

## **BAB II**

### **TINJAUAN PUSTAKA DAN DASAR TEORI**

#### **2.1 Tinjauan Pustaka**

Penelitian mengenai peringatan dini akan datangnya banjir sudah banyak dilakukan. Dinas Pekerjaan Umum (PU) bekerjasama dengan Jepang melalui lembaga JICA sudah mengeluarkan panduan atau langkah – langkah pencegahan dan peringatan akan datangnya banjir, dimana berlaku tiga prinsip, yaitu (1) Prediksi, (2) Interpretasi dan (3) Respond an Pengambilan keputusan. (PU,PSNxx, 2012).

Sedangkan penelitian serupa dengan yang dikerjakan oleh penulis telah dilakukan pula oleh Duta Widya,dkk (2015) yang menggunakan sistem pakar dalam pengambilan keputusan untuk memprediksi bahaya banjir. Disamping itu ada juga yang menggunakan SMS gateway dalam menyampaikan informasi peringatan dini ke pulik, seperti yang dikemukakan oleh Aryanto, dkk (2015) dan Sumarno, dkk. (2013).

Berbeda dengan penelitian yang telah dijelaskan sebelumnya, penulis hanya memantau tinggi muka air suatu bendungan dan memberikan peringatan bahaya berupa bunyi sirine yang bunyinya berbeda untuk setiap tingkatan (*level*) bahaya. Dan untuk sistemnya penulis menggunakan mikrokontroler sebagai unit pengolah informasi, layar LCD untuk menyampaikan pesan dan Alarm sebagai bunyi peringatan.

## 2.2 Dasar Teori

Sebelum lebih jauh dibahas tentang mikrokontroler ada baiknya perlu mengetahui perbedaan antara mikrokomputer, mikroprosesor, dan mikrokontroler, mengingat istilah tersebut sering ditemui dalam kehidupan sehari-hari. Mikroprosesor adalah CPU ( *Central Processing Unit* ) dari suatu komputer, tanpa memori, I/O, dan periferal yang dibutuhkan adalah suatu sistem lengkap. Misalnya 8088 dan 8086 yang merupakan salah satu contoh mikroprosesor buatan Intel. Untuk dapat bekerja, mikroprosesor membutuhkan perangkat pendukung yang dapat berupa RAM, ROM, dan I/O.

Perangkat mikroprosesor dikombinasikan dengan I/O dan memori (RAM/ROM ), akan dihasilkan suatu mikrokomputer. Pada kenyataannya mengkombinasikan CPU dengan memori dan I/O dapat juga dilakukan dalam level chip, yang menghasilkan *Single Chip Microcomputer* ( SCM ) untuk membedakannya dengan mikrokomputer. Untuk selanjutnya SCM dapat disebut mikrokontroler.

Perbedaan menonjol antara mikrokomputer seperti IBM PC dibanding SCM adalah pada penggunaan perangkat masukan/keluaran dan juga media penyimpanan program sementara mikrokontroler menggunakan EPROM sebagai penyimpan programnya.

Keuntungan pemakaian mikrokontroler dibanding mikroprosesor adalah pada mikrokontroler sudah mempunyai RAM dan peralatan I/O pendukung sehingga tidak perlu menambahkannya.

Melihat uraian di atas, dapat di ambil kesimpulan bahwa mikrokontroler adalah salah satu bagian dasar suatu sistem komputer. Meskipun mempunyai bentuk yang lebih kecil komputer pribadi (PC), mikrokontroler di bangun atas elemen dasar yang sama.

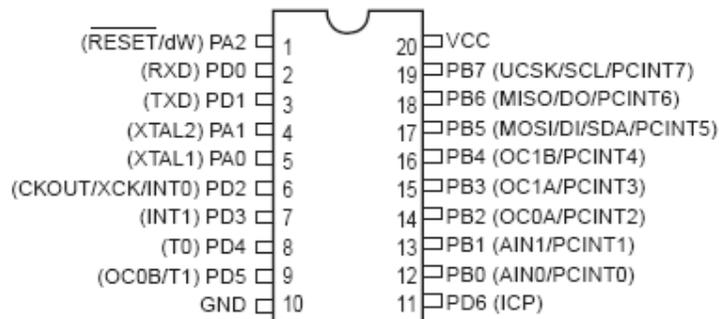
Secara sederhana komputer akan menghasilkan keluaran spesifik berdasarkan masukan yang diterima dan program yang dikerjakan. Seperti umumnya komputer, mikrokontroler adalah alat yang mengerjakan instruksi yang diberikan kepadanya, artinya bagian terpenting dan utama pada suatu sistem komputer adalah program itu sendiri yang dibuat oleh seorang pembuat program (*programmer* ). Program ini menginstruksikan komputer untuk melakukan jalinan yang panjang atas aksi sederhana untuk melakukan tugas yang lebih kompleks yang diinginkan oleh pembuat program.

### **2.2.1 Mikrokontroler**

Mikrokontroler yang digunakan dalam penelitian ini adalah mikrokontroler keluarga AVR yaitu ATtiny2313 yang merupakan mikroprosesor 8-bit CMOS dengan arsitektur RISC yang memiliki daya rendah dan unjuk kerja tinggi dengan dilengkapi 2K Bytes Downloadable Flash Memory. ATtiny2313 sering disebut sebagai *flash microcontroller* karena ROM internal yang digunakan adalah EEROM (*Electrically Erasable ROM*) Flash dengan kapasitas memori ROM 2 K Bytes (*internal*).

Keunggulan dari mikrokontroler ATtiny2313 adalah:

1. Kapasitas memori internal 2K Bytes (EEPROM)
2. 128 x 8-bit RAM internal
3. 128 x 8-bit EEPROM internal
4. 18 I/O line yang dapat diprogram
5. Timer 16 bit, 8 bit, PWM
6. Analog comparator
7. UART
8. SPI



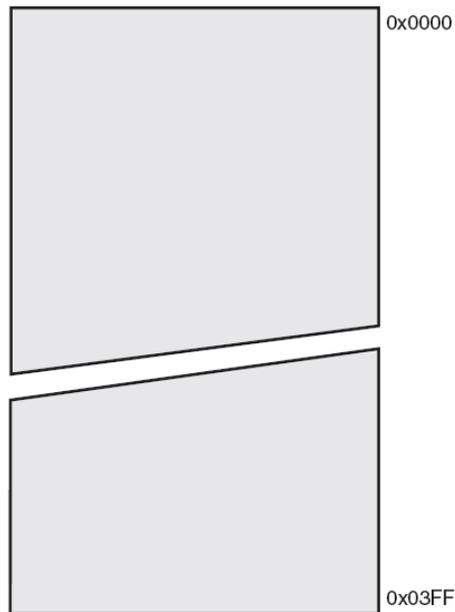
**Gambar 2.1. Konfigurasi pin ATtiny2313**

### 2.2.1.1. Organisasi memori

Program-program dan data-data pada komputer maupun mikrokontroler disimpan pada memori. Memori yang diakses oleh prosesor ini terdiri dari RAM EEPROM dan Flash ROM. Biasanya mikrokontroler mengimplementasikan pembagian ruang memori untuk data dan untuk program. Flash ROM biasanya berisi program untuk mengendalikan kerja mikrokontroler dan RAM biasanya

berisi data yang akan dieksekusi oleh mikrokontroler sedangkan EEPROM biasanya digunakan untuk menyimpan data yang sifatnya permanen.

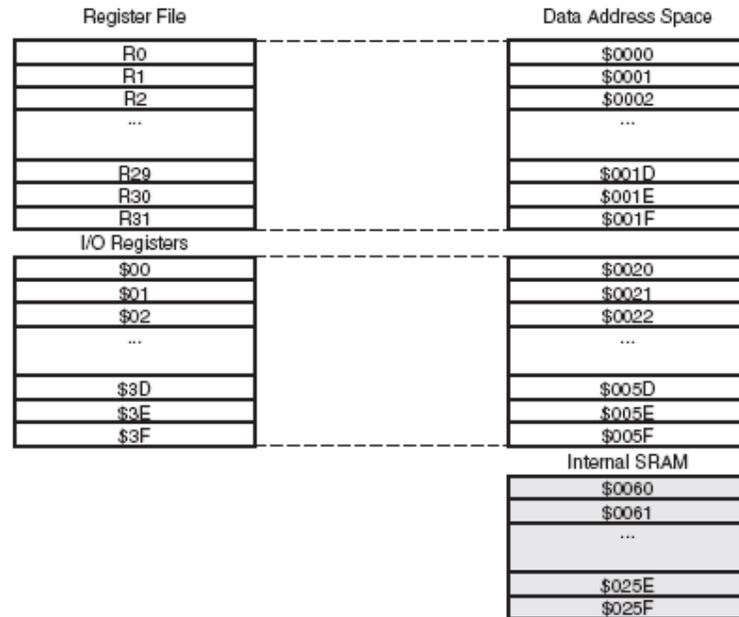
ATtiny2313 memiliki 2Kbyte Flash memori untuk menyimpan code program, karena semua perintah AVR adalah 16 bit atau 32 bit maka diorganisasikan menjadi 1K x 16. Untuk alasan keamanan flash program memori dibagi menjadi dua bagian, yaitu bagian Boot dan bagian aplikasi. Bagian boot digunakan untuk menyimpan program boot loader untuk yang biasanya digunakan untuk fasilitas update software aplikasi. Sedangkan bagian aplikasi digunakan untuk menyimpan program aplikasi yang didownload dari computer, dibawah ini adalah gambar ilustrasi pengelompokan flash memori.



Gambar 2.2 Peta memori Flash.

Data program ATtiny2313 terdapat 608 byte data yang terdiri 96 lokasi sebagai register file dan memori I/O dan sisanya 512 byte sebagai SRAM internal.

Register file R0 – R31 menempati alamat 0000H – 001FH dan I/O register menempati alamat 0020H – 005FH, sedangkan internal SRAM menempati alamat 0060H – 025FH, seperti yang diilustrasikan pada gambar berikut



Gambar 2.3 Peta memori data

### 2.2.1.2. Port masukan/ keluaran

Gerbang masukan dan keluaran atau sering dituliskan sebagai Port I/O pada ATtiny2313 memiliki 18 port I/O, yaitu Port A, Port B, Port C dan Port D yang masing-masing bersifat *bi-directional* atau dapat digunakan sebagai masukan maupun keluaran. Pengaturan sifat ini dapat dilakukan melalui register DD(*data direct*) dari masing-masing port. Disamping berfungsi sebagai data port, diantara port-port tersebut juga dapat digunakan sebagai port interupsi, port Analog in port UART maupun Port SPI, dimana fungsi-fungsi tambahan tersebut dapat dilihat pada susunan pin pada gambar 2.1 di atas, tabel 1.1 berikut

menampilkan konfigurasi dari port pada saat digunakan sebagai port input maupun port output

Tabel 2.1 Konfogurasi Pin Port

DDxn	PORTxn	PUD (in SFIOR)	I/O	Pull-up	Comment
0	0	X	Input	No	Tri-state (Hi-Z)
0	1	0	Input	Yes	Pxn will source current if ext. pulled low.
0	1	1	Input	No	Tri-state (Hi-Z)
1	0	X	Output	No	Output Low (Sink)
1	1	X	Output	No	Output High (Source)

### 2.2.2. Penampil LCD

LCD (*Liquid Crystal Display*) adalah suatu *display* dari bahan cairan kristal yang pengoperasiannya menggunakan sistem *dot matriks*. LCD yang digunakan dalam Skripsi ini adalah LCD M1632 dengan tampilan 16 x 2 baris dengan konsumsi daya rendah yang dilengkapi dengan mikrokontroler HD44780U buatan *Hitachi*. Perangkat ini sudah dilengkapi dengan fasilitas *high-contrast* dan tampilan karakter yang besar (*Wide-view*) TN LCD dan CMOS LCD.

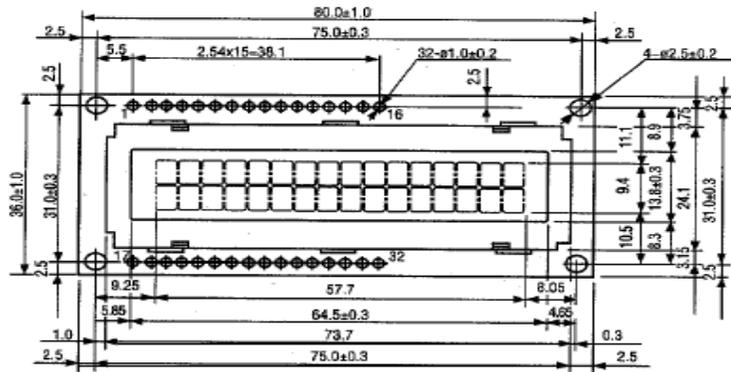
LCD M1632 memiliki tiga memori, yaitu DDRAM (*Display Data Random Access Memory*), CGRAM (*Character Generator Random Access Memory*), dan CGROM (*Character Generator Read Only Memory*). Bentuk fisik LCD M1632 ditunjukkan pada Gambar 2.4.



**Gambar 2.4. Bentuk fisik LCD M1632**

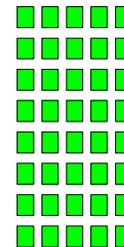
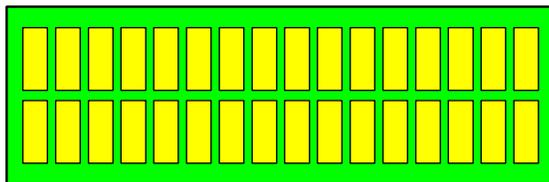
LCD (*Liquid Crystal Display*) M1632 memiliki fitur sebagai berikut:

- a. Terdiri dari 16 *segment* karakter, tiap segmen terdiri dari 5 x 7 *dot matrix* ditambah dengan kursor.
- b. *Duty ratio*: 1/16
- c. Menggunakan tegangan operasi yang rendah yaitu antara 2,7 *volt* sampai 5,5 *volt*.
- d. Sesuai untuk aplikasi yang membutuhkan kerja dengan kecepatan tinggi.
- e. LCD M1632 mempunyai 80 x 8-bit *display* sehingga mampu menulis 80 karakter maksimal (ditunjukkan pada Gambar 2.6).
- f. Banyak fungsi instruksi yang dapat digunakan seperti: *Display clear*, *Cursor home*, *Display ON/OFF*, *Cursor ON/OFF*, *Display character Blink*, *Cursor Shift*, dan *Display Shif*.
- g. Karena piranti ini didukung oleh mikrokontrol HD44780U maka piranti ini *support* dengan mikrokontrol HD44780S.
- h. Perangkat ini dilengkapi pula dengan *outomatic reset* sehingga keadaan akan selalu *defualt* pada saat LCD dinyalakan.
- i. Instruksi yang digunakan adalah *standart CMOS*.



Dimensi Diagram LCD

Tiap *segment* terdiri dari 5x8 *dot*



**Gambar 2.5. Dimensi LCD M1632**

**2.2.2.1 Display data random access memory**

DDRAM merupakan memori tempat karakter yang ditampilkan berbeda. Contoh, untuk karakter 'A' ditulis pada alamat 00, maka karakter tersebut akan tampil pada baris pertama dan kolom pertama dari LCD. Apabila karakter tersebut ditulis pada alamat 40, maka karakter tersebut akan tampil pada baris kedua kolom pertama dari LCD. Seperti pada Gambar 2.6.

<i>Display position</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
<i>DDRAM address</i>	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F

**Gambar 2.6. Alamat DDRAM M1632**

#### **2.2.2.2. Character generator random access memory**

CGRAM merupakan memori untuk menggambarkan pola sebuah karakter dimana bentuk dari karakter dapat diubah-ubah sesuai keinginan, tetapi memori ini akan hilang saat catu daya dimatikan/ *power supply* tidak aktif, sehingga pola karakter akan hilang.

#### **2.2.2.3. Character generator read only memory**

CGROM merupakan memori untuk menggambarkan pola sebuah karakter dimana pola tersebut sudah ditentukan secara permanen dari HD44780U sehingga pengguna tidak dapat mengubah lagi. Tetapi karena ROM bersifat permanen, maka pola karakter tersebut tidak akan hilang walaupun *power supply* tidak aktif.

Sebagai contoh pada saat HD44780U akan menampilkan data 41H yang tersimpan pada DDRAM, maka HD44780U akan mengambil data dialamat 41H (01000001b) yang ada pada CGROM yaitu pola karakter 'A'.

#### **2.2.2.4. Konfigurasi PIN**

LCD M1632 memiliki 14 *PIN* dengan fungsi yang berbeda-beda. Terdapat 3 *PIN* sebagai *bit* kontrol yaitu E sebagai *input clock*, R/W sebagai *input* untuk memilih *read* atau *write* dari RS sebagai *register select*, 8 *PIN* sebagai *bit* data yaitu DB0 sampai DB7 dan dua *pin* lainnya adalah VCC dan *Ground*. Konfigurasi *pin* LCD M1632 ditunjukkan pada Tabel 2.2.

**Tabel 2.2. Konfigurasi *pin* LCD M1632**

No	Nama <i>pin</i>	Keterangan
1	GND	0
2	VCC	+ 5 V
3	VEE	Tegangan kontras LCD
4	RS	<i>Register Select</i> , 0 = <i>Register Perintah</i> , 1 = <i>Register Data</i>
5	R/W	1 = <i>Read</i> , 0 = <i>Write</i>
6	E	<i>Enable Clock LCD</i> , Logika 1 setiap kali pengiriman atau pembacaan data
7	D0	Data Bus 0
8	D1	Data Bus 1
9	D2	Data Bus 2
10	D3	Data Bus 3
11	D4	Data Bus 4
12	D5	Data Bus 5
13	D6	Data Bus 6
14	D7	Data Bus 7
15	Led +	+ 5 V
16	Led -	0

#### **2.2.2.5. Register**

HD44780U mempunyai dua buah *register* yang aksesnya diatur dengan menggunakan kaki RS. Pada saat RS berlogika '0', maka *register* yang akan diakses adalah *register* perintah dan pada saat RS berlogika 1, maka *register* yang diakses adalah *Register* data.

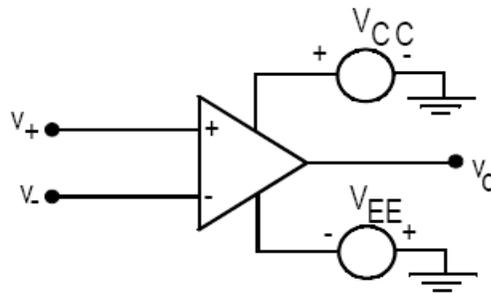
1. *Register* perintah ini adalah *register* dimana perintah dari mikrokontroler ke HD44780U pada saat proses penulisan data atau tempat status dari HD44780U dapat dibaca pada saat pembacaan data.
2. *Register* data ini adalah *register* dimana perintah mikrokontroler dapat menuliskan atau membaca data ke atau dari DDRAM. Penulisan data pada *register* ini akan menempatkan data tersebut ke DDRAM sesuai dengan alamat yang telah diatur sebelumnya.

Penulisan data ke *register* perintah dilakukan dengan tujuan mengatur tampilan LCD, inisialisasi dan mengatur *Address Counter* ataupun *Address Data*. Kondisi RS berlogika 0 menunjukkan akses data ke *register* perintah. RW berlogika 0 yang menunjukkan proses penulisan data. Sedangkan penulisan data pada *register* data dilakukan untuk mengirimkan data yang akan ditampilkan pada LCD. Proses diawali dengan adanya logika 1 pada RS yang menunjukkan akses ke *register* data, kondisi R/W diatur pada logika 0 yang menunjukkan proses penulisan data. Pengiriman data dari data *Bus* DB0 – DB7 diawali dengan pemberian pulsa logika 1 pada E *clock*. Dan akhiri dengan *pulsa* logika 0 pada E *clock*.

### 2.2.3. Penguat Operasional (*Operational Amplifier*)

#### 1. Dasar-dasar Penguat Operasional

Penguat operasional atau *Operational Amplifier (Op-Amp)* adalah suatu blok penguat yang mempunyai dua masukan dan satu keluaran. Op-amp biasa terdapat di pasaran berupa rangkaian terpadu (*integrated circuit-IC*).



**Gambar 2.7 Rangkaian dasar penguat operasional.**

Gambar 2.7 menunjukkan sebuah blok op-amp yang mempunyai berbagai tipe dalam bentuk IC. Dalam bentuk paket praktis IC seperti tipe 741 hanya berharga beberapa ribu rupiah. Seperti terlihat pada Gambar 2.6, op-amp memiliki masukan tak membalik  $v_+$  (*non-inverting*), masukan membalik  $v_-$  (*inverting*) dan keluaran  $v_o$ . Jika isyarat masukan dihubungkan dengan masukan membalik ( $v_-$ ), maka pada daerah frekwensi tengah isyarat keluaran akan “berlawanan fasa” (berlawanan tanda dengan isyarat masukan). Sebaliknya jika isyarat masukan dihubungkan dengan masukan tak membalik ( $v_+$ ), maka isyarat keluaran akan “sefasa”. Sebuah op-amp

biasanya memerlukan catu daya  $\pm 15$  V. Dalam menggambarkan rangkaian hubungan catu daya ini biasanya dihilangkan.

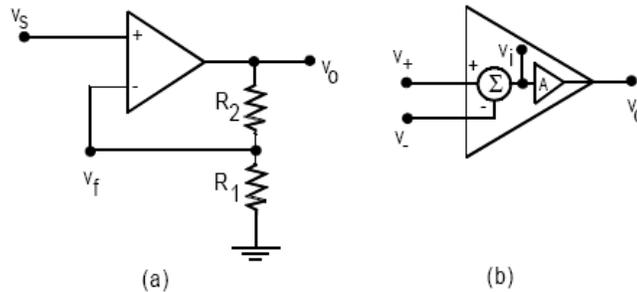
Idealnya, jika kedua masukan besarnya sama, maka keluarannya akan berharga nol dan tidak tergantung adanya perubahan sumber daya, yaitu :

$$V_o = A (v_+ - v_-)$$

dimana  $A$  berharga sangat besar dan tidak tergantung besarnya beban luar yang terpasang.

## 2 . Penguatan Tak-Membalik (*Non-Inverting Amplification*)

Op-amp dapat dipasang sebagai penguat tak membalik seperti Gambar 2.16. Terlihat bahwa masukan diberikan pada  $v_+$ .



**Gambar 2.8 Rangkaian penguat operasional tak membalik.**

Op-amp tersebut berfungsi sebagai

$$v_o = A (v_+ - v_-)$$

Dan selanjutnya kita dapat menuliskan untuk penjumlahan ( $\Sigma$ ) dan penguat ujung tunggal ( $A$ ) seperti pada Gambar 2.8.b.

$$v_i = v_+ - v_-$$

$$v_o = A v_i \text{ atau } v_o / v_s = 1 + (R_2 / R_1)$$

Penguat tak-membalik dapat juga dibuat menjadi bentuk khusus secara sederhana seperti ditunjukkan pada Gambar 2.17.

### **Gambar 2.9 bentuk khusus penguat operasional tak membalik.**

Basarnya tegangan keluaran adalah :

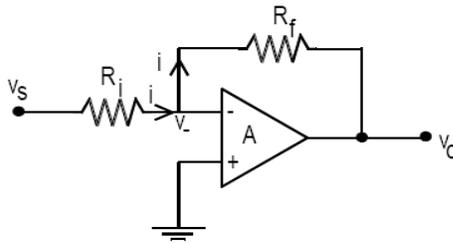
$$v_o \approx v_s$$

$$v_o / v_s = 1$$

Penguat seperti Gambar 2.8 menghasilkan penguatan + 1. Rangkaian ini sangat menguntungkan karena dapat memperoleh suatu penguat dengan hambatan masukan yang sangat tinggi ( $10 - 10^{12} \Omega$ ) dengan hambatan keluaran sangat rendah ( $10^{-3} - 10^{-1} \Omega$ ), yaitu mendekati kondisi ideal. Rangkaian ini disebut rangkaian pengikut (*follower*), suatu bentuk peningkatan dari penguat pengikut emitor. Jadi penguat ini berfungsi sebagai penyangga (*buffer*) dengan penguatan = 1.

### 3 . Penguatan Membalik (*Inverting Amplification*)

Pada penguat membalik, sumber isyarat dihubungkan dengan masukan membalik sedangkan masukan positif ditanahkan seperti terlihat pada Gambar 2.10.



**Gambar 2.10 Rangkaian penguat operasional membalik.**

Dari Gambar 2.10 terlihat bahwa sebagian dari keluaran diumpungkan kembali ke masukan melalui  $R_f$ . Penguat ini termasuk penguat pembalik negatif. Penguatan dari rangkaian ini dapat ditentukan sebagai berikut. Kita berasumsi bahwa arus  $i$  tidak melalui masukan, jadi arus  $i$  melewati  $R_i$  dan  $R_f$ . Maka,

$$v_s - v_- = i R_i$$

$$v_- - v_o = i R_f$$

$$v_o = -Av$$

Biasanya  $A$  berharga sangat besar sehingga  $v_o / A$  berharga sangat kecil dibandingkan dengan  $v_o$  dan  $v_s$ . Kita dapatkan penguatan lingkaran tertutup sehingga

$$v_o / v_s \approx -R_f / R_i$$

Ternyata *secara* sederhana dasarnya penguatan hanya merupakan perbandingan kedua hambatan yang dipasang.