = \frac{5 \times 1}{256} = 0.1953125 \text{ volt}$$

dan ini adalah basil yang ideal dan sempurna pada hal kenyatannya tidak demikian. Umumnya tidak linear, lebih lanjut dipersilahkan membaca *data sheet* untuk diskusi lebih lanjut tentang linearitas.

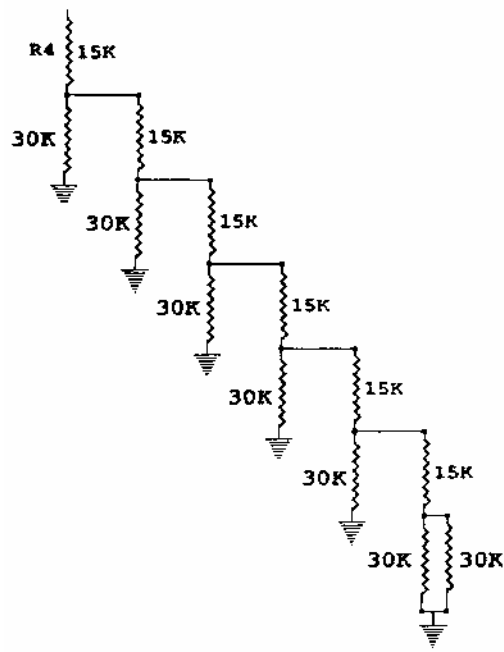
Kasus MSB yang baru saja dibahas merupakan hal yang relatif sederhana. Sekarang, misalnya, bit 6 saja yang aktif, sehingga rangkaian pada gambar 5.9 dapat disederhanakan menjadi rangkain pada gambar 5.10.



Gambar 5.9. Rangkaian tangga R 2R dengan bit 6 saja yang aktif

Arus yang melalui R1 (gambar 5.9) tidak mempengaruhi arus referensi pada masukan terinversi. Walaupun di satu ujung terhubung dengan tegangan referensi 5 volt, namun di ujung lainnya terhubung pada *ground*, sehingga arus yang melalui R1 tidak akan pernah sampai ke masukan tak-terinversi.

Resistor-resistor lainnya memberikan kontribusi terhadap arus pada masukan terinversi dan keluaran dari penguat. Arus yang melalui R2 terbagi menjadi 2, yang satu melalui R3 langsung menuju masukan terinversi sedangkan yang satunya lagi melalui R4 dan resistor-resistor lain menuju *ground*. Sekarang gambar 5.9 kita rubah sedikit, tanpa mengubah rangkaian itu sendiri seperti nampak pada gambar 5.10.



Gambar 5.10. Penyederhanaan rangkaian gambar 5.9

Pertama, perhatikan dua resistor 30K di bagian bawah, karena menggunakan konfigurasi paralel maka diperoleh :

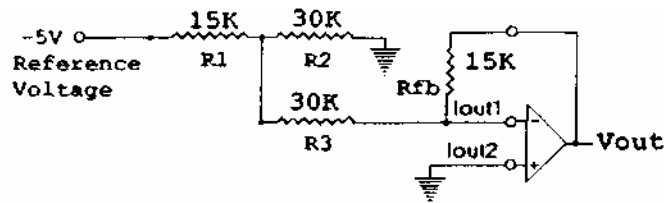
$$R_{paralel} = \frac{1}{\left(\frac{1}{30K} + \frac{1}{30K}\right)} = 15 \text{ K Ohm}$$

Hasil  $R_{paralel} = 15K$  ini diseri dengan resistor 15K menghasilkan

$$R_{Sen} = 15K + 15K = 30K$$

Nilai resistor 30K ini diparalel dengan resistor 30K disebelah kiri sehingga menghasilkan 15K lagi yang kemudian ditambahkan dengan 15K di atasnya menghasilkan 30K, demikian seterusnya. Nilai akhirnya adalah 30K Akan lebih baik jika anda mencoba membuktikan hal ini dengan corat-coret.

Hasil akhir ditunjukkan pada gambar 5.11. Nilai  $R_i$  menggantikan  $R_2$  pada gambar 5.9 dan nilai  $R_2$  menggantikan nilai  $R_4$  beserta resistor lain pada gambar 5.9 (lihat hasil perhitungan se-belumnya).



Gambar 5.11. Hasil akhir gambar 5.9

Karena masukan terinversi merupakan suatu *ground* semu, maka 2 resistor 30K seakan-akan dihubungkan secara paralel, sehingga menghasilkan 15K. Dan ini diseri dengan Ri sebesar 15K sehingga masukan inversi akan melihat resistansi totalnya 30K Ohm.

Arus yang melalui Ri adalah  $I = \frac{-5}{30K}$ , setengah bagian menuju *ground* melalui R2 dan setengahnya melalui R3 sehingga

$$I_{out1} = \frac{1}{2} I = -\frac{1}{12K} \text{ maka } V_{out} = -(15K \times (-\frac{1}{12K})) = 1.25 \text{ volt}$$

Atau, karena bit 6 bobotnya 64 maka bisa juga dihitung :

$$V_{out} = \frac{5 \times (\text{Bobot Digital})_{10}}{256} = 1.25 \text{ volt}$$

## 5.6. ADC (ANALOG-TO-DIGITAL CONVERTER)

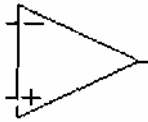
Kebalikan dari pengubah digital ke analog (DAC) adalah peng-ubah analog ke digital (ADC), yaitu suatu alat yang mampu untuk mengubah sinyal atau tegangan analog menjadi informasi digital yang nantinya akan diproses lebih lanjut dengan komputer.

Perlu dicatat bahwa data-data digital yang dihasilkan ADC hanyalah merupakan pendekatan proporsional terhadap masukan analog. Hal ini karena tidak mungkin melakukan konversi secara sempurna berkaitan dengan kenyataan bahwa informasi digital ber-ubah dalam *step-step*, sedangkan analog berubahnya secara kontinyu.

Misalnya ADC dengan resolusi 8 bit menghasilkan bilangan 0 sarapai dengan 255 (256 bilangan dan 255 *step*), dengan demikian tidak mungkin menyajikan semua kemungkinan nilai-nilai analog. Jika sekarang resolusinya menjadi 20 bit maka akan terdapat 1.048.575 *step*, semakin banyak kemungkinan nilai-nilai analog yang bisa disajikan. Penting untuk diingat, bagaimanapun juga pada sebuah *step* terdapat tak-

terhingga kemungkinan nilai-nilai analog untuk sembarang ADC yang dapat diperoleh di dunia ini. Sehingga apa yang dibuat manusia (*Human-made*) tidak akan pernah bisa menyamai kondisi dunia-nyata.

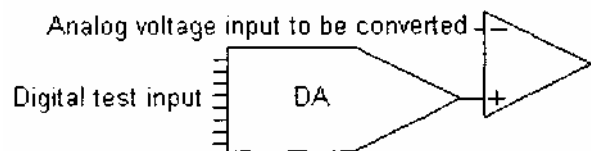
Suatu elemen yang penting dalam ADC, sebagaimana dijumpai pada ADC0809, adalah *komparator analog* yang ditunjukkan pada gambar 5.12.



Gambar5.12

Terlihat bahwa bentuknya mirip dengan Op-amp, hanya saja komparator analog ini menerima masukan analog dan menghasilkan suatu keluaran digital. Keluaran akan *HIGH* ("1") jika masukan analog arus + lebih besar dari arus -, selain itu keluarannya akan selalu *LOW* ("0").

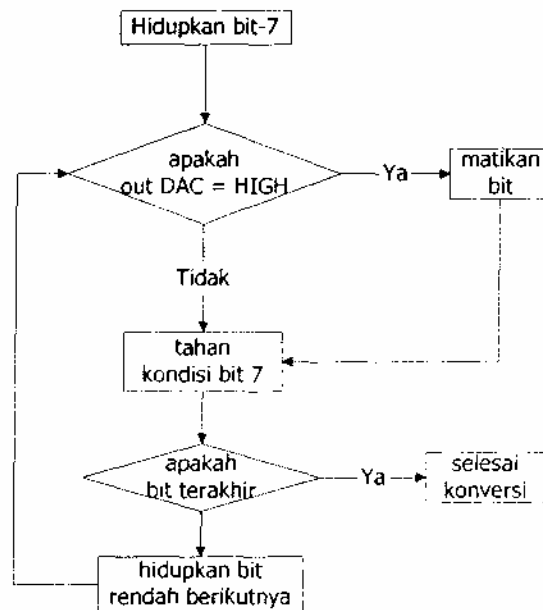
Komponen lainnya yang penting yaitu adanya DAC persis seperti yang telah dibahas sebelumnya. Masukan tegangan analog yang akan di konversi pada masukan - komparator, sedangkan keluaran dari DAC dihubungkan pada masukan + komparator. Perhatikan gambar



Gambar 5.12

Pertama kali DAC diinisialisasi dengan cara mengaktifkan bit-7 (*high order bit*) saja terlebih dahulu (jika DAC-nya 8 bit). Jika keluaran komparator adalah *LOW*, maka tegangan yang dihasilkan oleh DAC masih di bawah dari tegangan yang akan dikonversi, maka bit-7 tersebut tetap dijaga dalam kondisi *HIGH* (ON). Namun jika keluaran komparatornya adalah *HIGH*, artinya tegangan dari DAC terlalu tinggi, sehingga bit-7 di-*LOW*-kan saja. Bit-bit lainnya (dalam DAC) diuji dengan cara yang sama dan akhirnya dibiarkan *HIGH* atau dijadikan *LOW* tergantung dari status dari komparator. Proses ini dinamakan sebagai pendekatan beruntun atau *successive approximation* seperti digambarkan pada diagram alir berikut:





Sebagai contoh akan dilakukan konversi tegangan 3,21 volt. Diasumsikan bahwa konverter analog ke digital menyediakan suatu tegangan dan komparator akan membandingkan tegangan. Konverter pendekatan beruntun yang sebenarnya menggunakan arus. Dari penjelasan tentang DAC diperoleh persamaan

$$V_{out} = \frac{5 \times \text{Desimal}}{256}$$

dan kita ikuti tabel berikut

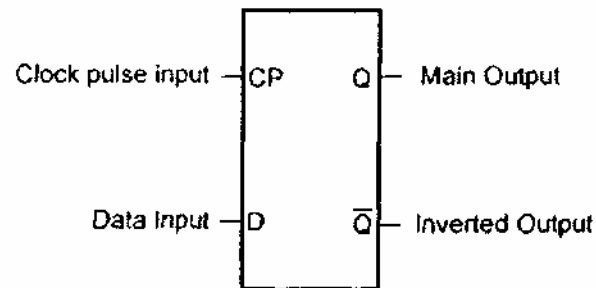
Bit Penguji	Nilai Biner DA	Desimal	Vout	Hasil Pertandingan
10000000	10000000	128	2,5	Bit ditahan
01000000	11000000	192	3,75	Bit didrop
00100000	10100000	160	3,125	Bit ditahan
00010000	10110000	176	3,4375	Bit didrop
00001000	10101000	168	3,28125	Bit didrop
00000100	10100100	164	3,203125	Bit ditahan
00000010	10100110	166	3,21421875	Bit didrop
00000001	10100101	165	3,22265625	Bit didrop

Akhirnya tiga bit dipertahankan, menghasilkan 10100100 (=164<sub>10</sub>) untuk menyajikan tegangan 3,21 volt.

ADC membutuhkan *clock* untuk bekerja, hal ini dapat di-sediakan oleh sinyal *clock* yang terdapat pada bus ISA sebesar 14,31818 MHz yang keraudian dibagi dengan 16 (akan

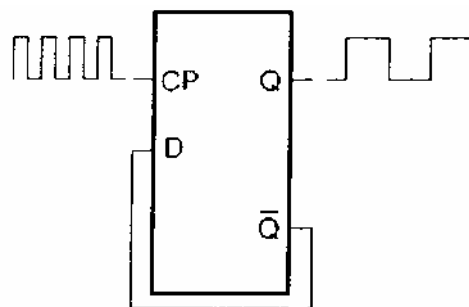
dijelaskan nanti, mengapa dibagi 16) menggunakan 74LS393 (*dual, 4-bit ripple converter*).

IC 74LS393 ini menggunakan *flip-flop* sebagai elemen dasarnya, pada gambar 5.14 ditunjukkan gambar dari satu jenis *flip-flop* yaitu *D flip-flop*.



Gambar 5.14

Jika masukan D adalah *HIGH*, kemudian ada pulsa *clock*, maka keluaran Q akan *HIGH* dan not-Q menjadi *LOW*. Sedangkan jika masukan D-nya *LOW*, maka setelah di *clock* keluaran Q akan *LOW* dan not-Q akan *HIGH*. Dengan kata lain, Q akan sama kondisinya dengan data masukan D dan not-Q akan kebalikannya setelah ada pulsa *clock*. Sekarang apa yang terjadi jika keluaran not-Q disambung langsung ke masukan D, perhatikan gambar 5.15.



Gambar 5.15

*Flip-flop* ini di *clock* pada saat sisi negatif (=falling edge = dari *HIGH* ke *LOW*), sebagaimana juga 74LS393. Ini artinya Q and not-Q hanya akan berubah saat ada transisi dari *HIGH* ke *LOW* pada masukan *clock*. Perhatikan runtunan kejadian sebagai berikut :

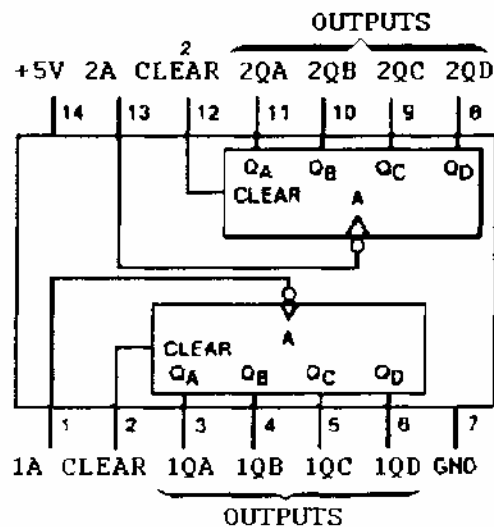
- Awalnya, misal, not-Q = *HIGH*, sehingga masukan D = *HIGH*,
- Q selalu kebalikan dengan not-Q, sehingga Q = *LOW* ;
- Terjadi *clock* pada *flip-flop*,
- Q menjadi *HIGH* (karena masukan sebelumnya *HIGH*) dan not-Q=*LOW*;

- Masukan D sekarang menjadi *LOW*;
- Terjadi *clock* pada *flip-flop*;
- Q menjadi *LOW*, dan not-Q = *HIGH*;
- demikian seterusnya.

Sehingga akan terlihat bahwa keluaran Q adalah setengah dari pulsa *clock*, dengan kata lain rangkaian pada gambar 5.15 merupakan pembagi dua atau pembagi biner. Dengan menghubungkan lebih dari 1 rangkaian seperti gambar 5.15 maka akan dihasilkan rangkaian pembagi sembarang bilangan biner (2, 4, 8, 16 dan seterusnya).

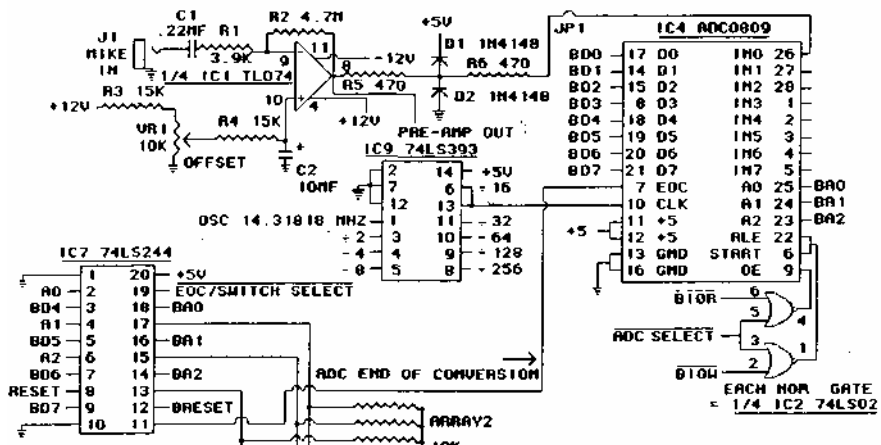
Pada gambar 5.16 ditunjukkan blok diagram dari 1C 74LS393. Masukan 1A dan 2A (masing-masing pin 1 dan 13) untuk *clock*. Masukan *clear* (pin 2 dan 12) untuk membuat agar semua keluarannya *LOW*. Jika pin 6 dan 13 (*Qu* untuk A dan masukan *clock* untuk B)

saling dihubungkan maka akan terbentuk pencacah *ripple* 8-bit dan sekaligus akan menghasilkan 8 macam pembagi (2, 4, 8, 16, 32, 64, 128 dan 256).



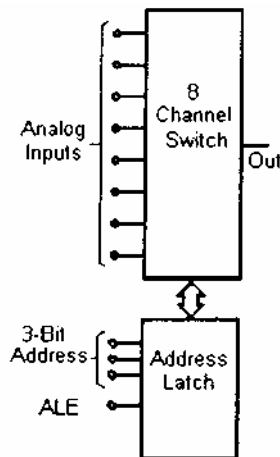
Gambar 5.16 Block Diagram 74LS393

Pada gambar 5.17 ditunjukkan suatu rangkaian antarmuka ADC 0809 dengan bus ISA. Masukan *clock* ke ADC 0809 diambil dari pin 6 dan 13 1C 74LS393. Frekuensi *clock* 14,31818 MHz jika dibagi 16 akan menghasilkan *clock* 894886,25 Hz yang membolehkan melakukan 11.000 pencuplikan data dengan tiap detik, yang juga cukup untuk merekam data dengan frekuensi hingga sekitar 5 KHz. (Ingat teori *Nyquist* pada penjabaran sebelumnya).



Gambar 5.17

Dua gerbang NOR (yang berasal dari IC 74LS02) pada rangkaian gambar 5.17 digunakan untuk menerjemahkan jalur *read select* dan *write select*. Salah satu dari 8 kanal ADC dipilih menggunakan jalur alamat tersangga (*buffered address line*) BA0, BA1, BA2. Masukan *ALE* (*Address Latch Enable*) pada ADC0809 harus *HIGH* agar konverter mengunci kanal terpilih untuk konversi, perhatikan gambar 5. 18.



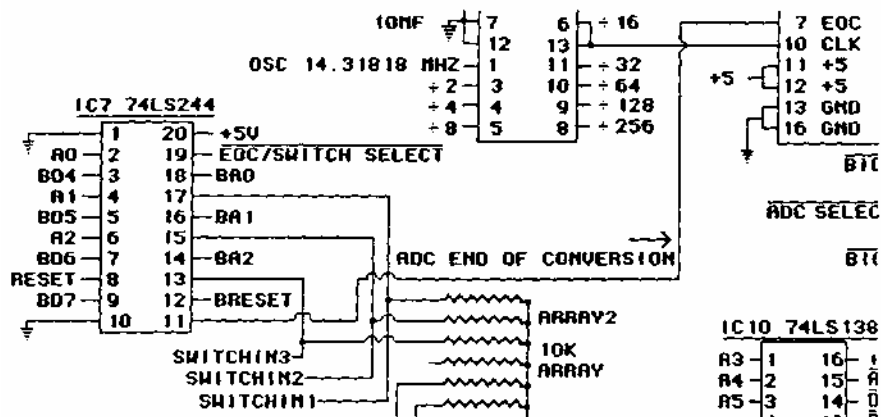
Gambar 5.18

Jalur *START* digunakan untuk memulai proses konversi. *ALE* dan *START* dihubungkan bersama untuk secara simultan mengunci kanal yang dipilih dan memulai proses konversi. Operasi tulis kanal (melalui bahasa pemrograman yang dipakai) digunakan untuk memilih kanal dan memulai konversi. *ADC select* akan bernilai *LOW* saat konverter dipilih, jika bernilai *HIGH*, artinya konverter tidak dipilih serta otomatis memaksa keluaran dari kedua gerbang NOR tersebut juga *LOW*.

Saat *ADC select* dan *BIOW* (*Buffered I/O Write- Aktif LOW*) bernilai *LOW* (aktif

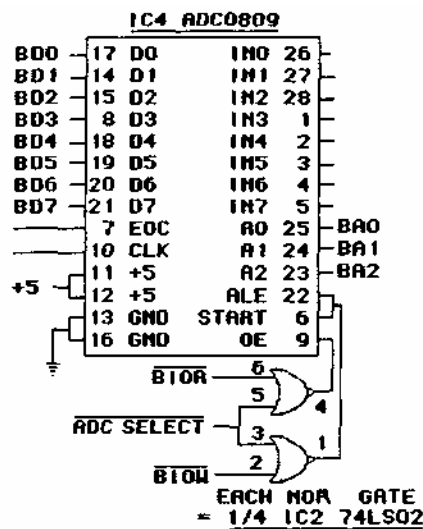
semua) maka keluaran dari gerbang yang bawah (lihat gambar 5.17) akan *HIGH* sehingga masukan ALE pada ADC0809 akan *HIGH* juga sehingga mengunci kanal yang dipilih oleh BA0, BA1 dan BA2. Pada saat yang sama, masukan *START* ADC0809 akan *HIGH* yang menyebabkan proses konversi pada kanal yang telah dipilih segera dilakukan.

Setelah selesai melakukan konversi, ADC0809 akan mengaktif-kan (*HIGH*) sinyal EOC (*End Of Conversi*) yang kemudian dapat dibaca pada BD7 melalui 74LS244, perhatikan gambar 5.19. Selanjutnya jalur OE (*Output Enable*) pada ADC0809 dapat diberi nilai *HIGH* agar hasil konversi dapat dibaca oleh komputer.



Gambar 5.19

Akhirnya OE dapat diaktifkan dengan cara mengaktifkan sinyal *ADC select* dan *BIOR* (*Buffered I/O Read*). Data kemudian ditempatkan melalui EDO sampai dengan BD7 dan siap dibaca melalui program, perhatikan gambar 5.20.



Gambar 5.20

