NuDAQ â

ACL-8454

MultiFunctions
Counter / Timer Card
User’s Guide

Manual Rev. 1.22: April 15, 2000

The information in this document is subject to change without prior notice in order to improve reliability, design and function and does not represent a commitment on the part of the manufacturer.

In no event will the manufacturer be liable for direct, indirect, special, incidental, or consequential damages arising out of the use or inability to use the product or documentation, even if advised of the possibility of such damages.

Trademarks

NuDAQ® is registered trademarks of ADLINKTechnology Inc.

Other product names mentioned herein are used for identification purposes only and may be trademarks and/or registered trademarks of their respective companies.

Getting service from ADLINK

- **Customer Satisfaction is always the most important thing for ADLINK Tech Inc. If you need any help or service, please contact us and get it.**

ADLINK Technology Inc.			
Web Site	http://www.adlink.com.tw		
Sales & Service	service@adlink.com.tw		
Technical Support	NuDAQ	nudaq@adlink.com.tw	
	NuDAM	nudam@adlink.com.tw	
	NuIPC	nuipc@adlink.com.tw	
	NuPRO	nupro@adlink.com.tw	
	Software	sw@adlink.com.tw	
	AMB	amb@adlink.com.tw	
TEL	+886-2-82265877	FAX	+886-2-82265717
Address	9F, No. 166, Jian Yi Road, Chungho City, Taipei, 235 Taiwan, R.O.C.		

- **Please inform or FAX us of your detailed information for a prompt, satisfactory and constant service.**

Detailed Company Information			
Company/Organization			
Contact Person			
E-mail Address			
Address			
Country			
TEL		FAX	
Web Site			
Questions			
Product Model			
Environment to Use	OS_____________ Computer Brand _____________ M/B : CPU : Chipset : Bios : Video Card : Network Interface Card : Other :		
Challenge Description			
Suggestions for ADLINK			

Contents

How to Use This Guide

This manual is designed to help you use the ACL-8454. The manual describes how to modify various settings on the ACL-8454 card to meet your requirements. It is divided into 5 chapters:

- Chapter 1, "Introduction," gives an overview of the product features, applications, and specifications.
- Chapter 2, "Installation & Configurations" describes the operation method and multi-functions of the ACL-8454. Users should read through this chapter to understand the configurations of the ACL-8454. The chapter will also teach user how to install the ACL-8454.
- Chapter 3, "Registers Format," describes the details of register format of the ACL-8454, this information is very useful for the programmers who want to control the hardware by low level programming.
- Chapter 4, "Signal Connection & Applications," describes the connectors' pin assignment and how to connect the outside signal and devices to / from the ACL-8454. Some applications also be introduced.
- Chapter 5, "C/C++ Library" introduces the C-language library for operating the ACL-8454. Some examples are shown too.

1

Introduction

The ACL-8454 is a general purpose counter/timer and digital I/O card.It is a compact-size add-on card for IBM AT compatible PC in control, monitoring and sensing applications. There are two version of ACL-8454: ACL-8454/6 and ACL-8454/12. ACL-8454/6 has two 8254 chips (6 counters) on board. ACL-8454/12 has four 8254 chips (12 counters).

On ACL-8454/6, four counters are used for external signal, two counters are cascaded as 32-bit timer. On ACL-8454/12, eight counters are used for external signal, the rest of the four counters are designed as two 32-bit cascaded counters.

This card has multi-configurations. The counters can be set as independent counters or cascaded counters. The gate control of every counter comes from either external source or internal cascaded counter. The clock source comes from either internal or external can be selected by jumper setting. A 10 MHz oscillator is used as the base clock of internal clock sources.

It is possible to use this card on variety of powerful counter/timer functions to match your industry and laboratory applications. Users can set the configuration to fit the variety of applications, such as timer, event counter, frequency generator, pulse width measurement, and frequency measurement.

The card also provides digital output and input port. There are 8 digital output channels and these channels can be used to control the external devices. There are digital input channels and these channels are shared the same signal lines with the external clock and the external gate signals. Whenever the external clock or gate signals are not used, they can be dedicatedly used as D/I. There are at least 8 D/I bits under default setting of the ACL-8454 because only two 8254 chips are used.

ACL-8454 provides one interrupt signal which comes from one of four internal or external interrupt sources. Three internal interrupt sources come from the counter output. One external interrupt source shares with one D/I pin. The interrupt can be used for watchdog timer or other applications. One of the 11 interrupt levels on the AT-bus can be selected by setting jumper.

Figure 1.1 shows the block diagram of the ACL-8454.

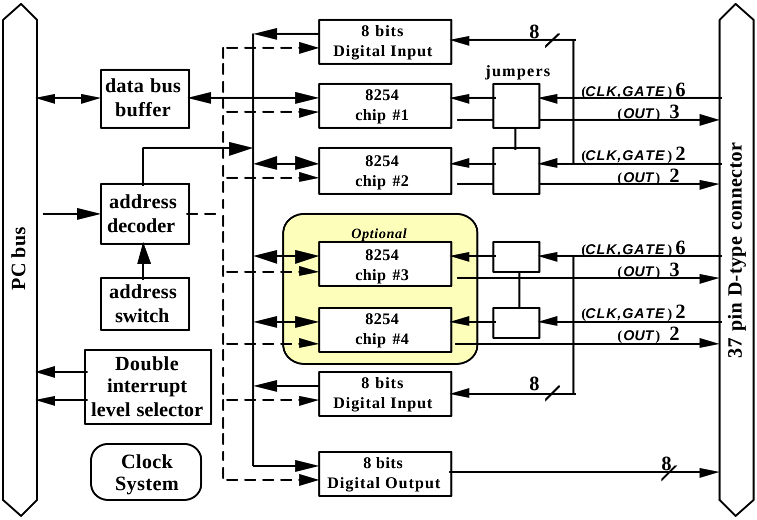


Figure 1.1. Block diagram of the ACL-8454

1.1 Features

The ACL-8454 Counter / Timer and Digital I/O Card provides the following advanced features:

- Default 6 counters/timers are installed and it is expandable to 12 counters/timers at most
- Multi-configurations of counters / timers:
- Flexible setting for each independent counter, the clock source could be external, internal or cascaded. The gate signal is external controlled or internal cascaded signal.
- 8 digital output channels
- 16 digital input channels shared with the same signal lines with external gate control and external clock input
- Two interrupt sources
- The first interrupt source comes from output of counter #6 and this signal is jumper selectable to one of 4 IRQ channels.
- The second interrupt source comes from either internal periodic or external source. This signal is jumper selectable to one of 5 IRQ channels.
- Synchronized 4 channels frequency measurement is possible.
- 37-pin D-type female connector.
- AT-Bus and compact size (Half-size only)

1.2 Applications

- Event counter
- Frequency generator
- Frequency synthesizer
- Pulse width measurement
- Low level pulse generator
- Time delay
- Industry automation
- Watchdog timer
- Laboratory and Industrial automation

1.3 Specifications

♦ **Programmable Counter/Timer**

- **Device :** 8254 x 2 (default), or 8254 x 4 (at most)
- **Counters :** 16-bit down counter
- **Clock source :** internal clock, external source or cascaded
- **Gate control :** default enable or external control

♦ **Digital I/O (DIO)**

- **# of input channels :** 16 channels
- **# of output channels :** 8 channels (dedicated output)
- **Electronics characteristics :** TTL compatible signal

♦ **General Specifications**

- **I/O Base Address** : 8 consecutive address space
- **Dual Interrupt IRQ** :
  Lower IRQ: IRQ 3,5,6,7,9 (5 levels jumper selectable)
  Higher IRQ: IRQ 10,11,12,15 (4 levels jumper selectable)
- **Connector** : 37-pin D-type female connector
- **Operating Temperature** : 0° C ~ 60° C
- **Storage Temperature** : -20° C ~ 80° C
- **Humidity** : 5 ~ 95%, non-condensing
- **Power Consumption** : +5 V @ 210 mA maximum
- **Dimension** : 163mm(L) X 108mm(W)

2

Installation

This chapter describes the configuration and function of the ACL-8454 and the steps to install the ACL-8454. At first, the contents in the package and unpacking information that you should care about are described. The versatile configurations of ACL-8454 are introduced so that you can configure it according to your applications. The default setting of ACL-8454 is shown at the end of this chapter.

2.1 What You Have

In addition to this *User's Manual*, the package includes the following items:

- ACL-8454 Enhanced Multi-function Counter/Timer Card
- ADLINK CD

If any of these items is missing or damaged, contact the dealer from whom you purchased the product. Save the shipping materials and carton in case you want to ship or store the product in the future.

2.2 Unpacking

Your ACL-8454 card contains sensitive electronic components that can be easily damaged by static electricity. The card should be unpacked on a grounded anti-static mat. The operator should be wearing an anti-static wristband, grounded at the same point as the anti-static mat.

Inspect the card module carton for obvious damage. Shipping and handling may cause damage to your module. Be sure there are no shipping and handing damages on the module before processing.

After opening the card module carton, extract the system module and place it only on a grounded anti-static surface component side up. Again inspect the module for damage. Press down on all the socketed IC's to make sure that they are properly seated. Do this only with the module place on a firm flat surface.

Note : DO NOT APPLY POWER TO THE CARD IF IT HAS BEEN DAMAGED.

You are now ready to install your ACL-8454.

2.3 PCB Layout of ACL-8454

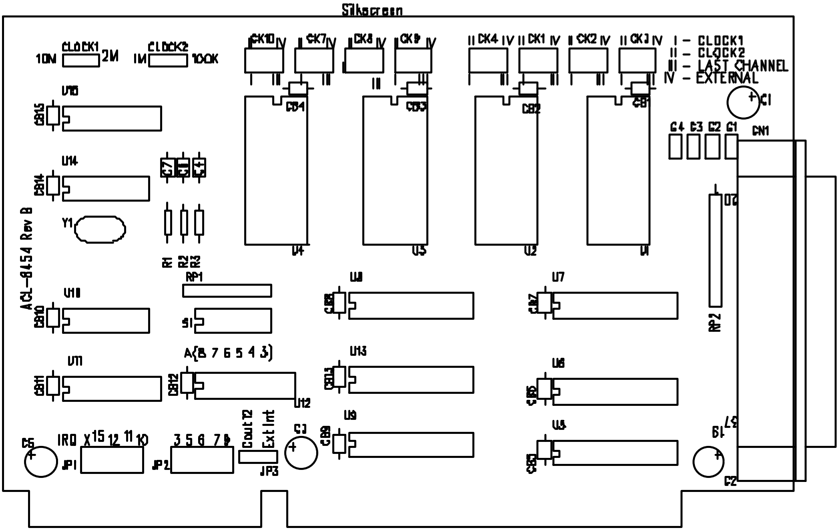


Figure 2.1 PCB Layout of the ACL-8454

2.4 Default Configurations

To operate the ACL-8454 correctly, users should understand the structure of ACL-8454 and details of the possible configurations. The block dagram of the ACL-8454 is shown in chapter 1. It contains the clock system, counter system, interrupt system and address decoder etc. The following sections describe the details and the default setting is list in Table 2.1.

Items	Default Configuration	Setting by:
Base Address	0x240	S1
Clock Source #1	2M Hz	CLOCK1
Clock Source #2	100K Hz	CLOCK2
8254 chips	U1, U2 installed U3, U4 not installed	--
Clock Sources of Counters	Internal Clock Source #1 (=2M Hz)	CK *n*
Gate Sources of Counter 1~4	External Gate	G1 ~ G4
Interrupt Level #1	IRQ 15	JP1
Interrupt Level #2	No Connection	JP2
Source of Interrupt Level #2	External Interrupt Sources	JP3

Table 2.1 Default Configuration of ACL-8454

You can change the ACL-8454's default configuration by setting jumpers and DIP switches on the card for your own applications. The card's jumpers and switches are preset at the factory. A jumper switch is closed (sometimes referred to as "shorted") with the plastic cap in-serted over two pins of the jumper. A jumper is open with the plastic cap inserted over one or no pin(s) of the jumper.

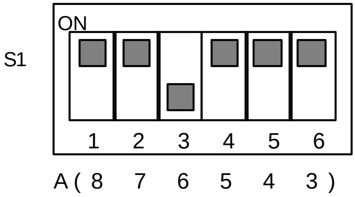
Before changing the default configuration, users must fully understand the operation of the ACL-8454. The setting and the basic operation theorem are discussed in this chapter. It is recommended to refer chapter 3 for details of the registers and to refer chapter 4 for application notes.

2.5 Base Address Setting

The ACL-8454 requires 8 consecutive address locations in the I/O address space. The base address of the ACL-8454 is restricted by the following conditions.

1. The base address must be within the range *Hex 200* to *Hex 3FF*.
2. The base address should not conflict with any reserved I/O address. (See Appendix A.)
3. The base address must not conflict with any add-on card on your own PC. Please check your PC before installing the ACL-8454.

The base address of ACL-8454 is selected by an 6 positions DIP switch **S1**. The default setting of base address is set to be ***HEX 240*.** All possible base address combinations are listed as Table 2.2. You may modify the base address if the default address has been occupied by another add-on card.



A3 ~ A8 control the BASE I/O address.

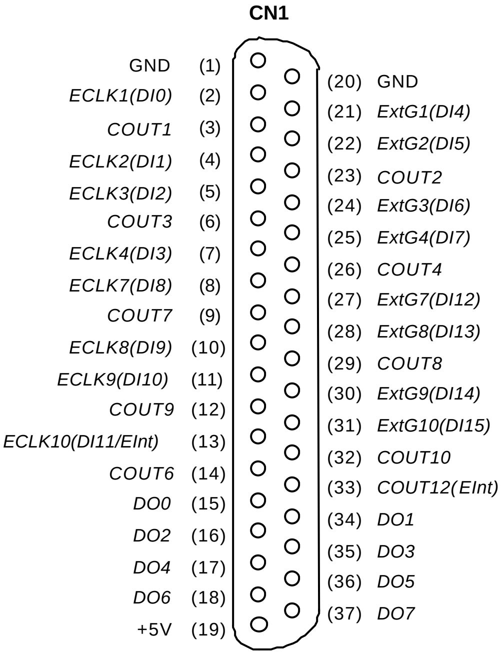
Figure 2.2 Default Base Address Setting

I/O port Address(Hex)	1 A8	2 A7	3 A6	4 A5	5 A4	6 A3
200-207	ON (0)	ON (0)	ON (0)	ON (0)	ON (0)	ON (0)
208-20F	ON (0)	ON (0)	ON (0)	ON (0)	ON (0)	**OFF (1)**
210-217	ON (0)	ON (0)	ON (0)	ON (0)	**OFF (1)**	ON (0)
218-21F	ON (0)	ON (0)	ON (0)	ON (0)	**OFF (1)**	**OFF (1)**
:	:	:	:	:	:	:
237-23F	ON (0)	ON (0)	ON (0)	**OFF (1)**	**OFF (1)**	**OFF (1)**
240-247 **(default)**	ON (0)	ON (0)	**OFF (1)**	ON (0)	ON (0)	ON (0)
248-24F	ON (0)	ON (0)	**OFF (1)**	ON (0)	ON (0)	**OFF (1)**
:	:	:	:	:	:	:
3F0-3F7	**OFF (1)**	**OFF (1)**	**OFF (1)**	**OFF (1)**	**OFF (1)**	ON (0)
3F8-3FF	**OFF (1)**	**OFF (1)**	**OFF (1)**	**OFF (1)**	**OFF (1)**	**OFF (1)**

* A3, ..., A8 is corresponding to PC Bus address lines

Table 2.2. Possible Base Address Combinations

2.6 Pin Assignment of Connector



Legend:

ECLK n : External clock source for counter #*n*
ExtG n : External gate signal for counter #*n*
COUT n : Counter/Timer output of counter #*n*
DO m : Digital output port channel #*m*
DI m : Digital input port channel #*m*
Eint : External interrupt signal input

Figure 2.3 Pin Assignment of Connector CN1

The pin assignment of the 37 pins D-type connector (CN1) is shown in Figure 2.3. The abbreviations of signal names are listed in Table 2.3. The signal name conventions are used through this manual. Some signal pins are multi-function signals. For example, the ***ECLK10*** are not only used as clock source of counter #10, but also be used as external interrupt source and be connected to digital input channel.

2.7 Clock System

The clock system of ACL-8454 provides the internal clock source for the 8254 chips. The block diagram of the clock system is shown in the Figure 2.4. Two clock sources, which named as ***CLOCK1*** and ***CLOCK2****,* are divided from the internal 10Mhz signal. The clock of every counter/timer can be one of the 4 sources: ***CLOCK1, CLOCK2, external clock source*** or ***cascaded source*** from the 'last' channel. Refers to next section for details of setting clock for each counter/timer.

The crystal on ACL-8454 is 10 MHz which is the highest frequency of the card. The maximum speed of the 8254 chip can run under 10 MHz. As users plug the lower speed 8254 chip into ACL-8454, the lower frequency clock source should be used. A frequency divider is used to generate 2MHz for such circumstance.

Notes: Certain versions of 8254 or 8253 chips, such as 8254-5, can run under only 2 MHz.

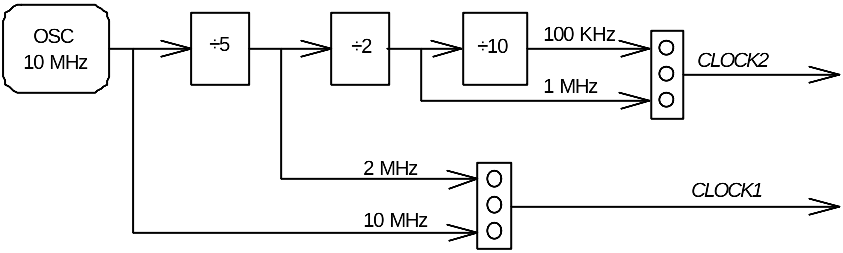
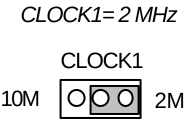
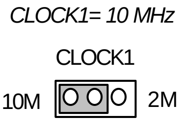


Figure 2.4 Clock system of ACL-8454

CLOCK1 (High Frequency)

The clock source of ***CLOCK1*** can be set by jumper 'CLOCK1'. The frequency could be **10 MHz** or **2 MHz**. Figure 2.5 shows the jumper setting and the corresponding frequency. The default setting of CLOCK1 is using 2 MHz.





(Default)

Figure 2.5 Jumper setting of *CLOCK1*

CLOCK2 (Low Frequency)

The clock source of ***CLOCK2*** can be set by jumper 'CLOCK2'. The frequency could be **1 MHz** or **100 KHz** which are divided from ***CLOCK1***. The ***CLOCK2*** is provided for low frequency applications. Figure 2.6 shows the jumper setting and the corresponding frequency. The default setting of CLOCK2 is using 100 KHz.

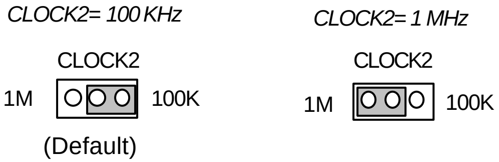


Figure 2.6 Jumper setting of *CLOCK2*

If user's application need a clock frequency lower than 100 KHz, the first method is to use external clock source, the second method is to use one counter to generate a frequency lower than 100 Khz then cascaded the low frequency signal to the other counter's clock source. This ***cascaded counter*** configuration is feasible on the ACL-8454 by jumper setting. See the next section for details of how to set cascaded counter.

2.8 Counters Architecture

There are at most four 8254 chips on the ACL-8454 card. The chip #1 (U1) and chip #2 (U2) are default mounted on the card, therefore 6 counters are default on board. It is possible to expand to four 8254 chips by plugging the additional chip #3 (U3) and chip #4 (U4) into ACL-8454 and totally 12 counters are available. The default counters on chip #1 and #2 are labeled as counter #1 to counter #6. The expandable counters on chip #3 and #4 are labeled as counter #7 to counter #12. Some counters are default configured as **independent counters** and the others are default configured as **cascaded counters**. Table 2.3 illustrates the relationship between the reference number of chips and the counters number.

8254 Chip Number	Reference Number	Counter Number	Type of Counter	Installed on
Chip #1	U1	Counter #1	Independent	ACL-8454/6 ACL-8454/12
		Counter #2	Independent	
		Counter #3	Independent	
Chip #2	U2	Counter #4	Independent	ACL-8454/6 ACL-8454/12
		Counter #5	Cascaded	
		Counter #6		
Chip #3	U3	Counter #7	Independent	ACL-8454/12
		Counter #8	Independent	
		Counter #9	Independent	
Chip #4	U4	Counter #10	Independent	ACL-8454/12
		Counter #11	Cascaded	
		Counter #12		

Table 2.3 Default Counters Architecture

There are three signals (2 input,1 output) for each counter, a clock input signal, a gate control signal, and an output signal. The Figure 2.7 illustrates the block diagram of 8254 counter. ***CLK1*** ~ ***CLK12*** are clock sources and ***GATE1*** ~ ***GATE12*** are gate control signals. The ***COUT1*** ~ ***COUT12*** are output of the counters. The Figure 2.8 shows all the labels and the inter-connection of the 8254 counters when all the 4 chips are insatalled. The ***COUT5*** and ***COUT11*** are used only for internal.

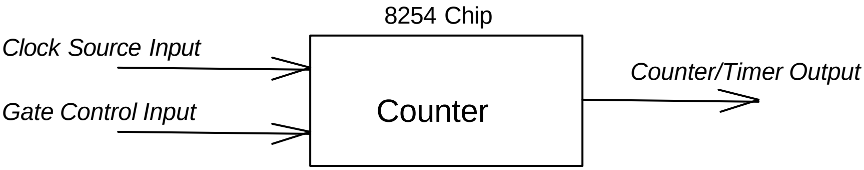


Figure 2.7 I/O pins of 8254 Counter

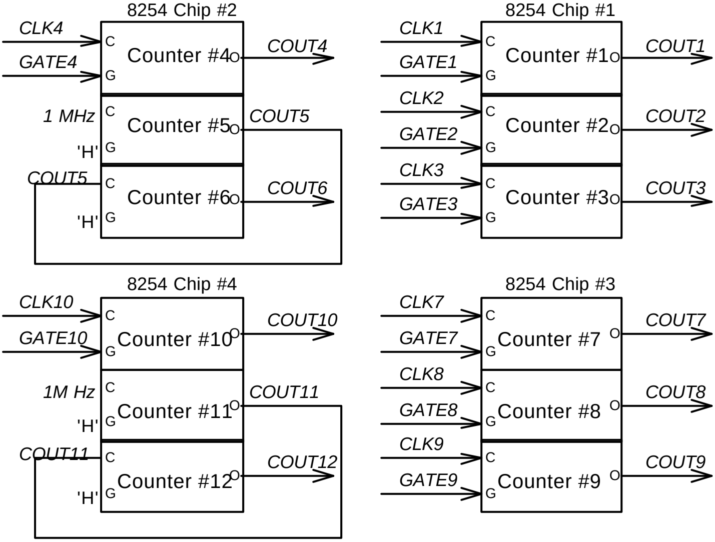


Figure 2.8 Counters Architectural

Independent Counters (Counter #1~#4, & Counter #7~#10)

The Counter #1 to Counter #4 and Counter #7 to Counter #10 are independent because the clock source and gate control of those counters can be set independently. These 8 counters are named as independent counters.

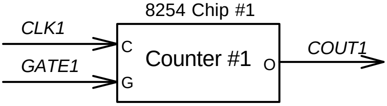


Figure 2.9 Example of 'independent counters'

Cascaded Counters

The connection of Counter #5, #6 and Counter #11, #12 are different with other independent counters. These four counters are named as cascaded counters because the clock sources of counter #5 and #11 come from fixed **1 MHz** and their output are cascaded to counter #6 and #12 respectively. In fact, counter #5, #6 and counter #11, #12 are designed for frequency divider by using 8254's square wave generator mode. The gate of these counters keep at 'H' level for enabling counters all the time. The ***COUT6*** and ***COUT12*** can precisely generate frequency upper to 250 KHz and lower to 0.000233 Hz. Note that the signals ***COUT6*** and ***COUT12*** can also be used as interrupt source. See ‘Interrupt Sources’ section for details. The following figure demonstrates a set of cascaded counter - counter #5 and #6.

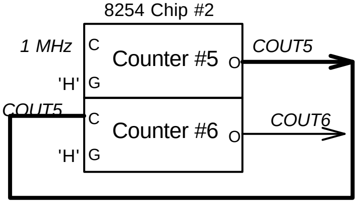


Figure 2.10 Example of ‘cascaded counter’

User Configurable Cascaded Counters

Although there are two cascaded counter on board, users may need more cascaded counters. User can configure the jumper for the clock source of every independent counters. Therefore, the independent counter output can be cascaded to the next counter's clock source by jumper setting. Figure 2.11 demonstrate an example of the user configurable cascaded counter. Refer to next section for details of the clock source setting.

Multi-Configurations

The ACL-8454 provides multi-configurations for many situations. Users may need more independent counters for some applications. Users can installed one more 8254 to chip #3(U3) by yourself to get 3 more independent counters. It is also possible to install one more 8254 chip to #4(U4) and get one more set of cascaded counter or to get another internal interrupt source(refer to section 2.11).

The versatile configurations of ACL-8454 depend on user‘s applications. You may also order the extent version which named as ACL8454/12 that all four chips are installed when shipping.

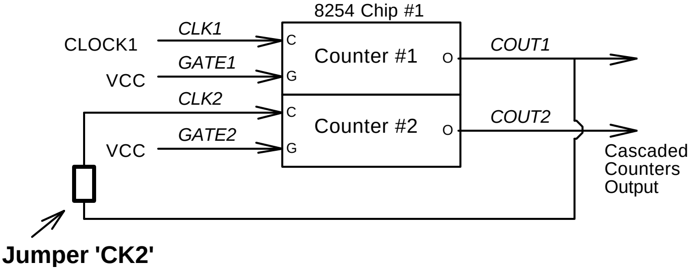


Figure 2.11 Example of ‘user configurable cascaded counters’

2.9 Clock Source Configurations

For every independent counter, four signals can be chosen as clock source by jumper setting. The clock source of counter #n comes from either ***CLOCK1***, ***CLOCK2***, the external clock source (***ECLK n***), or the cascaded counter output. Note that the clock source of the cascaded counters (counter #5, #6, #11 and #12) are fixed.

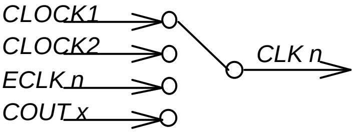
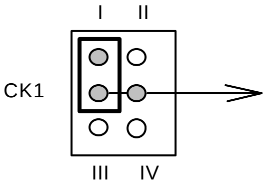


Figure 2.12 Clock Source of Counter #n

The two internal clock sources, ***CLOCK1*** and ***CLOCK2*** come from the clock system (see 'Clock System section'). The cascaded clock source comes from the output of the counter with smaller channel number. For example, the ***COUT1*** is cascaded to source of ***CLK2***, the ***COUT3*** is cascaded to source of ***CLK4****.* The exceptions are the cascaded source of ***CLK1*** comes from ***COUT4*** and the cascaded source of ***CLK7*** comes from ***COUT10****.*

The external clock source named as ***ECLK n*** comes from the 37-pins connector. Figure 2.13. demonstrates the clock source jumper setting counter channel 1 (***CLK1***) by jumper ‘ CK1 ’. In this figure, the clock source comes from the internal source ***CLOCK1***.



Sources of CLK1 (by setting jumper CK1)

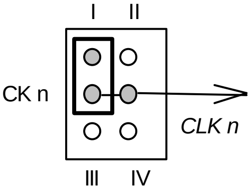
I:	CLOCK1
II:	CLOCK2
III:	LAST CHANNEL (COUT4)
IV;	EXTERNAL (ECLK1)

Figure 2.13 Setting the Clock Source of Counter #1

The jumper setting is flexible for user applications. There are four kinds

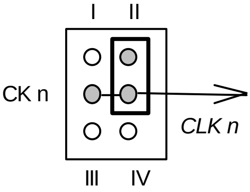
of configurations for every clock source jumper.

(1) Use internal clock source ***CLOCK1***.



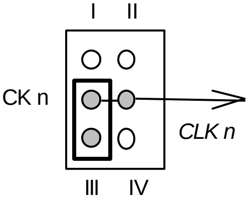
I: CLOCK1
II: CLOCK2
III: LAST CHANNEL
IV: EXTERNAL

(2) Use internal clock source ***CLOCK2***.



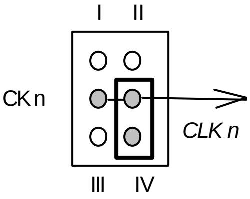
I: CLOCK1
II: CLOCK2
III: LAST CHANNEL
IV: EXTERNAL

(3) Use cascaded clock source from the last channel.



I: CLOCK1
II: CLOCK2
III: LAST CHANNEL
IV: EXTERNAL

(4) Use external clock source ***ECLK n***.



I: CLOCK1
II: CLOCK2
III: LAST CHANNEL
IV: EXTERNAL

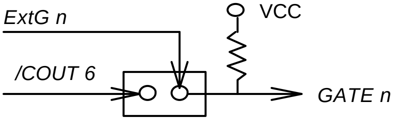
The Table 2.4 shows the reference number of the clock source jumpers and its corresponding counter/timer channels number. The default setting of every jumper is also shown. The clock sources of ***CLK1~4*** come from internal source '**CLOCK1'** and the clock sources of ***CLK7~10*** come from external sources.

Clock source of counter/timer	Reference number of jumper	Default clock source
CLK1	CK1	***CLOCK1***
CLK2	CK2	***CLOCK1***
CLK3	CK3	***CLOCK1***
CLK4	CK4	***CLOCK1***
CLK7	CK7	***ECLK7***
CLK8	CK8	***ECLK8***
CLK9	CK9	***ECLK9***
CLK10	CK10	***ECLK10***

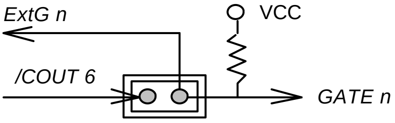
Table 2.4 Reference Number of Clock Setting Jumpers

2.10 Gate Control Configurations

The gate control signals of the independent counters are internally pulled high hence they are default enabled if no external gate used. When the external gate signals are used, the counters can be used to measure pulse width. Moreover, the gate of counter #1 ~ #4 come from the reverse of counter #6 output by jumper selecting. Therefore, the time interval of the counter gate can be precisely controlled and frequency measurement is possible. Figure 2.14 shows the jumper setting of gate control of counter #1~ #4.



(1) Jumper NOT installed : External gate control



(2) Jumper installed : Cascaded gate control

Figure 2.14 Gate source of counter #1 ~#4

The jumper setting is flexible for users‘ applications. There are two configurations: jumper installed or NOT installed. The default state of jumper is not installed and the counter can be controlled externally or just pull-high enable. When the jumper installed, the gate is cascaded from counter #6 output and the ***ExtG n*** signal is an output signal.

Note that the external gate source must NOT be used when cascaded gate source is used. The Table 2.5 shows the reference number of the gate control jumpers and its corresponding counter/timer channels number. The default setting of every jumper is also shown.

Jumper Label	Gate source of counter/timer	Default gate source
G1	***GATE1***	***ExtG1***
G2	***GATE2***	***ExtG2***
G3	***GATE3***	***ExtG3***
G4	***GATE4***	***ExtG4***

Table 2.5 Reference number of clock setting jumpers

2.11 Counter Outputs

The timer/counter output signals (***COUT n***) of 8254 are controlled by clock source, gate control and software programming. All the output of the 8 independent counters are sent to the 37 pins connector directly. The ***COUT6*** and ***COUT12*** of the two pairs of cascaded counters are also send to connector. Therefore, totally 10 counter outputs are sent to connector, see 'Pin assignment' for corresponding signal pin number.

In addition, the output signal may be used as clock source or gate control of the cascaded counters, see the above sections. It is possible to cascaded two counters by jumpers. The counters output ***COUT6***, ***COUT12*** are also used as internal interrupt source (refers to 'Interrupt System').

2.12 Dual Interrupt System

One Internal plus one external interrput sources

The ACL-8454 has ***two interrupt sources*** on board. That means the two interrupt levels or two IRQ lines can be used by this board. The block diagram of the interrupt system is shown in Figure 2.15. The two IRQ channels can be set by jumper **JP1** and **JP2** respectively. The high IRQ level is jumper wired to IRQ 10~15 and the low IRQ level is jumper wired to IRQ 3~9. The source of the high IRQ level comes from ***/COUT6*** which is the **reversed output** of cascaded counters.

Two internal interrput sources

For certain application, ***/COUT6*** is used to generate gate control pulse for measuring frequency and to generate interrupt too. This internal IRQ channel is used to calculate frequency data. However, another internal watchdog timer interrupt may be still necessary, Users can install 8254 chip #4 to get one more pair of cascaded counters (#11, #12). The ***COUT12*** can be used as watchdog interrupt source. Therefore, two internal interrupt sources is available.

The reversed ***COUT6*** interrupt is very useful because the ***/COUT6*** is also send to the gate control of counter #1 ~ #4 for frequency measurement. While ***/COUT6*** is high, the frequency measurement is counting. When ***/COUT6*** goes low, the counting is ending and an interrupt is generated from interrupt system so that the software can sample and memorize the result of frequency measurement. In watchdog application, using reversed ***COUT6*** has no problems because the frequcncy of ***COUT6*** and reversed ***COUT6*** is the same.

The sources of the low IRQ level comes from internal (***COUT12***) or external (***ECLK10***) source which are jumper selectable by **JP3**. Note the ***COUT12*** is not reversed counter output. The default setting (ACL-8454/6) of the higher and lower IRQ levels come from ***/COUT6*** and ***ECLK10*** respectively.

Note: The IRQ channels must not conflict with other add-on cards on your PC.

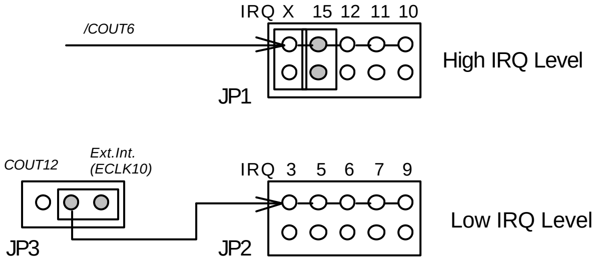


Figure 2.15. Block diagram of interrupt system

Dual interrput example

The dual interrupt system is very useful in some application. For example, most of the application needs a watchdog timer to monitor the system pereodicly, hence, one IRQ channel is used. In addition, the emergency control may be necessary, an additional external IRQ channel is helpful to handle the situation. Therefore, dual interrupt levels are necessary.

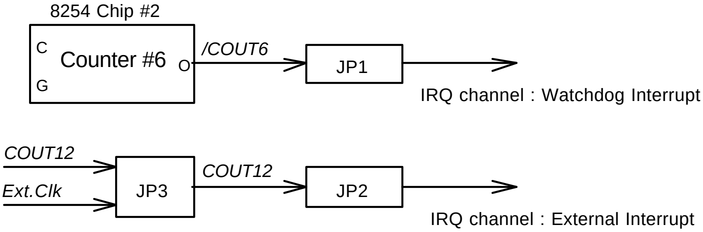

8254 Chip #2
C
Counter #6
O
G
/COUT6
JP1
IRQ channel : Watchdog Interrupt
COUT12
Ext.Clk
JP3
COUT12
JP2
IRQ channel : External Interrupt

Figure 4.8 Example of dual interrupt system

Internal or External Interrupt Selection

The interrupt system is very flexible. No matter under any system configuration, it is possible to generate interrupt internally or externally.

In fact, when chip #4 are not installed, the ***COUT12*** are floating and it's corresponding pin in the connector can be used as digital input pin or external interrupt source. Whenever 8254 chip #4 is installed, it is an internal interrupt source, otherwise, it is used as an external clock source or just a digital input signal. The following table lists the combinations of the interrupt sources. The label *'I'* means 'internal' source and the label '*e*' means 'external' source. The label

‘DI’ means the signal can be used as digital input pin dedicatedly.

	Chip #1	Chip #2	Chip #3	Chip #4	***/COUT6***	***COUT12***	***ECLK10***	Remarks
1	√	√	×	×	*i*	*e*	*DI*	ACL-8454 / 6
2	√	√	√	√	*i*	*i*	*e*	ACL-8454 / 12
3	√	√	√	×	*i*	*e*	*DI*	user installed
4	√	√	×	√	*i*	*i*	*e*	user installed

Table 2.5. Chips installation and interrupt sources

When 2 default chips (#1 and #2) are installed on ACL-8454, /***COUT6*** is internal but ***COUT12*** can be used as external interrupt source. ***ECK10*** is still used as digital input pin. If user installs the chip #3 by yourself, the interrupt sources are in the same configuration as condition 1.

When all 4 chips are installed, /***COUT6***, ***COUT12*** are internal interrupt sources, only ***ECLK10*** can be used as external. If user installs the chip #4 by yourself, the interrupt sources are the same configuration as condition 2.

2.13 Digital Input and Output

To program digital I/O operation is fairly straight forward. The digital input operation is just to read data from the corresponding registers, and the digital output operation is to write data to the corresponding registers. The digital I/O registers‘ format are shown in section 3.11. It is not necessary to set any jumper for digital I/O.

The ACL-8454 provides 16 digital input and 8 digital output channels through the connector CN1. The digital I/O signals are fully TTL/DTL compatible.

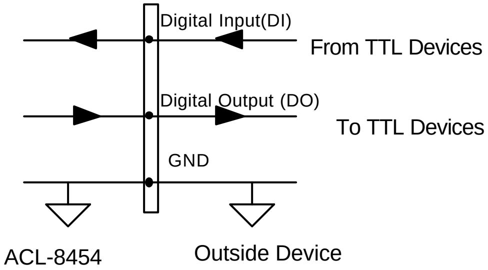


Figure 2.13 Digital I/O Connection

2.14 Summary of Default Setting

- Chip #1 and chip #2 are installed. Counter #1 ~ counter #6 are available. Chip #3 and chip #4 are not installed.
- Counter #1~ #4 and counter #7~#10
  CLK n = ***CLOCK1 n***
  GATE n = ***ExtG n***
- Counter #5 and Counter #6
  CLK5 = 1 MHz
  CLK6 = ***COUT5***
  GATE11,12 are default enable
- Counter #11 and Counter #12
  CLK11 = 1 MHz
  CLK12 = ***COUT11***
  GATE11,12 are default enable

2.15 Notes for Installing More 8254 Chips

The user can install 8254 chips to ACL-8454 card by yourself. Before install the additional 8254 chip, please make sure the ACL-8454 card is removed from the PC slot, no power is applied and no external daughter board is attached.

When plugging 8254 chips, please check the pin of chips must not be fold and install the chip into the socket carefully. After installing the chip, please configure the jumper setting of the ACL-8454 card according to your application. At first the clock source of counter #7 ~ #10 should be set. The gate source of counter #7~#10 can not be configured, however, user should notice whether if any external gate source or digital signal is connected to the signals. The un-wanted signal may control the gate source of the counters can cause your application fail.

The low IRQ source should also be checked because the counter #12 is installed and ***COUT12*** can only be used as internal interrupt source if 8254 chip #4 is installed. Please takes care the issues mentioned above

2.16 Daughter Board Connection

The ACL-8454 can be connected with daughter boards ACLD-9137, ACLD-9188, and DIN-37D to extend the bus.

3

Registers Format

This chapter describes details of the register format of the ACL-8454. This information is quite useful for the programmer who wish to handle the card by low-level program.

In addition, the low level programming is introduced. This information can help the beginners to manipulate the ACL-8454 in the shortest learning time.

3.1 I/O Port Address

The ACL-8454 requires 6 consecutive addresses in the PC I/O address space. There are four 8254 chips in ACL-8454, however, these 8254 chips use the same I/O address. Two chip select bits are used to select active chip. The Table 3.1 shows the I/O address of each register with respect to the base address.

I/O Address	Write	Read
Base + 0	Counter 0	Counter 0
Base + 1	Counter 1	Counter 1
Base + 2	Counter 2	Counter 2
Base + 3	Mode Control	No use
Base + 4	Chips select	DI low byte ***ECLK1~4, ExtG1~4***
Base + 5	Digital Output	DI high byte ***ECLK7~10, ExtG7~10***

Table 3.1. I/O Address Map of ACL-8454

3.2 Timer/Counter Registers

The 8254 occupies 4 I/O address locations in the ACL-8454 as shown blow. Users can refer to NEC's or Intel's data sheet for a full description of the 8254 features, condensed information is specified in Appendix C. Note that only one of the four 8454 chips can be enable in the same time.

Address : BASE + 0 ~ BASE + 3
Attribute : read / write
Data Format :

Base + 0	Counter 0 Register (R/W)
Base + 1	Counter 1 Register (R/W)
Base + 2	Counter 2 Register (R/W)
Base + 3	8254 Mode Control Register (W) 8254 Read Back Register (R)

3.3 Chip Select Register

There are four 8254 chips on board. However, only one 8254 can be selected at a moment. The bit 0,1 of the chip select register (CS1 and CS0) are used to select 8254 chip. Whenever a chip is selected (active), the 8254 I/O address (BASE+0 ~ BASE+3) is selected by this chip. The active chip is enabled by CS0 and CS1 according the following table.

Address : BASE + 4
Attribute : write only
Data Format :

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Base + 4	--	--	--	--	--	--	CS1	CS0

8254 Chip #	CS1	CS0
Chip #1	0	0
Chip #2	0	1
Chip #3	1	0

Chip #4	1	1

3.4 Digital Input Registers

There are 16 digital input channels on the ACL-8454. The digital input channels are common with the external gate signals (***ExtG 1~3 & ExtG 7~10***) and the external clock signals (***ECLK 1~3 & ECLK 7~10***). The external clock source ***ECLK n*** and external gate control ***ExtG n*** can be read back from the DI ports. When the external clock sources and the external gate signals are not used for counters, these channels can be used as digital input signal dedicatedly. For example, the chip #3 and #4 are not installed under default configuration, ***ECLK7~10*** and external gate ***ExtG7~10*** are not used hence the port (BASE+5) can be used as a digital input port dedicatedly. Even if these external clock or gate signals are used for counters, the input port can still monitor their signal level.

Address : BASE + 4 & BASE + 5
Attribute : read only
Data Format :

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Base + 4	*DI7*	*DI6*	*DI5*	*DI4*	*DI3*	*DI2*	*DI1*	*DI0*
Base + 5	*DI15*	*DI14*	*DI13*	*DI12*	*DI11*	*DI10*	*DI9*	*DI8*

or

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Base + 4	***ExtG4***	***ExtG3***	***ExtG2***	***ExtG1***	***ECLK4***	***ECLK3***	***ECLK2***	***ECLK1***
Base + 5	***ExtG10***	***ExtG9***	***ExtG8***	***ExtG7***	***ECLK10***	***ECLK9***	***ECLK8***	***ECLK7***

3.5 Digital Output Register

The register is a general purpose 8 bits digital output port. These signals can be used to control external devices.

Address : BASE + 5
Attribute : write only
Data Format :

Bit	7	6	5	4	3	2	1	0
Base + 5	***DO7***	***DO6***	***DO5***	***DO4***	***DO3***	***DO2***	***DO1***	***DO0***

3.6 Low-level Programming

To manipulate the ACL-8454, users may understand how to write a hardware dependant low-level program. The low-level programming can be carried out by using either assembly or high-level language such as BASIC or C language. The following gives examples to show how to use programming language to access an add-on I/O card.

Getting Start

Before programming, the add-on card should be correctly installed. After installing the card, the users should already understand how much system (PC) resources are used by this card, such as I/O address, IRQ channels, and even DMA channel, etc.

The second step is to study the register format and the operation theorem of the card. Then users can try to write low-level programs to operate it. Although the high-level program library is available, the low-level programming can improve the efficiency and perform functions which the library does not support. The low level programming is not difficult and may be necessary to understand.

Programming Language

The programming language to be used is dependent on users' familiarity and the system requirement. No matter what kind of language is used, the user must understand the syntax of the I/O instructions to access the I/O card. The following sections introduce the syntax of the often used programming language. In each section, the write (output) port instruction and the read (input) port instruction are shown. In the examples, the base address of the I/O card is assume as *HEX 240* and the port of the register to be access is BASE+2.

Assembly

To write an output port:

```
        out     240h,value
        out     240h,register
```

To read an input port

```
        in      A, 240h
```

BASIC language

To write an output port:

```
10      BASE=&H240
20      VALUE% = &H2F
30      OUT( BASE+2), VALUE %
     or
10      OUT( &H242 ), &H20
```

To read an input port

```
10      BASE=&H240
20      VALUE=INP( BASE+2)
     or
10      VALUE=INP( &H242 )
```

C language (Borland C++)

To write an output port:

```
#define    BASE      0x240
unsigned int         Value=0x2F;
outportb( BASE+2 ,  Value );
              or
outportb( 0x242 ,  0x2F );
```

To read an input port

```
#define    BASE      0x240
unsigned int         Value;
 Value = inportb( BASE+2 );
              or
Value = inportb( 0x242 );
```

Perform Functions

Users should study the operation theorem and the relative data sheet to understand how to operate this card, then use the low-level programming to perform those functions. Generally, the DIO control can be easily performed by only a few instructions, it is very suitable to use the low level programming.

As to the higher level functions such as the interrupt service routines, pulse width measurement, frequency measurement, etc, user may use the library or modify the examples to carry them out. However, fully understanding of the PC system is necessary for certain applications.

4

Applications

4.1 Timer / Counter Connection

The ACL-8454 has four 8254 chips on board. It offers 8 independent 16-bit programmable down counters and two pairs of cascaded counters. To implement your application, you can following the procedure to design your application and connect the signals.

1. Check if use a clock source with fixed frequency, if answer is ‘No’ then go to step 7.
2. Calculate the frequency of clock according to your application and decide the clock source: internal, external, or cascaded. Then decided which counter is used.
3. If internal clock source is used, configure the CK n jumpers according the calculation in step 2.
4. If external clock source is used, to generate a clock source outside the board and check the frequency.
5. If cascaded clock source is used, to configure and program the cascaded timer to generate the desired clock source.
6. If frequency of the clock is not fixed, only external source can be used and then connect the external jump wires according your application.
7. Decide the gate control source: always enable, externally control, or cascaded controlled.
8. To connect or configure the gate signal of counter/timers.
9. To program the counter/timers using desired mode.

4.2 Frequency Generator

Example 1 : To generate a 250 K Hz Square Wave.

step 1: To use **fixed** clock source because the output is a fixed frequency.

step 2: I**nternal 2M Hz** is suitable to generate 250K Hz frequency. Use Counter #4 for this application.

∵ 250 K Hz = 2 M Hz / 8

step 3: Set jumper ‘CK4’. The clock source is coming from internal ‘CLOCK1’. Note that the CLOCK1 must be set as 2 M Hz. However, the CLOCK1 is shared with other counters. You must consider the clock source of other counters which using CLOCK1 in the same time.

step 4 ~ 6 : Skip these steps.

step 7: The gate source is enable always. Note the jumper ‘G4’ should be removed and no external gate is used.

step 8: Connect the counter output to external device. The following block diagram illustrates the application

step 9: Write the control program. Please refer the ‘DEMO1.C’ source code.

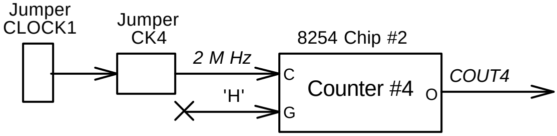


Figure 4.2.1 Example of frequency generator (1)

Example 2 : To generate a very low frequency of 1 pulse / 1 hour

step 1: To use **fixed** clock source because the output is a fixed frequency.

step 2: Because the desired frequency (1/3600sec=0.000278Hz) is too slow to use one counter to generate. Even if using cascaded counter architecture, the clock frequency of the first stage counter must be as low as possible. At first, try to use **internal** clock. Assume the clock of the first stage is 100 K Hz. then the frequency divider value of the two cascaded counters is:

100 K Hz / 0.000278 Hz = 360,000,000

It is really a large number. Share the divider value to the two counters, for example, the divider of the first stage can be 60000 and the divider of the second stage is 6000.

∵ 360,000,000 = 60000 x 6000

Therefore, ‘user configurable cascaded counters’ can be used to solve this problem. The counter #1 and #2 are used in this example.

step 3: Set jumper ‘CK1’. The clock source is internal ‘CLOCK2’ and CLOCK2 is defaultly set as 100 K Hz. Then set jumper ‘CK2’ and the clock source is coming for the output of the ‘Last channel (#1)‘.

step 4: Skip these steps.

step 5: Write and verify the control program for counter #1.

step 6: Skip these steps.

step 7: The gate source of the two counter is enable always. Note the jumper ‘G1’ and ‘G2’ should be removed and no external gate signals are used.

step 8: Connect the counter #2 output to external device. The following block diagram illustrates the application

step 9: Write and verify the control program. Please refer the ‘DEMO2.C’ source code.

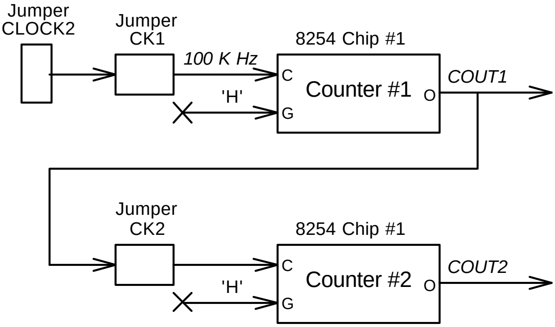


Figure 4.2.2 Example of frequency generator (2)

4.3 Pulse Width Measurement

Example : To measure pulse width (with D T < 32ms)

step 1: To use **fixed** clock source as base time interval (or base frequency).

step 2: Assume I**nternal 2M Hz** clock is used. The time base is

D t = 1/2M=5x10e-7 sec

The count range for measuring pulse width is:

D t < pulse width < ***D t*** *65535 (=32.768 msec)

If the specification of the pulse width to be measured is in the range, the 2M Hz can be used. Otherwise changing the base frequency of the counter. The counter #3 is used in this example.

step 3: Set jumper ‘CK3’ for internal 2 M Hz clock source.

step 4 ~ 6 : Skip these steps.

step 7: The external gate source is just the signal to be measured. The width of the high pulse can be measured. Note that if the pulse is shorter, the time resolution is worser. If the pulse is wider, the limitation of the maximum pulse width should be care.

step 8: The following block diagram illustrates the application

step 9: Write the control program. Please refer the ‘DEMO3.C’ source code.

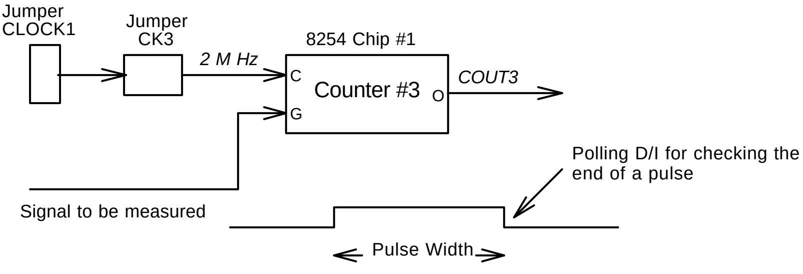


Figure 4.3 Example of pulse width measurement

4.4 Frequency Measurement

Example : To measure frequency around 1~100 K Hz

step 1: This application need two counters. One counter is used to generate a pulse whose time interval is very precise. The pulse is used to enable the other counter (counting counter) by gate control. On ACL-8454, internal gate control is possible. The internal gate is coming from /COUT6. In this example, the pulse generate is counter #6 and the counter #1 is used to measure frequency.

step 2: The maximum value of counting counter is no more than 65535, For measuring 100 K Hz frequency, the time interval should be within 1/100 K Hz x 65535 = 0.655 sec. If the time interval is wider then the measurement resolution is better, however, if time interval is too long the counting value will be overflow. That means the low pulse width of counter #6 output should shorter than 0.655 sec. User can try to generate the pulse by counter #6 by yourself.

step 3: Configur jumper ‘CK1’. The clock source is the signal to be measured.

step 4~6: Skip these steps.

step 7: Set the internal jumper ‘G1’. The gate is controlled internally.

step 8: The following block diagram illustrates the application

step 9: Write and verify the control program. The frequency of the signal is :

frequency = counting value of counter #1 / precise time interval

Please refer the ‘DEMO4.C’ source code.

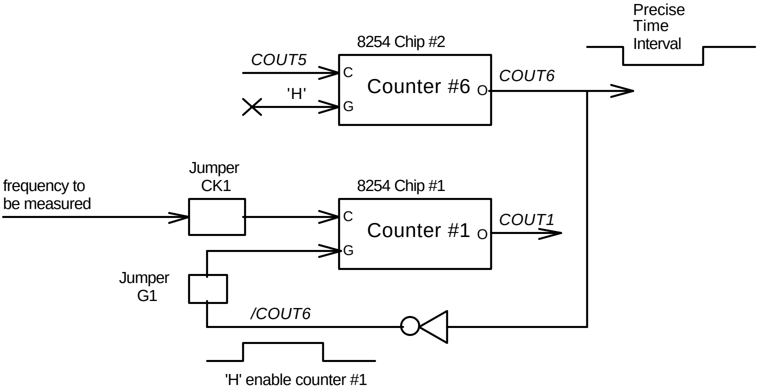


Figure 4.4 Example of frequency measurement (1)

The ACL-8454 can **synchronously** measure frequency from **four** channels because the internal gate control is connected to 4 counters (#1~4) in the same time. Furthermore, as the gate signal goes low, an interrupt is generated. The user can write an ISR to sample and calculate the frequency of all the 4 channels. This feature is very useful in control system.

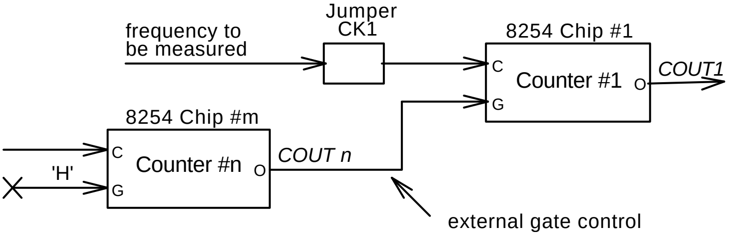


Figure 4.6 Example of frequency measurement (2)

Note that the precise gate signal is not nessarily coming from the internal source. The gate can be generate by any counter and route to gate of counters externally (refer Figure 4.6.). The frequency measurement application is quit complicate. User should also be care of the frequency range to be measured then design a system according to the system specification.

4.5 Event Counter

Example : To count external event in 1 sec

step 1: This application needs one counter to generate a time base of 1 sec and the second counter to count the event. The cascaded counter #5, #6 can perform the watchdog timer. The another counter #1 is used as an example to count external event. The clock source of counter #1 is the event signal and the frequency is not fixed.

step 2~6: Skip these steps.

step 7: The gate source is alwayse enable and the external gate must be removed.

step 8: Connect the signal according to Figure 4.7.

step 9: Write the control program. Please refer the ‘DEMO5.C’ source code.

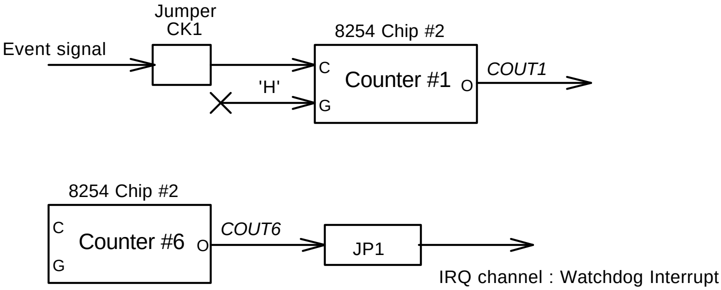


Figure 4.5 Example of event counter

5

C/C++ Library

There are more than 10 functions provided by the C language library. By using the C language library, it saves a lot of programming time.

If you need to perform some special functions which are not provided in the library, you can modify the library according to your requirement. The fully commented C source of the library is also included in your software library diskette. It is a good starting point for C language programmers who wish to modify the functions in the library.

In addition to library and source code, some demonstrating programs are also included in the disk. It will help you to understand the library more quickly.

Please refer to the demostration examples in the software to get examples of the using of the library

5.1 Installation

◆MS-DOS/Windows 3.1 Software Installation

With CD-ROM:

1. Turn your PC's power switch on
2. Put the CD-ROM into your CD-ROM drive
3. Type the commands:

Dos:

```
X:\> CD\NuDAQISA\8454\DOS
X:\NuDAQISA\8454\DOS> SETUP
```

(x identifies the drive that contains the CD-ROM)

After installation, all the files *of ACL-8454 Library & Utility for DOS* are stored in C:\ADLINK\8454\DOS directory.

Windows 3.1:

```
X:\> CD\NuDAQISA\8454\WIN31
X:\NuDAQISA\8454\DOS> SETUP
```

(x identifies the drive that contains the CD-ROM)

After installation, all the files *of ACL-8454 Library & Utility for Windows 3.1* are stored in C:\ADLINK\8454\W31 directory.

◆Win-95/Win-NT Software Installation

With CD-ROM:

step 1. Place the ADLink CD-ROM into the appropriate CD drive.

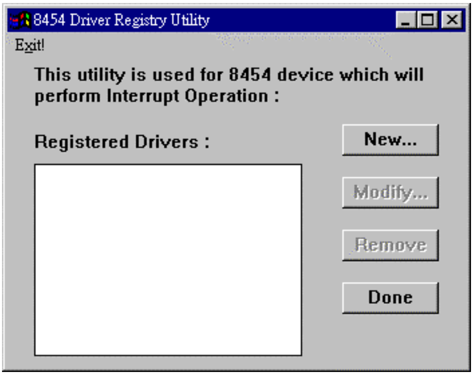
step 2. If auto-run setup program is not invoked automatically, please execut X:\setup.exe. (X indicates the CD-ROM drive)

step 3. Select NuDAQISA→ACL-8454→Win95DLL or NuDAQISA→ACL-8454→WinNT DLL to install software.

[Win-NT only]

When the software component installation process is completed and the system that ACL-8454 Library is installed on is ***Windows NT***, Setup will launch the *Driver Registry Utility (8454Util.exe)* for you to make the registry of the drivers that you want to perform ***interrupt operation***.

The *Driver Registry Utility* first shows the follwing window. If 8454 driver has been registered, it will be shown on the *Registered Driver* list.

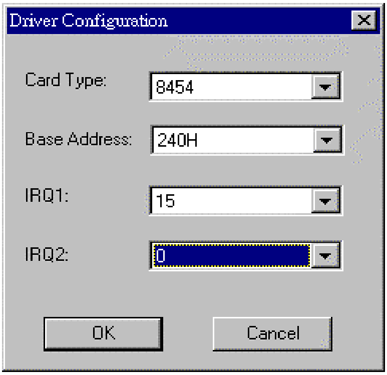


This utility is used for 8454 device which will perform Interrupt Operation :

Registered Drivers :

If the ACL-8454 device **DOES NOT** want to perform interrupt operation', please click "Done" or "Exit!" to exit this utility. Click "New…" button and a *Driver Configuration* window appears for you to set IRQ levels and base address.

ACL-8454 can support two interrupt interrupt lines. One is IRQ1(high IRQ) from /COUT6 which is the reversed output of internal cascaded counters and the other is IRQ2 (low IRQ) from COUT12 which is from internal timer or external source (ECLK10). The "Driver Configuration" window for ACL-8454 is:



Card Type: 8454

Base Address: 240H

IRQ1: 15

IRQ2: 0

If you don't need to use both of these two IRQ lines, set the unused IRQ level as "0". Then you can save one IRQ level for your system. After the setting for IRQs level and base address, click "OK" to register the driver.

When you finish the dirvers reigstry, select "Done" or "Exit!" to exit this utility. To make the registered drivers work, you have to re-start Windows NT system.

Note : If your device driver can not be started, please check if the resources (I/O Port Address or IRQ Level) conflict with other hardware device. You can use *8454Util* utility to change the resource setting. After the ACL-8454 DLL for Win-NT is installed, this utility is located in <InstallDir>\Util directory.

[Win-95 & Win-NT]

After you complete the installation of ACL-8454 Library, ACL-8454's DLL(8454.DLL) is copied to Windows System directory (default is C:\Windows\System for Win95 or C:\Winnt\System32 for Win-NT). The driver files for Win-95 (w95_8454.VXD) are copied to the Windows System directory (default is C:\ Windows\System) or the driver files for Win-NT (DIO.SYS and 8454.SYS) are copied to the Win-NT System Drivers directory (default is C:\Winnt\System32\Drivers).

5.2 W_8454_Initial

@ ***Description***

To initial the base address used which used by the following functions. The default base address set in the library is 0x240. However, you should call this function before using others functions.

@ ***Syntax***

Microsoft C/C++ (Windows, Win95 & Win-NT)

```
int W_8454_Initial(int base_address)
```

Visual Basic

Windows 3.11 Version:

```
W_8454_Initial (ByVal base_address As Integer) As
Integer
```

Win-95 or Win-NT Version:

```
W_8454_Initial (ByVal base_address As Long) As Long
```

C/C++ (DOS)

```
int _8454_Initial( int base_address )
```

@ Arguments

int base_address : base address of the card

@ Return Value

No_Error : No error
Base_Address_Error : When the base address is illegal

5.3 Set_Chip

@ Description

This is a macro which is used to select or active one of the four 8254 chips, Refer the 8454.H for the definition of the Set_Chip macro. This macro is used by other functions in this library, it may not necessary be used by users.

@ Syntax

Microsoft C/C++ (DOS, Windows, Win95 & Win-NT)

```
Set_Chip( int ChipNo )
```

@ Arguments

int ChipNo : chip number, equal 1 to 4.

@ Return Value

No return value

5.4 W_8454_Write_Counter

@ ***Description***

To write a command to a counter. The user can directly assign the counter number 1~12, therefore it is not necessary to care about the chips number and other details. The output and response of the counter is dependent on the ‘Mode’ argument and the configuration on hardware.

@ ***Syntax***

Microsoft C/C++ (Windows, Win95 & Win-NT)

```
int W_8454_Write_Counter(int CntrNo,int
Mode,unsigned int CntrVal)
```

Visual Basic

Windows 3.11 Version:

```
W_8454_Write_Counter (ByVal CntrNo As Integer, ByVal
Mode As Integer, ByVal CntrVal As Integer) As
Integer
```

Win-95 or Win-NT Version:

```
W_8454_Write_Counter (ByVal CntrNo As Long, ByVal
Mode As Long, ByVal CntrVal As Long) As Long
```

C/C++ (DOS)

int _8454_Write_Counter(int CntrNo,int Mode,unsigned int CntrVal)

@ ***Arguments***

int CntrNo	: Counter number, equals to 1~12
int Mode	: Operation mode of counter, equals to 1 ~6
unsigned int CntrVal	:The 16 bits counter value to write to the counter.

@ ***Return Value***

No_Error	: No error
Invalid_Counter_No	: CntrNo is out of range.
Invalid_Timer_Mode	: Mode is out of range

5.5 W_8454_Read_Counter

@ Description

To read the counter value from a counter. The user can directly assign the counter number 1~12, therefore it is not necessary to care about the chips number and other details.

@ Syntax

Microsoft C/C++ (Windows, Win95 & Win-NT)

```
int W_8454_Read_Counter (int CntrNo, int *Mode,
unsigned int *CntrVal)
```

Visual Basic

Windows 3.11 Version:

```
W_8454_Read_Counter (ByVal CntrNo As Integer, Mode
As Integer, CntrVal As Integer) As Long
```

Win-95 or Win-NT Version:

```
W_8454_Read_Counter (ByVal CntrNo As Long, Mode As
Long, CntrVal As Long) As Long
```

C/C++ (DOS)

```
int _8454_Read_Counter(int CntrNo, int *Mode,
unsigned int *CntrVal)
```

@ Arguments

int CntrNo	: Counter number, equals to 1~12
int *Mode	: Operation mode of counter, equals to 1 ~6
UINT *CntrVal	: address to save the read back counter value

@ Return Value

No_Error	: No error
Invalid_Counter_No	: CntrNo is out of range.

5.6 W_8454_Stop_Counter

@ Description

This function is used to to stop a specified counter. The user can directly assign the counter number 1~12, therefore it is not necessary to care about the chips number and other details.

@ Syntax

Microsoft C/C++ (Windows, Win95 & Win-NT)

```
int W_8454_Stop_Counter (int CntrNo, unsigned int
*CntrVal)
```

Visual Basic

Windows 3.11 Version:

```
W_8454_Stop_Counter (ByVal CntrNo As Integer,
CntrVal As Integer) As Long
```

Win-95 or Win-NT Version:

```
W_8454_Stop_Counter (ByVal CntrNo As Long, CntrVal
As Long) As Long
```

C/C++ (DOS)

```
int _8454_Stop_Counter(int CntrNo, unsigned int
*CntrVal)
```

@ *Arguments*

int CntrNo	: Counter number, equals to 1~12
UINT *CntrVal	: address to save the read back counter value

@ *Return Value*

No_Error	: No error
Invalid_Counter_No	: CntrNo is out of range.

5.7 W_8454_DO

@ *Description*

To write a 8 bits data to the digital output port.

@ *Syntax*

Microsoft C/C++ (Windows, Win95 & Win-NT)

```
int W_8454_DO (int DO_Value)
```

Visual Basic

Windows 3.11 Version:

```
W_8454_DO (ByVal DO_Value As Integer) As Integer
```

Win-95 or Win-NT Version:

```
W_8454_DO (ByVal DO_Value As Long) As Long
```

C/C++ (DOS)

```
int _8454_DO(int DO_Value )
```

@ *Arguments*

DO_value : the value to write to digital output port, only the 8 LSBs of the value is effective.

@ *Return Value*

Always no error

5.8 W_8454_DI

@ *Description*

To read the data of the digital input port. It is possible to read the 8 LSBs, 8 MSBs, or read only one bit channel by the DI relative functions.

@ *Syntax*

Microsoft C/C++ (Windows, Win95 & Win-NT)

```
int W_8454_DI( UINT *DI_Value )
int W_8454_DI_L( UINT *DI_LValue )
int W_8454_DI_H( UINT *DI_HValue )
int W_8454_DI_bit( int Bit_No )
```

Visual Basic

Windows 3.11 Version:

```
W_8454_DI (DI_Value As Integer) As Integer
W_8454_DI_L (DI_LValue As Integer) As Integer
W_8454_DI_H (DI_HValue As Integer) As Integer
W_8454_DI_bit (ByVal Bit_No As Integer) As Integer
```

Win-95 or Win-NT Version:

```
W_8454_DI (DI_Value As Long) As Long
W_8454_DI_L (DI_LValue As Long) As Long
W_8454_DI_H (DI_HValue As Long) As Long
W_8454_DI_bit (ByVal Bit_No As Long) As Long
```

C/C++ (DOS)

```
int _8454_DI( UINT *DI_Value )
int _8454_DI_L( UINT *DI_LValue )
int _8454_DI_H( UINT *DI_HValue )
int _8454_DI_bit( int Bit_No )
```

@ *Arguments*

UINT *DI_value : the 16 bits digital input value of port BASE+4 and BASE+5

UINT *DI_Lvalue : the digital input value of port BASE+4, note that only the low 8 bits is effective.

UINT *DI_Hvalue : the digital input value of port BASE+5, note that only the low 8 bits is effective.

int Bit_No : the bit channel number to be read back, the value should be in the range of 0 ~ 15

@ *Return Value*

Always no error for _8454_DI(), _8454_DI_L(), _8454_DI_H()

For _8454_DI_bit():

Invalid_Bit_Number (0xFF) : argument out of range

0 or 1 : the return value of the digital input channel

5.9 8454 Mode Functions

@ *Description*

This is a macro, instead of using function, which is used to program the mode of the counters. Refer the 8454.H for the definition of these macros The macro name are designed for easy to memorize or to be referenced. User can use these macros instead of using the '_8454_Write_Counter()' function.

@ *Syntax*

Microsoft C/C++ (Windows, Win95 & Win-NT)

```
W_8454_Interrupt_on_TC(CN,Val) == W_8454_Write_Counter( CN, 0, Val )
W_8454_One_Shoot(CN,Val)     == W_8454_Write_Counter( CN, 1, Val )
W_8454_Square_Wave(CN,Val)   == W_8454_Write_Counter( CN, 2, Val )
W_8454_Rate_Generator(CN,Val) == W_8454_Write_Counter( CN, 3, Val )
W_8454_SW_Strobe(CN,Val)     == W_8454_Write_Counter( CN, 4, Val )
W_8454_HW_Strobe(CN,Val)     == W_8454_Write_Counter( CN, 5, Val )
```

C/C++ (DOS)

```
_8454_Interrupt_on_TC(CN,Val) ==  _8454_Write_Counter( CN, 0, Val )
_8454_One_Shoot(CN,Val)       == _8454_Write_Counter( CN, 1, Val )
_8454_Square_Wave(CN,Val)     == _8454_Write_Counter( CN, 2, Val )
_8454_Rate_Generator(CN,Val)  == _8454_Write_Counter( CN, 3, Val )
_8454_SW_Strobe(CN,Val)       == _8454_Write_Counter( CN, 4, Val )
_8454_HW_Strobe(CN,Val)       == _8454_Write_Counter( CN, 5, Val )
```

@ *Arguments*

int CN	: Counter number, equals to 1~12
UINT Val	: The 16 bits counter value to write to the counter.

@ *Return Value*

No_Error	: No error
Invalid_Counter_No	: CntrNo is out of range.
Invalid_Timer_Mode	: Mode is out of range

@ *Example*

See 'DEMO.C' in library diskette

5.10 W_8454_INT_Enable

@ *Description*

This function is only available in Window 95 driver and Windows NT driver. The function is used to initialize and start up the interrupt control. After calling this function, every time an interrupt request signal generated, a software event is signaled. So that in your program, your can use wait operation to wait for the event. When the event is signaled, it means an interrupt is generated. Please refer to the samples program 8454int.c.

Note : The W_8454_INT_Enable and W_8454_INT_Disable are a pair of functions. That is, as the W_8454_INT_Enable is called, the W_8454_INT_Disable has to follow up behind it. Otherwise, the interrupt operation will not stop.

@ *Syntax*

Microsoft C/C++ (Win95 & Win-NT)

```
int W_8454_INT_Enable(int irq_no, int irql, HANDLE
*phEvent)
```

Visual Basic (Win95 & Win-NT)

```
W_8454_INT_Enable (ByVal irq_no As Long, ByVal irql
As Long, phEvent As Long) As Long
```

@ *Arguments*

irq_no : IRQ channel selected

```
1: Higher IRQ (From /COUT6)
2: Lower IRQ (From COUT12 or Ext. Int)
```

irql : The IRQ level of the IRQ channel specified in argument irq_no.

phEvent: The handle of the event for interrupt signals.

@ *Return Value*

NoError
INT_Not_Set

5.11 W_8454_INT_Disable

@ ***Description***

This function is only available in Window 95 driver and Windows NT driver. This function is used to stop interrupt signal generation.

Note : This function has to be called after the W_8454_INT_Enable is called.

@ ***Syntax***

Microsoft C/C++ (Win-95 & Win-NT)

```
int W_8454_INT_Disable(int irq_no)
```

Visual Basic (Win-95 & Win-NT)

```
W_8454_INT_Disable (ByVal irq_no As Long) As Long
```

@ ***Arguments***

irq_no : IRQ channel selected

```
1: Higher IRQ (From /COUT6)
2: Lower IRQ (From COUT12 or Ext. Int)
```

@ ***Return Value***

NoError
INT_Not_Set

Appendix 8254 Timer Chip Operation

The ACL-8454 has at most four interval 8254 chips on board. Refer to section 3.5 for the signal connection and the configuration of the counters. The following sections describe the details of the 8254 chip.

The 8254 Timer / Counter Chip

The Intel (NEC) 8254 contains three independent, programmable, multi-mode 16 bit counter/timers. The three independent 16 bit counters can be clocked at rates from DC to 5 MHz. Each counter can be individually programmed with 6 different operating modes by appropriately formatted control words. The most commonly uses for the 8254 in microprocessor based system are:

- programmable baud rate generator
- event counter
- binary rate multiplier
- real-time clock
- digital one-shot
- motor control

For more information about the 8254 , please refer to the NEC Microprocessors and peripherals or Intel Microsystems Components Handbook.

I/O Address

The 8254 in the ACL-8454 occupies 4 I/O address as shown below. Although there are four 8254 chips on board, however, only one chip is selected in one moment. The programming of 8254 is control by the registers BASE+0 to BASE+3. The functionality of each register is specified in the following sections. For more detailed information, please refer handbook of 8254 chip.

BASE + 0	LSB OR MSB OF COUNTER 0
BASE + 1	LSB OR MSB OF COUNTER 1
BASE + 2	LSB OR MSB OF COUNTER 2
BASE + 3	CONTROL BYTE

Control Byte

Before loading or reading any of these individual counters, the control byte (BASE+3) must be loaded first. The format of the control byte is :

Bit	7	6	5	4	3	2	1	0
	SC1	SC0	RL1	RL0	M2	M1	M0	BCD

- SC1 & SC0 - Select Counter (Bit 7 & Bit 6)

SC1	SC0	COUNTER
0	0	Select Counter 0
0	1	Select Counter 1
1	0	Select Counter 2
1	1	ILLEGAL

- RL1 & RL0 - Select Read/Load operation (Bit 5 & Bit 4)

RL1	RL0	OPERATION
0	0	COUNTER LATCH FOR STABLE READ
0	1	READ/LOAD LSB ONLY
1	0	READ/LOAD MSB ONLY
1	1	READ/LOAD LSB FIRST, THEN MSB

- M2, M1 & M0 - Select Operating Mode (Bit 3, Bit 2, & Bit 1)

M2	M1	M0	MODE
0	0	0	0
0	0	1	1
x	1	0	2
x	1	1	3
1	0	0	4
1	0	1	5

- BCD - Select Binary/BCD Counting (Bit 0)

0	16-BITS BINARY COUNTER
1	BINARY CODED DECIMAL (BCD) COUNTER (4 DIGITAL)
Note	The count of the binary counter is from 0 up to 65,535 and the count of the BCD counter is from 0 up to 9,999

Mode Definitions

In 8254, six operating modes can be selected. they are :

- **Mode 0** : Interrupt on Terminal Count
- **Mode 1** : Programmable One-Shot.
- **Mode 2** : Rate Generator.
- **Mode 3** : Square Wave Rate Generator.
- **Mode 4** : Software Triggered Strobe.
- **Mode 5** : Hardware Triggered Strobe.

All detailed description of these six modes are written in Intel Microsystem Components Handbook Volume II Peripherals.

Timer / Counter Applications

Please refer to Chapter 4.

Product Warranty / Service

Seller warrants that equipment furnished will be free form defects in material and workmanship for a period of one year from the confirmed date of purchase of the original buyer and that upon written notice of any such defect, Seller will, at its option, repair or replace the defective item under the terms of this warranty, subject to the provisions and specific exclusions listed herein.

This warranty shall not apply to equipment that has been previously repaired or altered outside our plant in any way as to, in the judgment of the manufacturer, affect its reliability. Nor will it apply if the equipment has been used in a manner exceeding its specifications or if the serial number has been removed.

Seller does not assume any liability for consequential damages as a result from our products uses, and in any event our liability shall not exceed the original selling price of the equipment.

The equipment warranty shall constitute the sole and exclusive remedy of any Buyer of Seller equipment and the sole and exclusive liability of the Seller, its successors or assigns, in connection with equipment purchased and in lieu of all other warranties expressed implied or statutory, including, but not limited to, any implied warranty of merchant ability or fitness and all other obligations or liabilities of seller, its successors or assigns.

The equipment must be returned postage-prepaid. Package it securely and insure it. You will be charged for parts and labor if you lack proof of date of purchase, or if the warranty period is expired.