CHAPTER IV

COMPUTER INTERFACING

In the present day, the microelectonic revolution has given low cost, very high performance computing power to anyone. By the aids of powerful microcomputers, the radio astronomers can use them to control observation, gather and reduce data. The more advantage of the computer control system is not only the data recorder as another readout device, but also the ability to process the gathered data and to recover some failure occurred at some part of the signal recording.

For the radio astronomy, it give a chance to calculated the physical quantities such as brightness and temperature of the source more effective than the other readout devices. By the very fast sampling system, the data can be gathered and integrated at any level of time constant which can be give the optimum for the data. By aids of numerical method, the digital filter and the drift correction can be introduced to the long-term calibrated records.

This chapter provide the importance principles behind the described system. We start with the general description about the IBM PC family, its bus system and the interrupt process. The next section is involve with the digital decoder circuits which activate a specific part of control system to avoid the ambiguous of the bus occupation. The time base and basic I/O interface will also be considered. we will proceed with the considerations of the digital converters which plays the important

role to interchange between the analog and the digital part of the measured system. The next part is dedicated to the very useful operational amplifier which applications can be seen in various way. Since we use the sampling process for our acquisition system, the effect of the sampling process to the output signal should be considered and the famous sampling theorem is involved. From this theorem, it bring us to be interested in the filters both analog and digital. In the last part of this chapter, we concentrate in some software consideration for control the system.

IBM PC Architecture

This section provide the brief discussion about the architecture of the IBM personal computer family which is mostly common used and available in Thailand than other computer system.

IBM was first announced the original PC microcomputer in 1981 (Norton, 1995|). The system used an Intel 8088 microprocessor running at 4.77 MHz. The processor operate with the 16-bit arithmetic. Since the available electronic components in that day are mostly support the 8- bit equipment. The bus system was designed to facilitate 8-bit interface. In 1984, IBM release the AT version with the fully 16-bit bus with 24-bit addressing. Further, the newer version of the personal computer was released in the later, the architecture of the latter version is the extension of the IBM PC/AT. Since the flexibility and accessibility of technology, there are many OEM which construct the compatible system. The compatible version of the PC's may cause the problem for interfacing.

One of the most advantage of the PC architecture are the supporting of external interfacing. It give a chance for communicate to the other extension devices. By the use of extended cards which are available from many manufactures, the computer system can be connected to the appropriate equipment, such as various type of video monitor, keyboard, etc. Since the accessibility of the information, it is possible for anyone to design and built the own system extending from the PC's for the specific purposes.
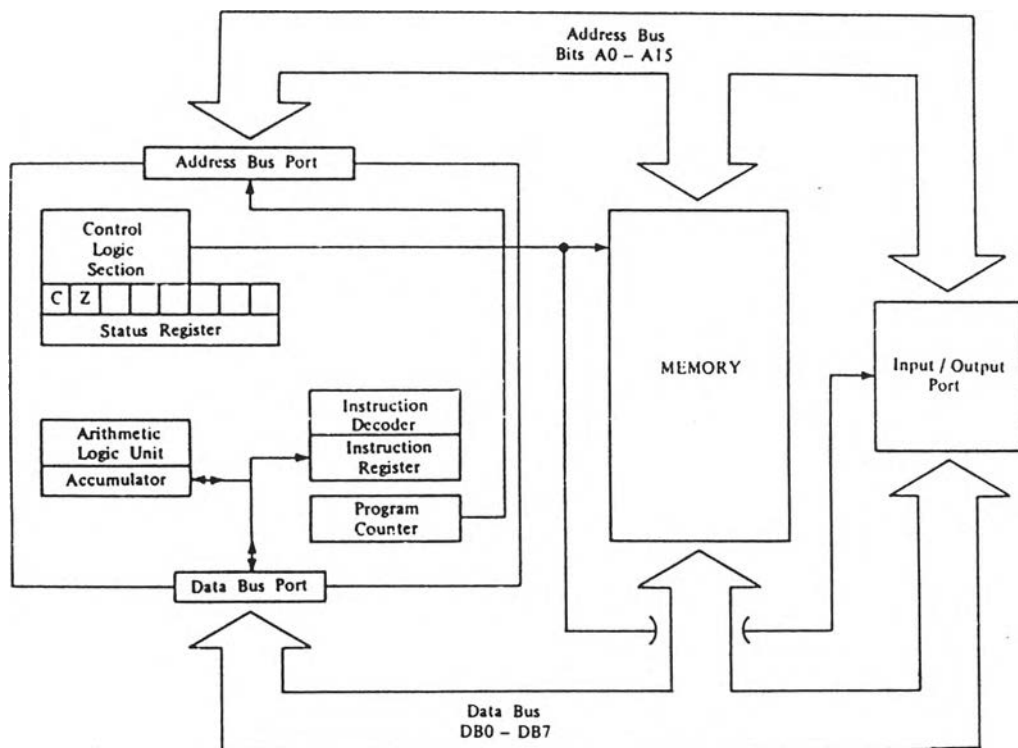

Fig. 4.1 Block diagram for typical computer system (Carr, 1991).

The block diagram for a typical computer can be shown simply in the Fig 4.1 . The computer can basically be divide into three parts which are *Central processing Unit (CPU), memory* and *communication port*. The function of the CPU is the control the hardware operation and manipulate the basic mathematical operation. The memory

is used to storage the data and the instructions for control overall operations and the communication ports provide the interfacing to the external world.

All parts use the *bus system* to interchange information each other . We can classified the bus in three types. The *data bus* convey the data and instruction. The *address bus* determined the source and destination of data transfer. The both are governed by many control wire which are called collectively as the *control bus*. The control bus provide the sufficient signals for control the basic operation such as read the data from a specific memory address.

The data bus is commonly used by various devices, so that the ambiguous may be occurred. To avoid this situation, the measure to identify the proper device have to be made. The virtual area where the data are transferred to a specific device is called *port* . Hence, any devices which are attached to the data bus have to occupy the specific port number except memory which is separated by the special control signal from the CPU. The specific port number can be determined by the circuit called the *decoder* which detail consideration will be made in the next section. However, the block diagram as seen by Fig 4.1 is common for any system, but the CPU 80XXX family is also support another functions which are called *interrupt* and *Direct memory Access (DMA)* .

It can be seen that the typical system shown in Fig 4.1, the CPU have to monitor all device regularly. It is not the effective system since the CPU run time have to share to all devices. For the most device, the monitor by CPU regularly is not critical, i.e. the disk drives transfer the data when the diskettes is read or written, the keyboard will be operated only when the keys are pressed. If there are many such

devices are connected to the computer system, the useful time have to be sacrificed. It is more effective that all devices are monitored in the proper time when that device sent the request to CPU. This process is called the interrupt. The interrupt signal will occur only when the request from the specific device is occur. The CPU stop the current process and then jump to the address contained the set of instructions for service the device. After complete the instructions, the CPU return to the old point and success operation will be done. The devices can operate by itself and request the CPU when the data transfer is require and the CPU has more time to operate another instruction.

The interrupt request for a device have an individual number. The interrupt request number are connected to the specific address contained to the interrupt service routine. The address number resided by the service routine is called *the interrupt vector*. The CPU indicate the interrupt vector as look up the table contained the vector information. The interrupt vector table was assigned reside at the lowest part of the memory map, say address o, in the PC system. When the CPU have the request from the specific device. The interrupt request number is decoded to the position in the interrupt vector table which coincide with the IRQ number.

Them interrupt of this type is called the *hardware interrupt* which is distinguished from the *software interrupt*. The hardware interrupt can be divided in two categories, *maskable* and *non-maskable*. The non-maskable hardware interrupt is generated when the error occur in device such as divide by zero. It occur with no measure can be recover, in contrast to the maskable one which may be set to be discarded by reset the interrupt flag in register flag of CPU. The IRQ number have

priority . The high priority number will be treat if the two interrupt signal occur in the same time. From the PC AT technical reference, each of IRQ numbers is assigned for various devices as seen Table 4.1 from high priority. When the assigned devices absent the interrupt number may be used in designed system.

Interrupt CTLR 1	Interrupt CTLR 2	Processor Interrupt No. [Hex]	Function
IRQ0		08	Timer Output 0
IRQ1		09	Keyboard (Output Buffer Full)
IRQ2		0A	Realtime Clock Interrupt
	IRQ8	71	Software Redirected to INT 0AH (IRQ2)
	IRQ9	72	Reserved
	IRQ10	73	Reserved
	IRQ11	74	Reserved
	IRQ12	75	Coprocessor
	IRQ13	76	Fixed Disk Controller
	IRQ14	77	Reserved
	IRQ15	78	Reserved
IRQ3		0B	Serial Port 2
IRQ4		0C	Serial Port 1
IRQ5		0D	Parallel Port 2
IRQ6		0E	Diskette Controller
IRQ7		0F	Parallel Port 1

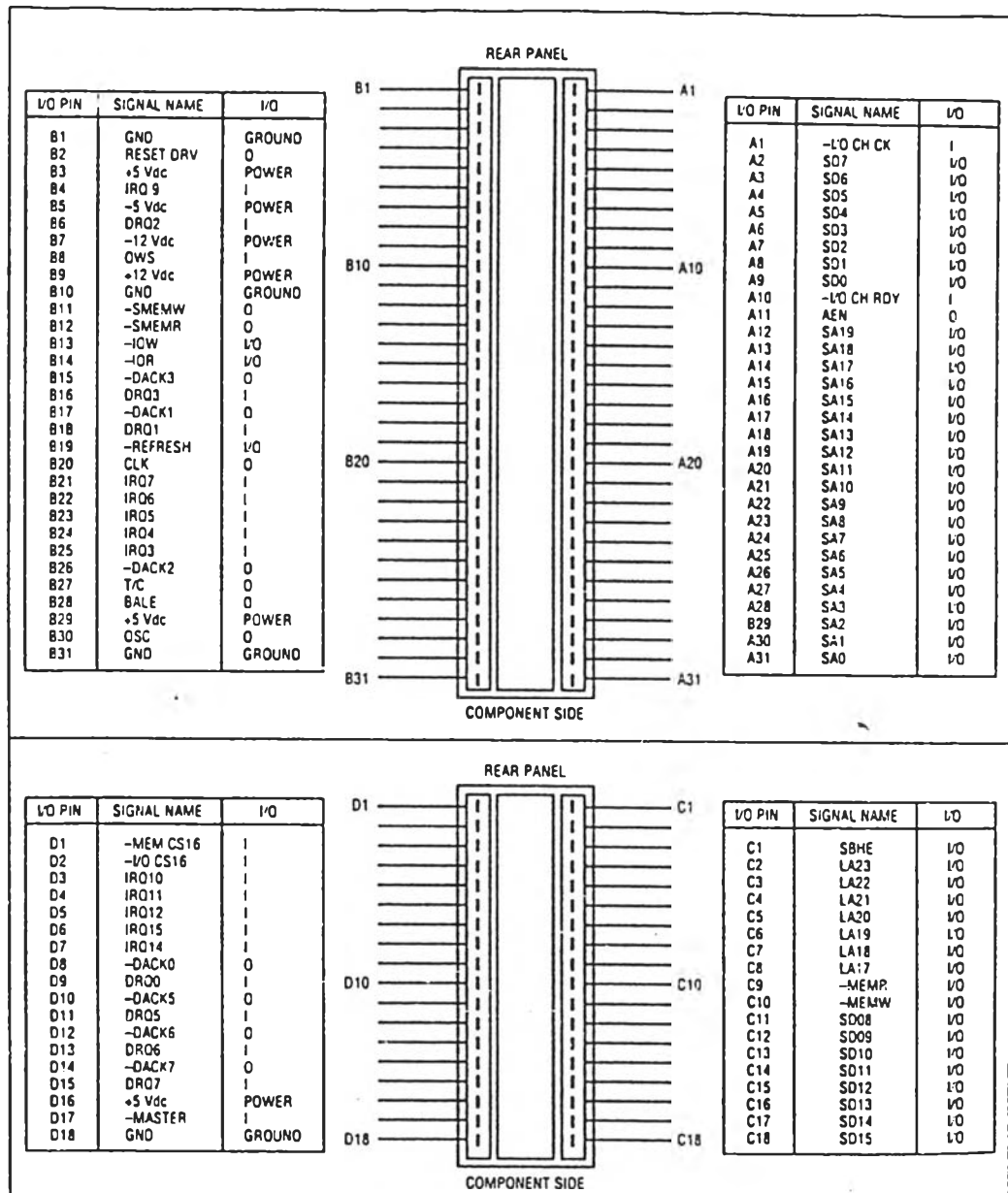Table 4.1   Interrupt assignments in IBM AT (ไพศาล, 2534 ; IBM, 1984).

In the PC's system, the 256 interrupt vector are available the interrupts are not only hardware interrupt but also the most is software interrupt. The software interrupt is similar to the hardware interrupt, but request signal is replaced by the instruction INT in the assembly language. Some software interrupts are dedicated to the *Basic*

Input and Output Service (BIOS) which are the low level such as read or write the hard disk. However, there are available for the user to define the new routine for some purpose. The user-defined routines have to be implemented by a specific way. The advantage of the user-defined routine can be impressed by the *TSR (terminate and stay resident)* programs such as Norton commander.

For the device which need the high speed for transfer data, it is effective to transfer data directly to the memory without the transfer process managed by the CPU. The process is called the *Direct Memory Accesses (DMA)* which is used by the disk drive controller. However, this topic is beyond the scope of the thesis, it is not considered in more detail.

Digital Decoder and Buffer Circuit

Since many devices commonly use the data bus, the process by which the CPU can transfer the data to and from a specific devices is called the *decoder*. The specific device have an individual port number. The CPU will send the number of the port which is the source or destination of data transfer via the address bus. However, every devices are received the same address which specific port but only the device which can decode the port number will response. Since the address bus also commonly use with the memory. The control signal for separating between memory and other devices have issued from the CPU. To determine what direction of data transfer, the CPU have also to send the signal "write" or "read" to the device, in the external slot.

I/O PIN	SIGNAL NAME	I/O
B1	GND	GROUND
B2	RESET DRV	O
B3	+5 Vdc	POWER
B4	IRQ 9	I
B5	-5 Vdc	POWER
B6	DRQ2	I
B7	-12 Vdc	POWER
B8	OWS	I
B9	+12 Vdc	POWER
B10	GND	GROUND
B11	-SMEMW	O
B12	-SMEMR	O
B13	-IOW	I/O
B14	-IOR	I/O
B15	-DACK3	O
B16	DRQ3	I
B17	-DACK1	O
B18	DRQ1	I
B19	-REFRESH	I/O
B20	CLK	O
B21	IRQ7	I
B22	IRQ6	I
B23	IRQ5	I
B24	IRQ4	I
B25	IRQ3	I
B26	-DACK2	O
B27	T/C	O
B28	BALE	O
B29	+5 Vdc	POWER
B30	OSC	O
B31	GND	GROUND

I/O PIN	SIGNAL NAME	I/O
A1	-I/O CH CK	I
A2	SD7	I/O
A3	SD6	I/O
A4	SD5	I/O
A5	SD4	I/O
A6	SD3	I/O
A7	SD2	I/O
A8	SD1	I/O
A9	SD0	I/O
A10	-I/O CH RDY	I
A11	AEN	O
A12	SA19	I/O
A13	SA18	I/O
A14	SA17	I/O
A15	SA16	I/O
A16	SA15	I/O
A17	SA14	I/O
A18	SA13	I/O
A19	SA12	I/O
A20	SA11	I/O
A21	SA10	I/O
A22	SA9	I/O
A23	SA8	I/O
A24	SA7	I/O
A25	SA6	I/O
A26	SA5	I/O
A27	SA4	I/O
A28	SA3	I/O
B29	SA2	I/O
A30	SA1	I/O
A31	SA0	I/O

I/O PIN	SIGNAL NAME	I/O
D1	-MEM CS16	I
D2	-I/O CS16	I
D3	IRQ10	I
D4	IRQ11	I
D5	IRQ12	I
D6	IRQ15	I
D7	IRQ14	I
D8	-DACK0	O
D9	DRQ0	I
D10	-DACK5	O
D11	DRQ5	I
D12	-DACK6	O
D13	DRQ6	I
D14	-DACK7	O
D15	DRQ7	I
D16	+5 Vdc	POWER
D17	-MASTER	I
D18	GND	GROUND

I/O PIN	SIGNAL NAME	I/O
C1	SBHE	I/O
C2	LA23	I/O
C3	LA22	I/O
C4	LA21	I/O
C5	LA20	I/O
C6	LA19	I/O
C7	LA18	I/O
C8	LA17	I/O
C9	-MEMR	I/O
C10	-MEMW	I/O
C11	SD08	I/O
C12	SD09	I/O
C13	SD10	I/O
C14	SD11	I/O
C15	SD12	I/O
C16	SD13	I/O
C17	SD14	I/O
C18	SD15	I/O

PC AT 16-bit extension slot.

PC AT Expansion Bus slots signal/pin definitions

SD0-SD15 - System Data Bits 0 through 15
SA0-SA23 - System Address Bus bits 0 Through 24
IRQ2-IRQ15 - Interrupt Requests Levels 2 through 15
DRQ0-DRQ7 - Direct Memory Access Requests 0 thru 7
DACK0-DACK7 - Direct Memory Access Acknowledge 0 thru 7

Note: The remaining signals provide Bus timing information, status and commands. For a detailed
description of these signals consult Interfacing to the PC (SAMS 1995) or the IBM Technical Reference Manuals.

Fig. 4.2 PC AT 16-bit extension slot (Norton, 1995).

In PC system, all control signal lines are provided with the data and address buses according to the Industry Standard Architecture (ISA). In the older PC and PC XT version, the 8-bit data bus and 20-bit address bus was considerably sufficient. All buses was arranged in the longer slot in the external PC AT main board as shown in Fig. 4.2. When the AT version is introduced the full 16-bit data transfer, the additional signals are required. For the computability to the previous version, the extended data bus was designed to be accessible only when the CPU request the 16-bit data transfer and the device required 16-bit interface.

The normal stage of the most signal in the extension slot are high impedance. If the stage become to "low" the function of the control signals are accomplished. Hence the most control signal are indicated by the minus or the line above the signal name to distinguish from the other which activate by the high signal.

Since logically low is used to activate the specific function, if this signal lines are disconnected, the function remain the normal stage. For example, the input signal *I/O 16-bit chip select (-IO CS16)* in the PC's ISA bus have to be sense logical "low" from the device when require the 16-bit interfacing. In the situation of the older 8-bit extension card was plug-in the AT slot, the 8-bit interfacing remain accessible since the -IO CS16 leave to the high impedance stage although the any no connection is made.

The decoder circuit may be simply *gates* but for the complex port decoder which involve the many lines of the address bus the more complex decoder IC such as 74LS139 may be applied. For example, the 74LS139 have dual decoder/

demulitiplexer. Each one is work independently. Each one has 3 input and 4 output which has the truth table as seen in the Table 4.2.

INPUT			OUTPUT			
$\overline{E}$	A_o	A_1	0	1	2	3
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

Table 4.2 Function table for 74LS139.

We can see that the outputs remain "high" or "1" in binary notation until the pin $\overline{E}$ (mean "enable") is activated by the logically "low" or "0" in binary notation. The outputs are depended on the logical state of the input A_o and A_1. The only one ouput which is corresponding to the binary number become "low". If the inputs are connected to the address bus and the output of 74LS139 decoder are connected to difference devices which are activated when the "low" pulse, the only one device will be operated. The number which is assigned on the address bus is the port number of the operating device as seen in Table. 4.2.

There are the another remaining problem for the common used of buses. When many devices are connected to the system bus, the source current in the each line of

bus can deliver theoretically to the infinite component but it is impossible in practical world. The output of the common gate can deliver the current to a few ICs. From this reason, the buffer should be used in order to prevent the shortage of current in the system.

The buffer usually have the open-collector architecture which can deliver the far more current than normal digital IC to the connected component. the most available buffer IC is the three-state buffer. The three-state buffer refer to the buffer which have three possible logic state, i.e. high, low and high impedance. The high impedance state is the normal stage, say no current is delivered to the output. It may be considered as it is switch off. The logic high and low will be occurred at the outputs only when the enable pin is activated such as 74LS139. The advantage of the three-state buffer is the ability to disconnect the circuits at the input and output of the buffer and to connect them together when they expected to working together.

The two buffer can be arranged to transfer in bi-directional, i.e. send and receive. The bi-directional arrangement of many buffer with the same package can be used in the data bus and the address of the interfacing system.

Counter and Timer

In the digital system, some part of the circuit have to be control by signal pulse. The pulse may by generated by the decoder circuit which occur when a certain condition is met. In some situation, the periodic pulse have to be generated, such as in the digital converter circuit. We generate the sampling pulse to ADC in the time we

want to measure the signal, and to DAC when we reconstruct the sampling data to analog signal in the CD player.

It is possible to generate the periodic signal from on oscillator circuit. In the digital system, the square wave can be used to be the time base signal. The precise square wave can be generated by the crystal oscillator circuit as shown in Fig. 4.3 .
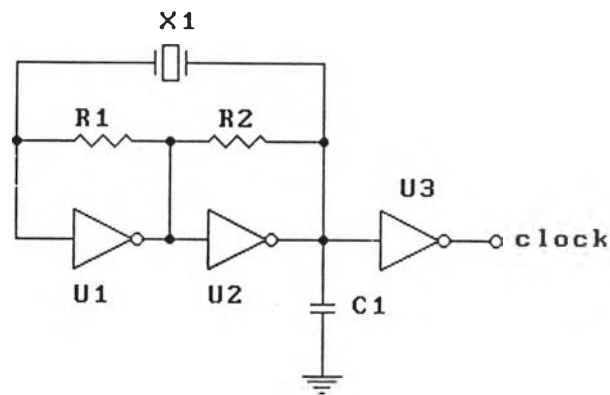
Fig 4.3 Crystal oscillator.

The period of the square wave determined by the resonant frequency of the crystal X_1 . The oscillation occur because the NOT GATES U_1 and U_2 are feed back by R_1 and R_2 respectively.

Since it need a small time for a not gate to invert the signal from input to its output. The logic state change to invert state at the output. The input of the gate response such the logic state from the output itself and invert the logic state again. Thus the square wave will be generated to the output. If we use only the feed-back scheme to the not gate, the period of the square wave is depend on the delay time for inverting of the gate. By adding the crystal X_1 , the period of the square wave will be defined since the Crystal will oscillate at the resonant frequency. The capacitor C_1 and

the last gate U_3 is use to shaping the output signal and operate like a buffer to contribute the more maximum current deliver to the next attached current.

The crystal oscillator circuit as shown in Fig. 4.4 is fixed frequency. The more versatile circuit should be made for varying the frequency of the pulse. The idea behind such circuit is the counting.

We can use the circuit as shown in Fig. 4.4 to generate the time base signal, each waveform of the square wave called *clock*. If the number of the clock is counted and the change of the digital state in an digital component depend on the number of the clock, which is generated by the time base circuit. That component can be use as the *counter* and the output signal can be used to generate the frequency variation signal. For example, The T flip-flop will change the current state of the output at the ' high to low' edge of the clock, i.e. two clocks will generate one period of the output signal as seen in the Fig 4.4. By the way the frequency of the square wave at the input is divided by two.
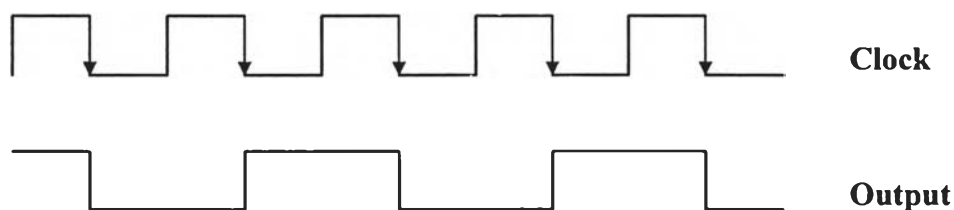
Fig. 4.4 Timing diagram for T Flip-Flop

For a digital component like T Flip-flop, but the n clocks used to complete a period of square wave, the frequency of the square wave can be vary by change the value of n. There are many IC which serve as the purpose such as the *8253*

programmable interval timer which is one of the chip set for the IBM PC system. The number n can be preliminary defined by sent the two consecutive bytes to a counter in 8253. Each byte of the data represent the least significant byte and the most significant byte in the 16-bit data format, respectively. When the start signal is sent, the number n is decreased by the 8253 until the number of n becomes zero and the output signal will be occur which depend on the operation mode which is set (see 8253 data sheet). For the negative pulse, the mode 2 (rate generator) can be assigned to the 8253.

Basic I/O Interface

As we can read from the previous section, the computer can communicate with the external device via the port of this device. The basic operation is read from or write to the assigned port. In the output case, the data send from PC will be disappear after the sending process is finished.

If we want the data at an instant time remain at the input terminal of the extended device, we may use *latch*. The latch will be operated when the latch enable signal is sent. The data will transparent to the output when the latch enable hold at a certain logic state. When the transition occur, the data present one set-up time before the transition is stored until the enable signal occur again. For example, the 74LS373 allowed the data from 8-bit input transfer to output directly when the pin E is high state. When the pin E go to low, the data is saved until the latch enable is set again.

Another way to hold the output signal is using the D flip-flop. The operation is similar to the latch except the latch enable signal is replaced by the clock. The data

will be changed only when the transition is occurred. For example, the 74LS374 will hold the data which occur near the 'high to low' edge of the clock, and the data held until the next transition occur. The data occur at another time will be ignore which differ from latch in which the data allow to change when the latch enable go high.

When many devices are connected in the same manner, the clumsiness is arise, it is easier to use the *8255A programmable peripheral interface*. The 8255A provide the three I/O ports , and one internal port which use to control operation of the chip. The port decoding can be internally occur when two bits address are sent according to the port number. The ports can be divide in two independent group, i.e. group A and group B. The group A is the combination of the port A (00H) and the upper nibble (4 bits) of port C (10H). The group B is combinations of the lower nibble of port C. In the basic operation mode, it may be programmed the ports to be input, output or bi-directional. Moreover, the individual bit in port C can be set independently. For the output operation mode, it is operated as the latch buffer.

The 8255A does not only provide the basic operation but also support the more advance function such as the unidirectional hand shaking in mode 1 and the bi-directional handshaking in mode 3. The more information is provided in the 8255A data sheet.

Digital Converter

This section is brief discussions for the digital converts, i.e. the digital to analog converter (DAC) and the analog to digital converter (ADC). The converters

are importance part since they are in the connections between the difference signal levels, say analog and digital.

The DAC converts the combinations of bits to the corresponding level in analog signal. For example, a 8-bit DAC0808 is set the full scale by the reference voltage 10 V. The output voltage V_o can be written as

$$V_o = V_{ref} \sum_{i=1}^{n} \frac{A_i}{2^n} \qquad (4.1)$$

Where A_i = binary code of bits, where the lest significant bit (LSB) is the A_8[*].

n = the resolution of the DAC , i.e. 8 in the case of DAC0808.

It is clear that the value of the output is discrete. The output voltage difference for the consecutive digital value are the resolution of the DAC. The most DAC operate by the switching principle. The reference voltage is divided by the factor of 2^n and form the n terminal. Each one is connected to the switch which is controlled by a bit which significant in the data format is corresponding to the voltage reference. The summation of the voltage from the 'on' switches is the output voltage.

We con convert the digital to analog signal, on the other hand, the inversion can be performed by ADC. The ADC can be classified into 3 type depend on the conversion method, i.e. direct conversion, partial conversion in sequence and integration conversion (Analog devices, 1992).

---

[*] The notation refer to the specification from National Semiconductor's Linear Handbook for data sheet of DAC0808.

The direct conversion is theoretically the simplest, fastest method. It is based on the 1-bit conversion by the comparator as shown in Fig. 4.5. When the signal is greater than the reference of an individual comparator, the output is negative which give the logic 'high' to the corresponding bit of output or vice versa. The type of converters is sometimes called *flash ADC*. In practice, the space, input capacitance, and power required by the large numbers of comparator have limited the resolution.
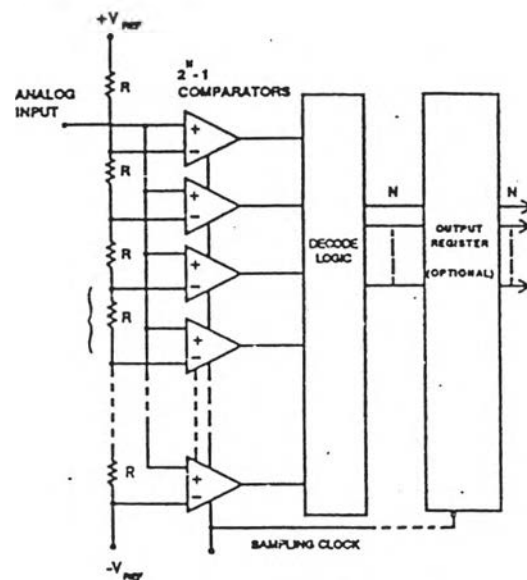
Fig. 4.5 Direct conversion ADC block diagram (Analog Devices, 1992).

The limited may be solve by the partial converter method by which the more steps of conversion are performed. The successive approximation is the most popular methods , the block diagram is as shown in the Fig. 4.6.

When the analog input is compare to the voltage level which is generated by the high speed DAC, the comparator justifies the input and the DAC generated voltage level, successively. The process will be complete when the input and reference

differ least than 1/2 LSB and the digital output is displayed as the digit of the DAC. The start value for DAC is operated is the most significant bit (MSB), say in 12-bit converter is 800H. If the input is greater than the reference voltage from DAC, it set the MSB of the DAC and reset in vice versa.
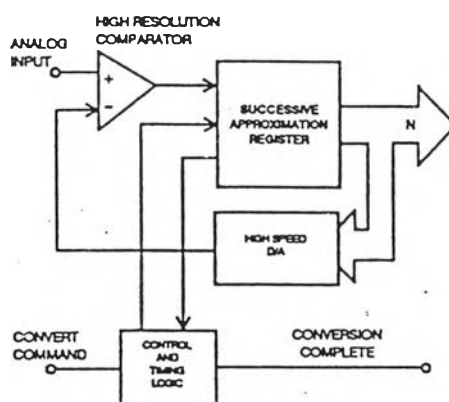
Fig 4.6 Successive conversion ADC block diagram (Analog Devices, 1992).

The process continue to the lower bit, respectively in the same way as MSB. The process complete after the n successive operation where n is the resolution of the ADC. The time is more consumed if the resolution is greater. After process is done, the conversion complete signal will send out to mark the end of conversion. The successive approximation is one of the most popular method for the most ADC such as the ADC1674 which used in the our constructed instrument.

The last conversion technique is integrating conversion such as in the dual-slope converter. The idea behind this method is the measurement of time which corresponding to the level of the input signal. Since it is unused, the discussion is beyond the scoop of this thesis report.

We can also find the input voltage V_i from the digital output X by the following expression, which is given for the n-bit ADC with the full scale trimmed for $X = 2^n$ equal to V_{ref}.

$$V_i = V_{ref} \frac{X}{2^n} \qquad (4.2)$$

However, the analog to digital conversion process is not complete because we have to discuss about the time consumed in the conversion process. If the input signal is change between the conversion time, the digital output is not accurate, especially in the successive approximation ADC. The signal have to be held to keep constant between the conversion time. The circuit which support the purpose is called *sample and hold* circuit. In some ADC chip, this function is build in the package, but in the most ADC, the circuit may be devised, if the measured signal, such as the audio signal, will change rapidly.

Operational and instrumentation amplifier

One of the most useful building block in analog circuit is *the operational amplifier*, or *op-amp*, which is available as an integrated circuit at very low cost.

The op-amp have two input and one output, it is represented by the triangle, as shown in Fig. 4.7. The inputs are call the inverting(-) and non-inverting(+). In fact the both input are the same in physical property but the symbol is applied to the distinguished them.

The ideal op-amp has the following characteristics.

1. Infinite difference gain, say

$$V_o = A(V_+ - V_-) \tag{4.3}$$

where A = open-loop gain (= infinity for the ideal op-amp).

V+ = input voltage at the non-inverting terminal.

V. = input voltage at the inverting terminal

2. Infinite input impedance, i.e., no current enters the input V_+ and V_-.

3. Zero output impedance.

The realistic op-amp differ in following aspect.

1. Finite gain A and decrease as $1/f$. The product of gain an the bandwidth is typically constant in order of 0.1 to 100 MHz (Derenzo, 1990).

2. Finite input impedance, typically 1 MΩ (bipolar) and 10^6 MΩ (FET input).

3. The output V_o is not zero, although the input terminals V_+ and V_- have the same value.

4. The output current is limited and the output voltage limited in the range of supplied voltage.

We start with the ideal case consideration and progress to the realistic situation. Since there is no current flow from the an input terminal to the other, it can be considered that an input the voltage at the both terminals must be same value, say $\Delta V_i = 0$.

It is possible to use the op-amp with no feed back loop, but the most functions can be given by the feed back system. One of the basic circuit arrangement used in the feed back loop is the differential amplifier as shown in Fig. 4.7.
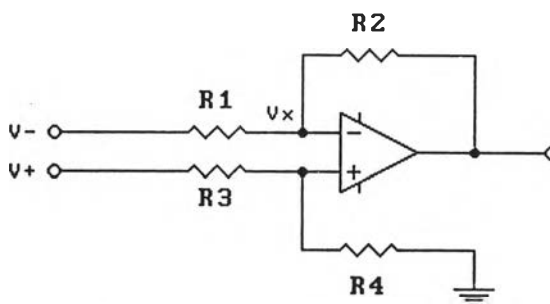
Fig. 4.7 Differential amplifier circuit.

We apply the superposition theorem to the circuit, we have

$$V_o = V_{0-} + V_{+0} \qquad (4.4)$$

where V_{0-} = output voltage when the non-inverting input is setting to be zero.

V_{+0} = output voltage when the inverting input is setting to be zero.

The first term of the right hand side can be found by setting $V_+ = 0$, and applying voltage at V_- terminal. The inverting input can be considered as the ground or sometimes called *virtual ground*. Since there are no current flow through the op-amp input terminal, the current flow directly from the input V_- to the output V_o. Hence

$$\frac{V_- - 0}{R_1} = \frac{0 - V_o}{R_2}$$

$$V_o = -\frac{R_2}{R_1} \qquad (4.5)$$

Let the inverting input terminal as V_x. We consider the situation when the terminal V_- is set to be zero and the input signal is applied to the V_+ terminal. The R_3 and R_4 form the voltage divider circuit because there is no current flow through the op-amp terminal. Thus

$$V_x = \left(\frac{R_4}{R_3}\right) V_+$$

From the previous discussion the voltage at the inverting terminal V_- is equal to the voltage on the non-inverting terminal V_+. We applied this condition we have

$$\frac{0 - V_x}{R_1} = \frac{V_x - V_o}{R_2}$$

$$V_o = \left[1 + \frac{R_2}{R_1} \right] V_x$$

Substitute V_x to the above equation, we have

$$V_o = (\frac{R_4}{R_3 + R_4})(\frac{R_1 + R_2}{R_1}) V_+ \qquad (4.6)$$

Since the V_o in (4.5) is the V_{0-} and in (4.6) is V_{+0}, we substitute them to (4.4).

Thus, we obtain

$$V_0 = \left(\frac{R_2}{R_1} \right) \left[(\frac{R_4}{R_3 + R_4})(\frac{R_1 + R_2}{R_1}) V_+ - V_- \right] \qquad (4.7)$$

We can select the value of each resistance. For the realistic op-amp, there are a small current flow between the input terminals since the finite input impedance. To minimize this effect, the balance current path should be applied, i.e. the input impedance for both input terminal have to equal. By this reason, the resistors should be arrange for $R_1 // R_2 = R_3 // R_4$, which is

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

$$\therefore \quad \left[\frac{R_4}{R_3 + R_4}\right]\left[\frac{R_1 + R_2}{R_2}\right] = \frac{R_3}{R_1}$$

Substitute to (4.7)

$$V_o = \left[\frac{R_2}{R_1}\right]\left\{\frac{R_3}{R_1}V_+ - V_-\right\} \tag{4.8}$$

If we select $R_1 = R_3$ (and follow the balancing current condition R_2 must be the same value of R_4). Hence,

$$V_o = G_d (V_+ - V_-) \tag{4.8}$$

where G_d = close loop gain = R_2/R_1

The above expression show that the output voltage is direct proportional to the difference of the input voltage from the V_+ and V_- terminal. Since the terminal of the op-amp have finite input impedance, the input impedance may be improve if we attached the voltage follower circuit as the buffer to each input terminal as shown in Fig 4.8.

The differential amplifier will have the higher input impedance than the normal circuit. This arrangement is usually called the *instrumentation amplifier* since it always use in the measuring instrument which need the isolation of the measured system from the instrument. Because the high input impedance, the source of the input

signal will not be loaded by the instrument and the measured system is not disturbed

by the measurement, which is the importance principle for measurement system.
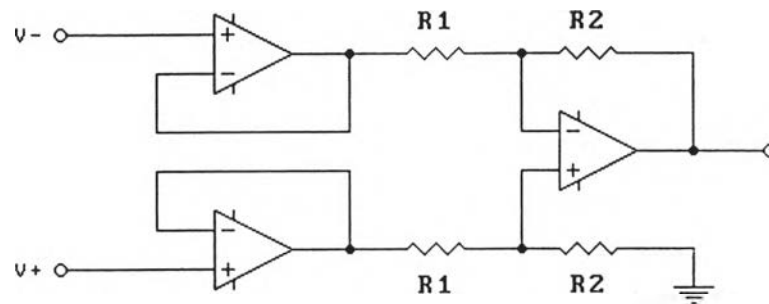
Fig 4.8 Differential amplifier with the input buffers.

The circuit as shown in Fig. 4.8 was tested and its operation was satisfactory. The disadvantage for this circuit come from there have to adjust at least two resistors to the same value for the proper result. However, the modified circuit can be devised to eliminate the problem which is shown in Fig. 4.9.
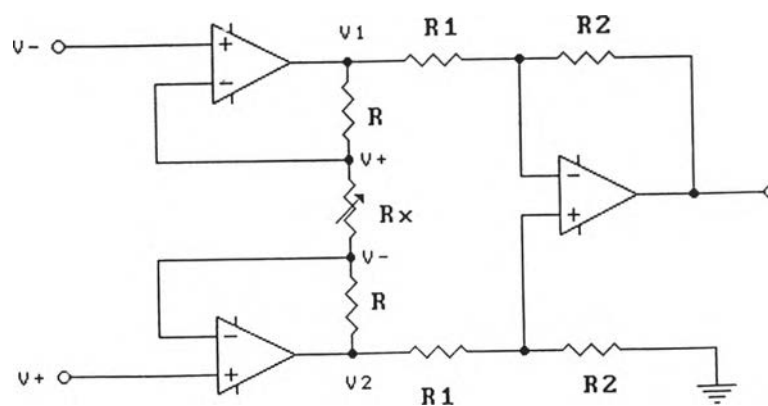
Fig. 4.9 Instrumentation amplifier.

The modified circuit differ from the previous circuit only in the buffer part, the three additional resistors are included. The voltage at the V'_- and V'_+ have the same values to the voltage at the input terminals V_- and V_+ respectively.

Since there are no current pass the input of the buffers, the current flow directly from V_1 to V_2, thus

$$\frac{V_1 - V_2}{2R + R_x} = \frac{V_- - V_+}{R_x}$$

$$V_2 - V_1 = \left[1 + \frac{2R}{R_x}\right](V_+ - V_-) \qquad (4.9)$$

Comparing Fig 4.7 with Fig. 4.9 ,we can see that the $(V_2 - V_1)$ in (4.8) is equal to the term $(V_+ - V_-)$ in (4.9). We substitute this result to (4.8). We have

$$V_o = \left[\left(\frac{R_2}{R_1}\right)\left(1 + \frac{2R}{R_x}\right)\right](V_+ - V_-) = G(V_+ - V_-) \qquad (4.10)$$

where G = instrumentation amplifier gain.

From the (4.10), we can see that the differential gain varies with the value of the variable resistor R_x , which is more flexible than the previous circuit. The

interesting result of (4.10) is when it is open circuit at the terminals of R_x ($R_x = \infty$) , the unity gain occur.

In practical, the characteristic of the realistic op-amp should be considered. The open-loop gain is finite and depend on the frequency as shown in Fig 4.10. The close loop gain was limited by open loop gain, i.e. $G \leq A$. The effect of limitation of gain can be found when we operate the op-amp at high frequency. The effect of gain depend on the frequency of an op-amp can be determined by *the gain-bandwidth product* which can be provided in the data sheet of an the op-amp. The Bode plot of gain V_s frequency for an op-amp is illustrated in the Fig 4.10. The gain-bandwidth product for the open-loop as shown in the figure is 10 MHz.
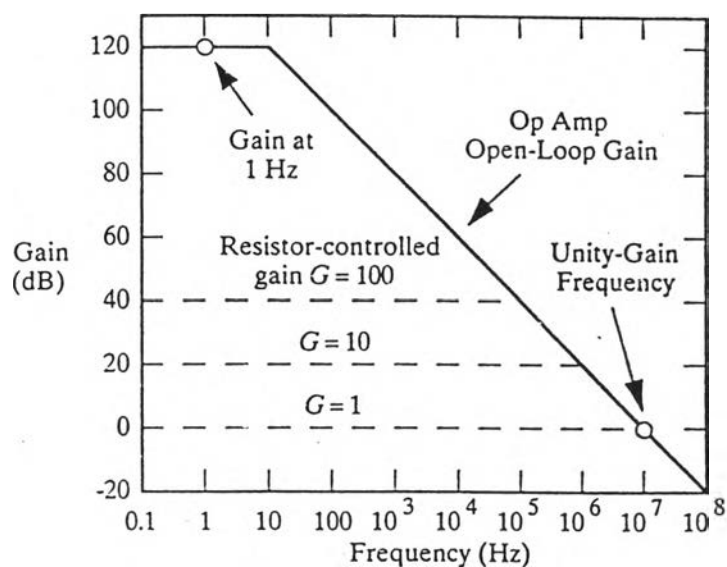
Fig 4.10 Bode plot for dynamic response of an *realistic* op-amp (Derenzo, 1990) .

One of the problem for applications of the op-amp is the *common-mode gain*, G_c. Although the common voltage is applied to the differential input, the output

voltage is not zero. Under the condition of a common-mode signal V_c is applied to the both input. The output voltage in real op-amp can be written as

$$V_o = G_c V_c$$

The common-mode gain cause the output voltage to the op-amp circuit (Derenzo, 1990). In general case, the output voltage from the op-amp is greater than ideal op-amp, say

$$V_o = G(V_+ - V_-) + G_c V_c \quad ; \quad V_c \approx \frac{1}{2}(V_+ - V_-)$$

The common-mode gain should be as small as possible the ratio of the amplifier gain to the common-mode gain is called *the common-mode rejection ratio, CCMR*, which is

$$CCMR = G / G_c$$

The typical value for CCMR is 10^3 to 10^7 for common op-amp. It may be sometimes expressed in term of decibel notation, say $20\log(CCMR)$.

The offset voltage may occur, although the common-mode voltage... is zero. The offset voltage is individual value for each op-amp. Fortunately, the manufacturer

always provide the measure to rescue the problem. For example, the input offset

voltage can be trimmed for LF351 by the additional resistor 10 kΩ as seen in Fig 4.11.
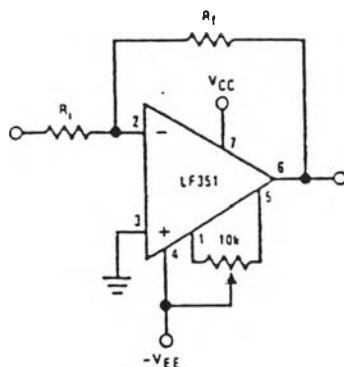
Fig 4.11 Trimming method for offset voltage for LF351 (National Semiconductor

Corp., 1986).

In the case of instrumentation amplifier, the offset should be adjusted at the
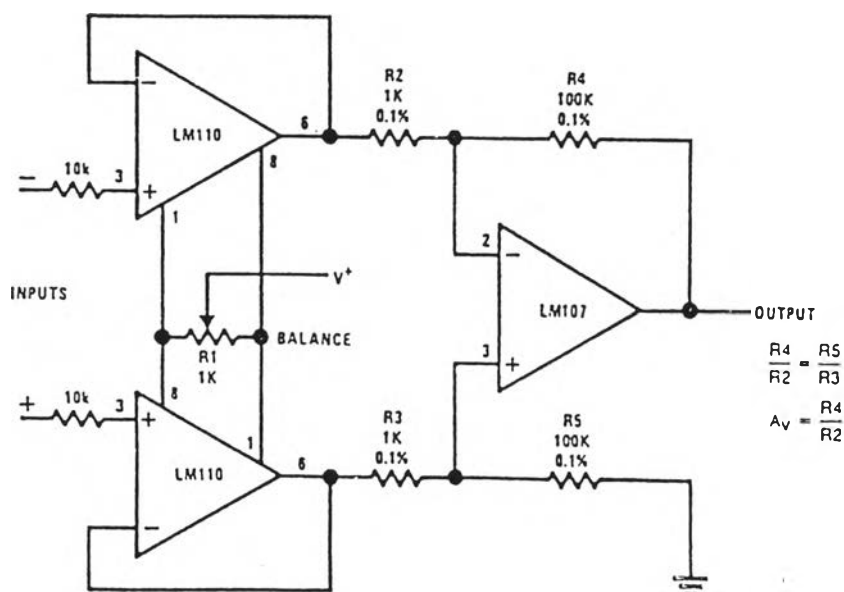
both input buffer as seen in Fig 4.12.

Fig 4.12 compensation of the offset voltage for instrumentation amplifier.

Sampling Theorem and anti-aliasing filter

In digital measurement, the analog signal is sampled. We should carefully consider the effect of the sampling process to the output signal. For the formal analysis, we may represent any signal from the sampling the function x(t) by the function x_n which is

$$x_n = x(t_n) = x(nT_s + t_0) \qquad (4.11)$$

where n = positive integer number

T_s = period of sampling

t_0 = arbitrary initial time constant

The t_0 may be set to zero for general case. It's possible to be considerable the n 's value as the time t_k. In the frequency approach, the signal x(t) may be represent by the complex number (Hamming, 1977) , say

$$x_n(\omega) = e^{i\omega' t_n} = e^{i\omega n} \;\; ; \quad \omega = \omega' T_s$$

where $\omega' = \dfrac{2\pi}{T}$; T = period of the signal.

For any value of ω it can be written in form of

$$\omega = (m\pi \pm \omega_0) \qquad\qquad (4.12)$$

where m = integer number 0,1,2,...

ω_0 = angular frequency $-\pi < \omega_0 < \pi$.

By the complex notation, we can see that the $x_n(\omega_0)$ equal to the many value from the corresponding folded frequency. When the signal is reconstructed to the original function $x_n(\omega)$, there are many possible frequency ω of periodic function $x(t)$ which satisfy the sampling value as seen in Fig. 4.13
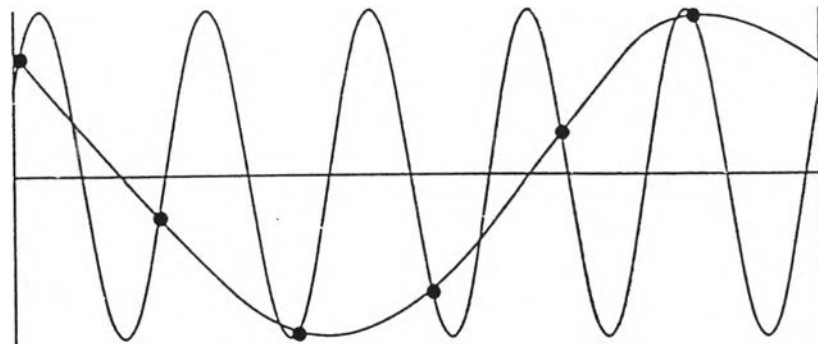

Fig. 4.13 Frequency aliasing (Derenzo, 1990).

Moreover, if the original signal $x(t)$ composed of many components which is common to the most situation, the problem of the undistinguished amplitude for the each frequency may be occurred.

This situation is called *frequency aliasing*. In the sampling system we can see that the highest frequency which aliasing never be occurred is

$$\left| \omega_{max} \right| \leq \pi$$

$$\left| \frac{T_s}{T_{min}} \right| \leq \frac{1}{2}$$

$$T_{min} \geq 2T_s \tag{4.13}$$

It mean that there are at least two sampling per a period of maximum frequency component of the measure signal required to present from the aliasing. On the other hand, it implied that the sampling rate is at least twice of the maximum frequency of the measured signal. The fact is called *sampling theorem* or *Nyquist's theorem* and the maximum frequency as previous discussion is called the *Nyquist's frequency*, say

$$f_s \geq 2f_{max} \tag{4.14}$$

If we are interested in regions of the spectral density, the frequency component which is beyond the Nyquist's frequency have to eliminate before sampling process to prevent the aliasing. Although we have the effective digital filters, it is importance to applied the analog low-pass filter to the measured signal before sent to ADC. The low-pass filter which serve this purpose may be called the *anti-aliasing filter*.

The anti-aliasing filter can be constructed by the sample RC filter circuit which is called the *passive filter*. Since the signal will be decreased magnitude and it

may be degraded by the non-ideal properties, it is more effective to use the *active filter*.

The active filter is constructed by the op-amp. The most simplest form of the active low-pass filter is shown in Fig. 4.14.
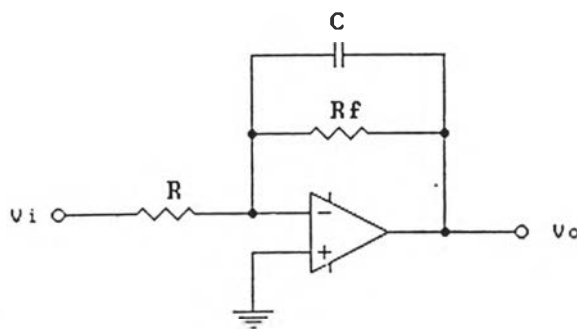

Fig. 4.14 First order low-pass filter.

From (4.5) we have

$$V_o = -\left(\frac{R_f}{R}\right)\frac{1}{1 + j\omega' R_f C} V_i$$

Amplifier gain G and phase different depend on frequency which is

$$\left.\begin{array}{l} G = \left(\dfrac{R_f}{R}\right)\dfrac{1}{1 + j\omega' R_f C} \\[3mm] |G| = \left(\dfrac{R_f}{R}\right)\dfrac{1}{\sqrt{1 + (j\omega' R_f C)^2}} \\[3mm] \Delta\Phi = -\arctan(\omega' R_f C) \end{array}\right\} \qquad (4.15)$$

The amplifier's gain is a function of ω'. The effective of the filter can be determined by take log to the absolute gain equation.

$$S \;=\; 20 \log G$$

$$= -10 \log [1+(\omega' R_f C)^2] + c$$

Consider the asymptotic line occur when the $(\omega' R_f C)^2 \gg 1$, we have

$$S \cong -20 \log(\omega') + c$$

In common use the R_f is set to be equal to R or dc gain equal to unity. It mean that the asymptotic have the slope in Bode plot equal to -20 dB. Some time, it may be called the *first order* or *-20 dB/decade low-pass filter.*
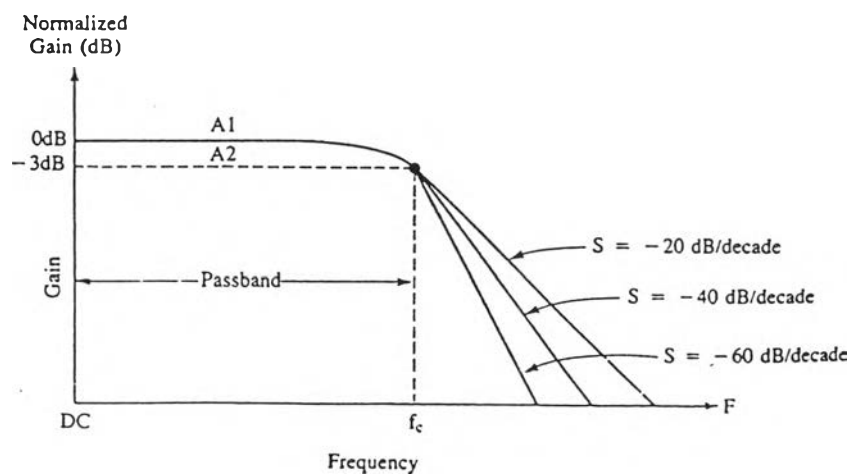
Fig 4.15 Frequency response of the low-pass filters with difference order (Carr, 1991).

The frequency response of the filter may be seen in Fig. 4.15. It can be seen that the frequency response is not exactly the rectangular shape as we want, the asymptotic value determine how rapid to roll-off at the edge of the rectangular. It can be considered that the frequency which the power of signals decreased to a half of the maximum value as the cut-off frequency. Since the power is direct proportional to the square of voltage the cut-off frequency can be defined as

$$\frac{P_o}{P_i} = \left(\frac{V_o}{V_i}\right)^2 = G^2 = \frac{1}{2}$$

$$G = \frac{1}{\sqrt{2}}$$

Hence, for the first order low-pass filter, the cut-off frequency can be found as follow.

$$G = \frac{1}{\sqrt{2}} = \frac{1}{\sqrt{(\omega'_c RC)^2 + 1}}$$

$$\omega'_c = 1 / RC$$

where ω'_c = cut-off frequency

It is possible to cascade the filters to optimized the asymptotic line, but the multistage filter may cause the ripple in the pass band since it is difficult to provide the identical passive components such as resistors and capacitors. However, the second order low-pass filter may be devised by use a single op-amp. The second order low-pass filter (-40 dB/decade) can be seen in the Fig. 4.16
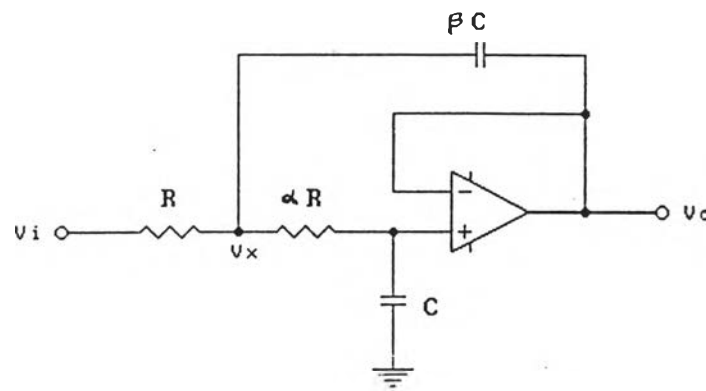

Fig 4.16 Second order low-pass filter.

From the condition $\Delta V = 0$ at the input terminals of the op-amp, we have

$$Vx = (1 + j\alpha\omega RC)\, V_o \qquad (4.16)$$

since there are no current flow though the op-amp, so

$$\frac{V_i - V_x}{R} + \frac{V_o - V_x}{1/j\omega\beta C} = \left(\frac{1}{\alpha\omega + 1/j\omega C}\right)V_x$$

Multiply the above equation by R and substitute $k = \omega RC$ we have

$$\left(\frac{jk}{1 + jk\alpha} + 1 + jk\beta\right)V_x = V_i + jk\beta V_o$$

Substitute V_x from (4.16) , finally we have

$$G = [\ (1-\alpha\beta k^2)\ +\ jk(\alpha+1)\ ]^{-1} \tag{4.17}$$

The magnitude of G can be written as

$$|G| = \left\{1 + [(\alpha + 1)^2 - 2\alpha\beta]k^2 + (\alpha\beta)^2 k^4\right\}^{\frac{1}{2}} \tag{4.18}$$

For the maximum-flat amplitude response, the coefficient of have to be zero (Carson, 1990), which is

$$(\alpha + 1)^2 - 2\alpha\beta = 0$$

$$\beta = \frac{(\alpha + 1)^2}{2\alpha}$$

Hence, (4.17) and (14.18) can be written as

$$G = \left[(1 - \frac{(\alpha+1)^2}{2} k^2) + jk(\alpha+1) \right]^{-1}$$

$$|G| = \left[1 + \frac{(\alpha+1)^4}{4} k^4 \right]^{\frac{1}{2}} \qquad (4.20)$$

$$\Delta\Phi = -\arctan\left[\frac{k(\alpha+1)}{1 - \frac{(\alpha+1)^2}{2} k^2} \right]$$

For the maximum-flat amplitude response, the cut-off frequency can be found in the same way as the first order, and we recall the $k_c = \omega_c RC$ where the ω_c is the cut-off angular frequency, we have

$$k_c = \frac{\pm\sqrt{2}}{\alpha+1} \qquad (4.21)$$

From (4.20), we can see that it decrease more rapid than the gain for the order low-pass filter. We use the asymptotic condition $\omega \gg 1$, we have

$$S \cong -40 \left[\log\left(\frac{\alpha+1}{4} RC \right) + \omega \right]$$

The frequency response curve is decreased in order of -40 dB/decade.

We may chose $\alpha = 1$ for the convenient to design, we recall the gain and phase shift, we have

$$\left.\begin{array}{l} |G| = \dfrac{1}{\sqrt{1+4k^4}} \\[4mm] \Delta\Phi = -\arctan\left[\dfrac{2k}{1-2k^2}\right] \end{array}\right\} \tag{4.22}$$

where $k = \omega RC$ and by (4.) we also have

$$k_c = \pm\dfrac{1}{\sqrt{2}}$$

The positive value will be allowable for k_c. Thus the cut-off frequency can be written by

$$f_c = \dfrac{1}{2\sqrt{2}\pi RC} \tag{4.23}$$

The phase difference at cut-off frequency is

$$\Delta\Phi = -\dfrac{\pi}{2} \tag{4.24}$$

In practice, we have to choose the R and C which is match to serve the cut off frequency f_c . Sometimes, the combination of resistors or capacitors have to selected to arrange for the purpose.

Digital Filter

Although the aliasing frequency is filter out from the measured system. The noise which contaminate the signal is remain. Since the sampling signal is contain in binary code, it is possible to use the numerical method to filter the noise from the data. This filters behave in the same way of the analog filter, it is called the "digital filter". The digital filter is easier maintenance, more stable, more effective and flexible than the analog filter, since it is rather software than a lamps of electronic components, which performance depend greatly on the quality of the components and ambient environment.

In formal analysis the digital filter can be define by the following formula (Hamming. 1977).

$$y_n = \sum_{k=-\infty}^{\infty} c_k x_{n-k} + \sum_{k=1}^{\infty} d_k y_{n-k} \qquad (4.25)$$

The coefficient c_k and d_k are constants. Thus, the digital filter is the linear combination of equal spaced sample x_{n-k} and the computed value of the output y_{n-k}. For each successive n , the formula shift one data point along the whole data. The first

summation represent the *nonrecursive* and the other represent the *recursive* term. The familiar example of the nonrecursive filter is the smoothing by 5 terms.

$$y_n = \frac{1}{5}(x_{n-2} + x_{n-1} + x_n + x_{n+1} + x_{n+2})$$

The example for the recursive filter is the trapezoidal integration formula, i.e.

$$y_{n+1} = y_n + \frac{1}{2}(x_n + x_{n+1})$$

The sampling signal x_n can be written as the integral of the frequency component, which is

$$x_n = \int_{-\infty}^{\infty} e^{i\omega n} d\omega$$

where n = integer number of sampling.

we apply the x_n to the nonrecursive filter, we have

$$y_n = \sum_{k=-\infty}^{\infty} c_k \int_{-\infty}^{+\infty} e^{i\omega(n-k)} d\omega$$

$$= \int_{-\infty}^{+\infty} \left[\sum_{k=-\infty}^{\infty} c_k e^{-i\omega k} \right] e^{i\omega n} d\omega$$

$$= \int_{-\infty}^{+\infty} H(\omega) e^{-i\omega n} d\omega$$

where $\quad H(\omega) = \sum_{k=-\infty}^{\infty} c_k e^{-i\omega k} \equiv \sum_{k=-\infty}^{\infty} d_k e^{i\omega k} \quad ; \quad d_{-k} = c_k$

The H(ω) is called the *transfer function*. We can see that the filtered signal y_n is clearly the compositions of the frequency components as original signal x_n, but the spectrum information is adjusted as the same way of the analog filter. The frequency response of the filter is defined by the transfer function H(ω). It is analogues to the instrument as shown in Fig. 4.17.
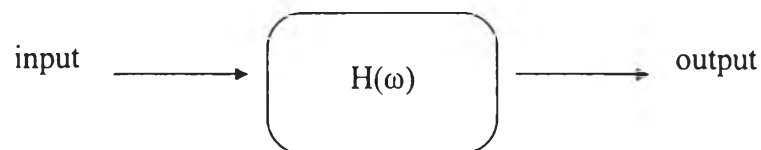

Fig 4.17 Digital filter block diagram.

For example the low pass filter have the transfer function H(ω) as shown in Fig. 4.18. The transfer function can be written as

$$H(\omega) = \sum_{n=-\infty}^{\infty} d_k e^{i\omega k} \quad ; \quad \omega = \frac{\cdot 2\pi f}{f_s}$$

$$= \sum_{n=-\infty}^{\infty} d_k e^{i2\pi k\left(\frac{f}{f_s}\right)}$$

$$= \sum_{n=-\infty}^{\infty} d_k e^{i\omega_k t}$$

We denote that $\omega_k = 2\pi k$ and $t = f / f_s$. Hence, we have the formal discrete Fourier's expansion for the signal have a unity period we expand t over the Nyquist's frequency range. Thus
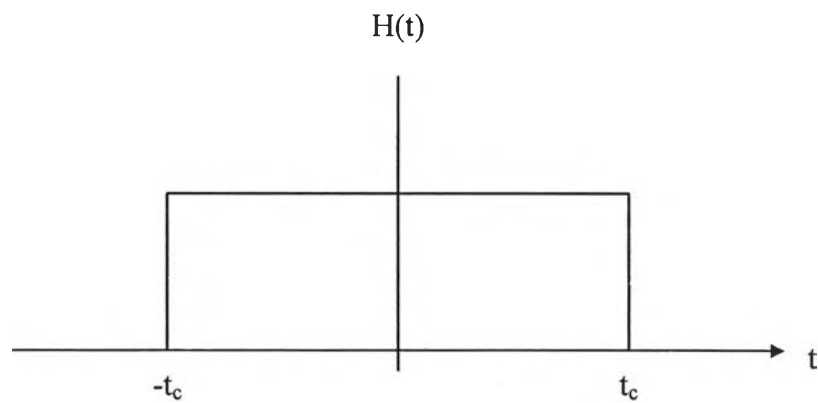
H(t)

Fig. 4.18 Transfer function H(t) for low-pass filter in complex notation.

$$H(t) = \begin{cases} 1 & ; \quad |t| \leq t_c \\ 0 & ; \quad \text{elsewhere} \end{cases}$$

From invert Fourier's transform

$$d_k = \int_{-\frac{1}{2}}^{+\frac{1}{2}} H(t)e^{-i\omega_k t}dt$$

The coefficient c_k can be solve from d_k since $c_k = d_{-k}$. Thus,

$$c_k = \begin{cases} 2t_c & ; \ k = 0 \\[2em] \dfrac{\sin(\omega_k t_c)}{\pi k} & ; \ k \neq 0 \end{cases}$$

(4.26)

where $t_c = f_c / f_s$; $\omega_k = 2\pi k$.

We can obtain the processed signal y_n by substitution of the c_k into the nonrecursive function, i.e.,

$$y_n = \sum_{k=-\infty}^{\infty} c_k x_{n-k}$$

(4.27)

In practical world, it is impossible to operate the digital filter over the infinite limit of k, since the limitation of data and the time consuming process. Thus it is necessary to truncate the filter as

$$y_n = \sum_{k=-N}^{N} c_k x_{n-k}$$

The problem arise when the series is truncated. The frequency response in the pass-band does not remain smooth and transition band occur, as shown in Fig. 4.18. Cornelius Lanczos observed that the ripple can be eliminated by the integrating

(averaging) over the range depend on the number of point N, say 1/N. If the transfer function H(ω) can be written as

$$g_N(t) = \sum_{k=-N}^{N} a_k e^{i\omega_k t}$$

The Lanczos' smoothed function can be written as

$$h_N = \frac{1}{1/N} \int_{t-\frac{1}{2N}}^{t+\frac{1}{2N}} g_N(t)dt$$

Substitute the transfer function to the smoothing function we have,

$$h_N(t) = N \sum_{k=-N}^{N} \int_{t-\frac{1}{2N}}^{t+\frac{1}{2N}} a_k e^{i\omega_k t}dt$$

$$= \sum_{k=-N}^{N} \left[\frac{\sin(\pi k/N)}{\pi k / N} \right] a_k e^{i\omega_k t}$$

$$= \sum_{k=-N}^{N} \sigma(N,k) a_k e^{i\omega_k t}$$

We can see that the operation of Lanczos' integral (or *window*) is the some as multiply by the *sigma factor* $\sigma(N,k)$. the result of the smoothing for N=10 can be seen in Fig. 4.19.
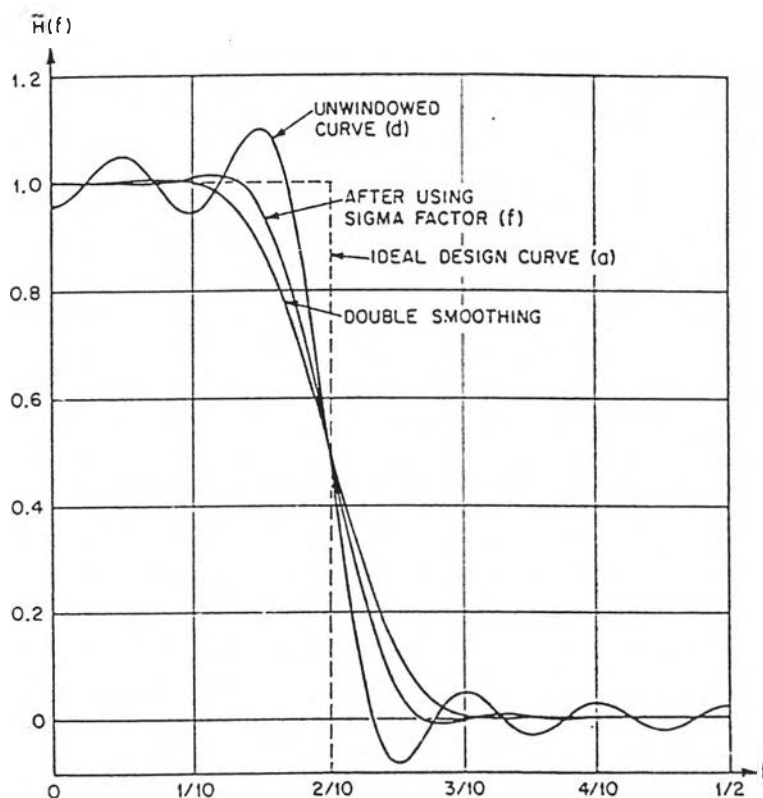

Fig 4.19 Transfer function of a low-pass filter with and without window (Hamming, 1977).

The transition band arise from the Lanczos' window and its width can be determined from the previous definition by the following relation

$$\Delta f_t = \frac{f_s}{N} \tag{4.28}$$

For the digital low-pass filter, the coefficient c_k after the smoothing by Lanczos' widow can be written as

$$c_k = \begin{cases} 2t_c & ; \ k = 0 \\ \left[\dfrac{\sin(\pi k / N)}{\pi k / N}\right] \cdot \left[\dfrac{\sin(\omega_k t_c)}{\pi k}\right] & ; \ k \neq 0 \end{cases} \tag{4.29}$$