5300 User's Manual

Doc. #02793 Rev 0498

OCTAGON SYSTEMS CORPORATION®

6510 W. 91st Ave. Westminster, CO 80030
Tech. Support: 303–426–4521

COPYRIGHT

TRADEMARKS

Micro PC™, PC SmartLink™, Octagon Systems Corporation®, the Octagon logo and the Micro PC logo, and CAMBASIC™ IV are trademarks of Octagon Systems Corporation. QuickBASIC® is a registered trademark of Microsoft Corporation. Turbo C™ is a trademark of Borland International, Inc.

NOTICE TO USER

The information contained in this manual is believed to be correct. However, Octagon assumes no responsibility for any of the circuits described herein, conveys no license under any patent or other right, and makes no representations that the circuits are free from patent infringement. Octagon makes no representation or warranty that such applications will be suitable for the use specified without further testing or modification.

Octagon Systems Corporation general policy does not recommend the use of its products in life support applications where the failure or malfunction of a component may directly threaten life or injury. It is a Condition of Sale that the user of Octagon products in life support applications assumes all the risk of such use and indemnifies Octagon against all damage.

IMPORTANT!

Please read before installing your product.

Octagon's products are designed to be high in performance while consuming very little power. In order to maintain this advantage, CMOS circuitry is used.

CMOS chips have specific needs and some special requirements that the user must be aware of. Read the following to help avoid damage to your card from the use of CMOS chips.

Using CMOS Circuitry in Industrial Control

Industrial computers originally used LSTTL circuits. Because many PC components are used in laptop computers, IC manufacturers are exclusively using CMOS technology. Both TTL and CMOS have failure mechanisms, but they are different. This section describes some of the common failures which are common to all manufacturers of CMOS equipment. However, much of the information has been put in the context of the Micro PC.

Octagon has developed a reliable database of customer-induced, field failures. The average MTBF of Micro PC cards exceeds 11 years, yet there are failures. Most failures have been identified as customer-induced, but there is a small percentage that cannot be identified. As expected, virtually all the failures occur when bringing up the first system. On subsequent systems, the failure rate drops dramatically.

- Approximately 20% of the returned cards are problem-free. These cards, typically, have the wrong jumper settings or the customer has problems with the software. This causes frustration for the customer and incurs a testing charge from Octagon.

- Of the remaining 80% of the cards, 90% of these cards fail due to customer misuse and accident. Customers often cannot pinpoint the cause of the misuse.

- Therefore, 72% of the returned cards are damaged through some type of misuse. Of the remaining 8%, Octagon is unable to determine the cause of the failure and repairs these cards at no charge if they are under warranty.

The most common failures on CPU cards are over voltage of the power supply, static discharge, and damage to the serial and parallel ports. On expansion cards, the most common failures are static discharge, over voltage of inputs, over current of outputs, and misuse of the CMOS circuitry with regards to power supply sequencing. In the case of the video cards, the most common failure is to miswire the card to the flat panel display. Miswiring can damage both the card and an expensive display.

- **Multiple component failures** - The chance of a random component failure is very rare since the average MTBF of an Octagon card is greater than 11 years. In a 7 year study,

Octagon has never found a single case where multiple IC failures were not caused by misuse or accident. It is very probable that multiple component failures indicate that they were user-induced.

- **Testing "dead" cards** - For a card that is "completely nonfunctional", there is a simple test to determine accidental over voltage, reverse voltage or other "forced" current situations. Unplug the card from the bus and remove all cables. Using an ordinary digital ohmmeter on the 2,000 ohm scale, measure the resistance between power and ground. Record this number. Reverse the ohmmeter leads and measure the resistance again. If the ratio of the resistances is 2:1 or greater, fault conditions most likely have occurred. A common cause is miswiring the power supply.

- **Improper power causes catastrophic failure** - If a card has had reverse polarity or high voltage applied, replacing a failed component is not an adequate fix. Other components probably have been partially damaged or a failure mechanism has been induced. Therefore, a failure will probably occur in the future. For such cards, Octagon highly recommends that these cards be replaced.

- **Other over-voltage symptoms** - In over-voltage situations, the programmable logic devices, EPROMs and CPU chips, usually fail in this order. The failed device may be hot to the touch. It is usually the case that only one IC will be overheated at a time.

- **Power sequencing** - The major failure of I/O chips is caused by the external application of input voltage while the Micro PC power is off. If you apply 5V to the input of a TTL chip with the power off, nothing will happen. Applying a 5V input to a CMOS card will cause the current to flow through the input and out the 5V power pin. This current attempts to power up the card. Most inputs are rated at 25 mA maximum. When this is exceeded, the chip may be damaged.

- **Failure on power-up** - Even when there is not enough current to destroy an input described above, the chip may be destroyed when the power to the card is applied. This is due to the fact that the input current biases the IC so that it acts as a forward biased diode on power-up. This type of failure is typical on serial interface chips.

- **Serial and parallel** - Customers sometimes connect the serial and printer devices to the Micro PC while the power is off. This can cause the failure mentioned in the above section, *Failure upon power-up*. Even if they are connected with the Micro PC on, there can be another failure mechanism. Some serial and printer devices do not share the same power (AC) grounding. The leakage can cause the serial or parallel signals to be 20-40V above the Micro PC ground, thus, damaging the ports as they are plugged in. This would not be a problem if the ground pin is connected first, but there is no guarantee of this. Damage to the printer port chip will cause the serial ports to fail as they share the same chip.

- **Hot insertion** - Plugging cards into the card cage with the power on will usually not cause a problem. (**Octagon urges that you do not do this!**) However, the card may be damaged if the right sequence of pins contacts as the card is pushed into the socket. This usually damages bus driver chips and they may become hot when the power is applied. This is one of the most common failures of expansion cards.

- **Using desktop PC power supplies** - Occasionally, a customer will use a regular desktop PC power supply when bringing up a system. Most of these are rated at 5V at 20A or more. Switching supplies usually require a 20% load to operate properly. This means 4A or more. Since a typical Micro PC system takes less than 2A, the supply does not regulate properly. Customers have reported that the output can drift up to 7V and/or with 7-8V voltage spikes. Unless a scope is connected, you may not see these transients.

- **Terminated backplanes** - Some customers try to use Micro PC cards in backplanes that have resistor/capacitor termination networks. CMOS cards cannot be used with termination networks. Generally, the cards will function erratically or the bus drivers may fail due to excessive output currents.

- **Excessive signal lead lengths** - Another source of failure that was identified years ago at Octagon was excessive lead lengths on digital inputs. Long leads act as an antenna to pick up noise. They can also act as unterminated transmission lines. When 5V is switch onto a line, it creates a transient waveform. Octagon has seen submicrosecond pulses of 8V or more. The solution is to place a capacitor, for example 0.1 µF, across the switch contact. This will also eliminate radio frequency and other high frequency pickup.

TABLE OF CONTENTS

PREFACE

This manual is a guide to the proper configuration and operation of your 5300 Counter/Timer Card. Installation instructions, card mapping information and jumpering options are described in the main body of the manual; the schematic and technical specifications are included in the appendices.

NOTE: Additional application information is available on the Octagon Bulletin Board: Read AN–0060. (303–427–5368)

The 5300 provides timing, counting, frequency measurement, frequency generation, and pulse width modulation. It is designed to be used with any Octagon Micro PC control card. This combination provides a modular system which is easy to set up, modify and use. You can also use your 5300 in conjunction with other Micro PC expansion cards, allowing you to tailor your system for a wide variety of applications.

CONVENTIONS USED IN THIS MANUAL

1. Information which appears on your screen (output from your system or commands or data that you key in) is shown in a different type face.

   Example 1:

   ```
   Octagon 5066 ROM BIOS Vers X.XX
   Copyright (c) 1998, Octagon Systems, Corp.
   All Rights Reserved
   ```

   Example 2:

   ```
   Press the <ESC> key.
   ```

2. Italicized refers to information that is specific to your particular system or program, for example,

   Enter *filename*

   means enter the name of your file. Names of other sections or manuals are also italicized.

3. Warnings always appear in this format:

   WARNING: The warning message appears here.

4. Paired angle brackets are used to indicate a specific key on your keyboard, for example, <ESC> means the escape key; <CTRL> means the control key; <F1> means the F1 function key.

5. All addresses are given in hexadecimal.

SYMBOLS AND TERMINOLOGY

Throughout this manual, the following symbols and terminology are used:

W[–]	Denotes a jumper block and the pins to connect.
NOTE	Information under this heading presents helpful tips for using the 5300.
WARNING:	Information under this heading warns you of situations which might cause catastrophic or irreversible damage.
PWM	Pulse Width Modulation.
TTL Compatible	0–5V logic levels.
H	The suffix “H” denotes a hexadecimal number. For example, 1000H in hexadecimal equals 4096 in decimal.

TECHNICAL SUPPORT

If you have a question about the 5300 Counter/Timer Card and can't find the answer in this manual, call Technical Support. They will be ready to give you the assistance you need.

When you call, please have the following at hand:

- Your *5300 Counter/Timer Card User's Manual*
- A description of your problem

The direct line to Technical Support is 303–426–4521 and the e–mail address is support@octagonsystems.com.

This page intentionally left blank.

DESCRIPTION

The 5300 Counter/Timer I/O Card provides timing, counting, frequency measurement, frequency generation, and pulse width modulation (PWM). The 5300 can be used for precision timing, speed measurement, event counting, and related tasks. For example, you can use the 5300 to control pick and place machines; to measure and adjust motor speed; or to count objects on a conveyor belt.

The 5300 measures 4.5 in. x 4.9 in. and uses one slot of the Micro PC card cage. It is compatible with all Micro PC Control Cards. You can use as many cards as space in the card cage will allow.

The 5300 uses three 82C54 chips for counting and timing. An 82C55 chip provides eight general–purpose digital I/O lines. The 5300 has two programmable time bases, six counters and six counter/timer inputs. Three of the counter inputs are optically isolated for greater safety and less interference from noise.

The counter/timers operate at 5V (TTL) levels with signal conditioning to minimize noise. They have gate inputs and counter outputs. All clock inputs go through Schmitt trigger buffers.

MAJOR FEATURES

Time Base 0 of the two programmable time bases is an 8 MHz source with a 16–bit divider. It can generate frequencies from 122 Hz to 4 MHz. The other time base has an 8 MHz source and is cascaded to create a 32–bit divider. Output frequencies range from 0.0005 Hz to 2 MHz.

Counter/Timers

The six counter/timer sections have clock, gate, and output leads available. Different modes of operation are possible by using various clock configurations. The lines are terminated in 10– and 34–pin IDC connectors. Every other lead is ground, which minimizes the chance of cross coupling and noise. The count inputs have signal conditioners to minimize noise effects. These inputs and the gate inputs have pull–up resistors.

Digital I/O

The 5300 has eight digital I/O lines that can be used for general purpose control and input. The lines can be programmed as inputs or outputs in groups of four.

Interrupts

A jumper determines whether the output of a counter or one of the digital I/O lines is the interrupt source. The jumper can also be configured so that there is no interrupt source. The selected source can be jumpered to IRQ2 through IRQ7.

Access Indicator

An LED indicator flashes briefly whenever the 5300 is accessed. This is useful when troubleshooting the card to know if a communications link has been established.

Opto–isolated inputs

Three counter inputs are optically isolated and terminated in a screw terminal block. The isolation between channels and ground is 2500V. However, for operator safety, no more than 32V should be applied.

Counters

The maximum count rate input is 1 MHz. The input range is 4 to 12 volts but can be increased up to 48V when used in conjunction with an external resistor.

CHAPTER 2 INSTALLATION

This chapter includes information on setting the base address, selecting the interrupt request line, and installing the 5300 in the Micro PC card cage. Refer to the *Operation* chapter for more information.

The 5300 Counter/Timer I/O Card uses one slot of the Micro PC card cage. It may be used with any Micro PC Control Card. You can use as many cards as space in the card cage will allow.

WARNING: The 5300 contains static sensitive CMOS components. The greatest danger occurs when the card is plugged into a card cage. The 5300 becomes charged by the user and the static discharges to the backplane from the pin closest to the card connector. If that pin happens to be an input pin, even TTL inputs may be damaged. To avoid damaging your card and its components:

1. Ground yourself before handling the 5300 Counter/Timer Card.
2. Disconnect power before removing or inserting the 5300 Card.

EQUIPMENT

You will need the following equipment (or equivalent) to use your 5300.

- 5300 Counter/Timer Card
- Micro PC Control Card
- Micro PC Card Cage
- Power Supply or Module
- Software appropriate for your system

INSTALLATION

Before installing the 5300, refer to Figure 2.1 for the location of various connectors and jumpers.

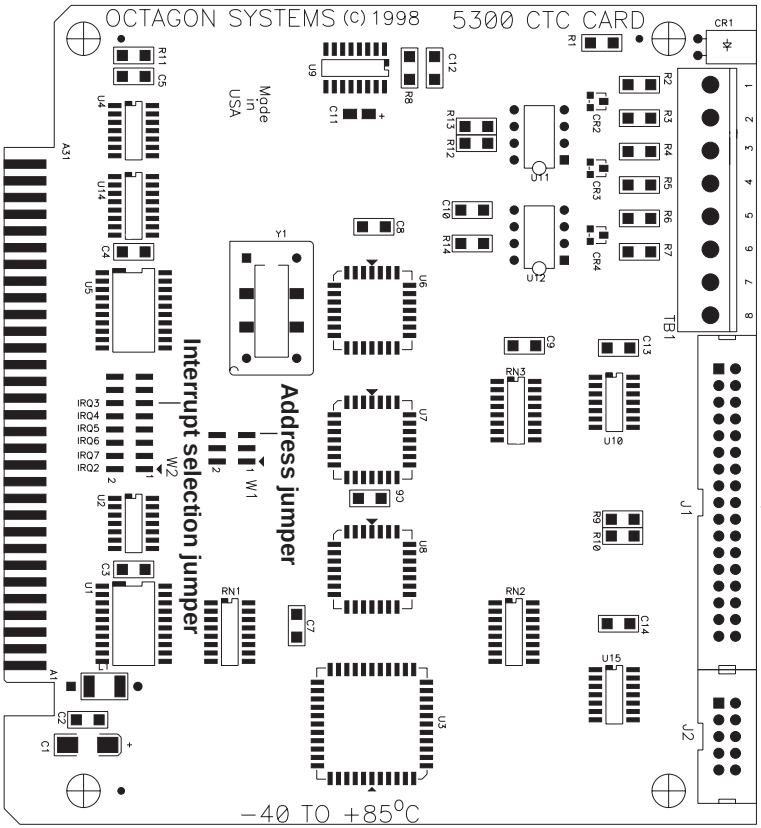


Figure 2–1—5300 Component Diagram

Base Address

Jumper block W1 determines the base address of the 5300. As shipped, the address is set to 100H. If there is another card in your system with a base address of 100H, you must use a different base address for the 5300 or the other card.

To change the base address, change the jumper connections in block W1. Connect the appropriate pins with push–on connectors. The following table lists the jumper connections and corresponding base addresses.

5300 Base Address Select: W1	
Pins Jumpered	**Base I/O Address**
[1-2][3-4][5-6]	100H*
[3-4][5-6]	110H
[1-2][5-6]	120H
[5-6]	130H
[1-2][3-4]	140H
[3-4]	150H
[1-2]	160H
No jumpers	170H

* = default

INTERRUPT SOURCE & INTERRUPT REQUEST LINES

As shipped, the card does not have any interrupt lines selected. In this configuration, the pins in jumper block W2 are jumpered for no interrupts. The following tables list the jumper configurations and corresponding interrupt request lines and sources. Connect appropriate pins with push–on connectors.

There are two possible interrupt sources on the 5300: The CTC 2 Output 0 signal, or the PIO 0 digital signal when configured as an output. You can choose either signal as a source, but you can not use both signals at the same time. Select the desired interrupt source via jumper block W2.

There are six interrupt request lines, IRQ2 through IRQ 7, which go to the bus. Pins 1 through 12 of jumper block W2 select the interrupt request line; pins 13–16 select the interrupt source.

You will need to write your own interrupt handler for either interrupt source. Refer to Example 1 programming example in the *Applications* chapter.

5300 Interrupt Request Lines: W2	
Pins Jumpered	**IRQ**
[1-2]	IRQ2
[11-12]	IRQ3
[9-10]	IRQ4
[7-8]	IRQ5
[5-6]	IRQ6
[3-4]	IRQ7
[1-3]	No interrupt selected*

* = default

5300 Interrupt Sources: W2	
Pins Jumpered	**Base I/O Address**
[13-14]	PIO 0 Output
[15-16]	CTC2 Output 0
[14-16]	No interrupt selected*

* = default

Installing the Card in the Card Cage

WARNING: Take care to correctly position the 5300 in the card cage. The V_{CC} and ground signals must match those on the backplane. Figure 2–2 shows the relative position of the 5300 as it is installed in the card cage.

1. Turn card cage power off.

2. Position the cage so that the backplane is away from you, the power module is to the right, and the open side of the cage is closest to you. The lettering on the backplane should be right side up (for example, you should be able to read “A31” on the backplane), with the words OCTAGON SYSTEMS CORP. running vertically along the left side of the backplane. This position is “feet down” for a table mount cage and “feet back” for a panel mount.

3. Slide the card into the card cage. The components on the card should face to the left. The lettering on the card (Octagon Systems Corp.) should be on the top edge of the card and the gold contact fingers toward the backplane. (See Figure 2–2)

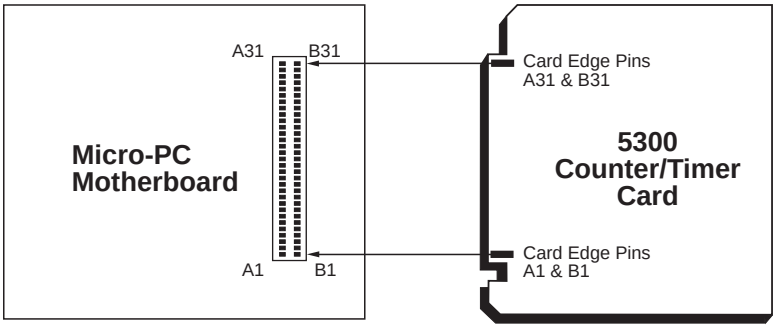


Figure 2–2—Card Edge Orientation

4. Plug the card into the backplane.

5. Turn on power to the card cage.

6. The amber LED will light briefly whenever the card is accessed.

TROUBLESHOOTING

If you have trouble getting your system to work properly, remove all cards except the Control Card and the 5300 card from your system. Check the power module and jumpers. If you still encounter difficulties, please contact Technical Support at either the telephone number, 303–426–4521, or the e–mail address, support@octagonsystems.com.

Power Module

Make sure the power cord is properly connected between the power module and the power source.

Make sure the Control Card and 5300 are receiving power. The power module voltage should be in the 5 to +/–0.25V range when measured at the connector pins. The power module ripple should be less than 50 mV.

Jumper Configurations

The 5300 is shipped with jumper connections in place for Base I/O Address 100H and no interrupts selected. Jumper changes are usually not needed to get the system running. If you changed the jumpers and the system is not working properly, return the system to the original jumper positions. If the system still does not work, contact Technical Support.

DESCRIPTION

The 5300 Counter/Timer Card has three 82C54 counter/timer chips. Each chip has three counters/timers. CTC1 is a general purpose counter/timer; CTC2 has optically isolated clock inputs; CTC3 can be used as a timer only.

All counter/timer I/O lines go to J1 and J2. You can use a UTB–34 to connect external wiring to J1. You can also use an STB–10 to connect external wiring to J2. The clock inputs on CTC1 and J1 are buffered with a Schmitt trigger. The clock inputs on CTC2 come from either the optical isolator and terminal block or directly from J1 (without optical isolation).

The 82C54 counters are all programmable. Generally, you will program the mode (which is the function) and the number of counts which must occur (the variable). For example, you might specify that a certain number of counts must occur before a pulse is generated.

CASCADING

Each counter can count from 0 to 65,535. If you need to count to a higher number, you will need to cascade the counters—connect them so that the count output from one counter goes to the next counter clock input. Connections to allow cascading must be done externally to the card through J1 and J2.

For example, if you need to count to 150,000, physically wire the output from one counter to the clock input of another counter. One way to do this is to connect pin 17 to pin 7 via the UTB–34, which will send the output from counter 2 to counter 1 input on CTC1.

CTC1, CTC2, AND CTC3: ADDRESSING

The CTC1, CTC2, and CTC3 chips on the 5300 card are addressed according to the following table:

5300 Counter/Timer Chip Addresses	
Description	**Address**
CTC1	Base + 04H
CTC2	Base + 08H
CTC3	Base + 0CH

Each CTC has a counter 0, counter 1, counter 2, and control word register, which are all addressed in the following table:

8254 Register Addresses	
Description	**Address**
Counter channel 0	Counter/timer chip address + 00H
Counter channel 1	Counter/timer chip address + 01H
Counter channel 2	Counter/timer chip address + 02H
Control word register	Counter/timer chip address + 03H

For example, the address of CTC1, counter 2, is Base + 06H.

CTC1 AND CTC2: COUNTERS AND TIMERS

The input and output lines of CTC1 and CTC2 connect to J1 and J2 respectively. The three clock inputs of CTC2 are also available from the opto isolators at TB1.

J1: Counter/Timer & Digital I/O Port					
Pin#	**Function**	**Direction**	**Pin#**	**Function**	**Direction**
1	CTC1 Clock 0	In	18	Gnd	
2	Gnd		19	CTC3 Output 0	Out
3	CTC1 Gate 0	In	20	Gnd	
4	Gnd		21	CTC3 Output 2	Out
5	CTC1 Output 0	Out	22	Gnd	
6	Gnd		23	PIO 0	In/Out
7	CTC1 Clock 1	In	24	PIO 1	In/Out
8	Gnd		25	PIO 2	In/Out
9	CTC1 Gate 1	In	26	PIO 3	In/Out
10	Gnd		27	PIO 4	In/Out
11	CTC1 Output 1	Out	28	PIO 5	In/Out
12	Gnd		29	PIO 6	In/Out
13	CTC1 Clock 2	In	30	PIO 7	In/Out
14	Gnd		31	CTC2 Clock 2	In
15	CTC1 Gate 2	In	32	CTC2 Clock 1	In
16	Gnd		33	CTC2 Clock 0	In
17	CTC1 Output 2	Out	34	+5V	

J2: Counter/Timer Port		
Pin#	**Function**	**Direction**
1	CTC3 Output 2 / 2*	Out
2	CTC3 Output 2 / 2	Out
3	CTC2 Output 0	In
4	CTC2 Gate 0	In
5	CTC2 Output 1	Out
6	CTC2 Gate 1	In
7	CTC2 Output 2	Out
8	CTC2 Gate 2	In
9	Gnd	
10	Gnd	

* = Inverted output of pin 2.

CTC2 Optically Isolated Inputs

The optical isolator inputs are available at TB1 and go to the clock inputs on CTC2. You can bypass the isolation by connecting directly to J1.

Use the opto isolators if your environment (plant wiring, for instance) could generate spurious ground loops or differences in potential. Opto isolators are often used for switch inputs and magnetic pick–ups in electrically noisy conditions, or for anything that needs to be electrically isolated.

The isolation voltage is 48V. Opto isolator inputs require 4 to 12 volts to trigger the clock. You can use higher voltages provided that you limit the current to 15 mA by placing a resistor in series with the input. The following table shows which resistor to use with different voltage levels.

5300 Input Voltage Resistors	
Voltage	**Resistor**
4.0V - 12V	No resistor
5.2V - 16V	150 ohms, 0.125W or greater
7.4V - 24V	430 ohms, 0.5W or greater
14.2V - 48V	1300 ohms, 2W or greater

Inputs are not polarity sensitive. You can apply + and – to any input pair.

The following table shows the TB1 terminal block inputs and corresponding clock inputs on CTC2.

TB1: Terminal Block, Opto–Isolated Inputs		
Pin#	**Function**	**Direction**
1	–CTC2 Clock 2	In
2	+CTC2 Clock 2	In
3	+CTC2 Clock 1	In
4	–CTC2 Clock 1	In
5	+CTC2 Clock 0	In
6	–CTC2 Clock 0	In
7	GND	
8	+5V	

CTC1 and CTC2 Gates

Gate inputs are used to turn the counter/timer on and off. They are all TTL (0–5V) compatible; all have 10K pull–up resistors. The gate inputs are not isolated. Two of the gate inputs on CTC1 go to port A of the 82C55A PIO chip.

CTC3: PRECISION TIMER

CTC3 is configured as a programmable precision timer. The 8 MHz crystal oscillator provides high resolution, so you can precisely time events. CTC3 has two output lines which connect to J1; no other counter lines are available. Although it is used exclusively as a timer, you must still specify the counter mode. Two of the counters in CTC3 are cascadable, so you can get longer (i.e., lower frequency) time periods. For example, you can create a large frequency divider by connecting the output from CTC3 Output 0 (pin 19 on J1) to CTC1 Clock 0 (pin 1 on J1). This is useful for PWM output and can result in high resolution even at very low frequencies. Precision with this system can be as fine as 0.1%.

Since the time interval can be programmed very precisely, you can create a finely–tuned PWM output. For example, instead of specifying that the pulse be on for 1 mS and off for 9 mS, you can specify that it be on for .01 mS and off for .09 mS—maintaining the same ratio but achieving a much finer resolution.

You can also configure CTC2 Output 0 as the interrupt source. Connect the output of CTC3 Output 0 (pin 19 on J1) to CTC2 Clock 0 (pin 33 on J1). The output from CTC2 Output 0 goes to jumper block W2, which determines the interrupt select line (IRQ2 – IRQ7). Using your interrupt handler, you can process precisely timed interrupts. Refer to example 1 in the *Applications* chapter.

82C55A: DIGITAL I/O

82C55A general purpose, digital input/output IC. It has three 8–bit ports and a control port which is addressed according to the following table:

5300 Port Addresses	
Description	**Address**
Port A	Base + 00H
Port B	Base + 01H
Port C	Base + 02H
Control Register	Base + 03H

On power–up or reset, all three ports are in the input state. This configuration will not cause any operational problems or interference with the counters.

You can alter which ports are inputs or outputs by writing a control command to the control register in the 82C55A. The examples below assume the base address is 100H. The following table lists the control commands for the different input/output possibilities.

5300 I/O Control Register Commands					
HEX	**DEC**	**Port A***	**Port B***	**Upper Port C***	**Lower Port C***
90H	144	IN	OUT	OUT	OUT
91H	145	IN	OUT	OUT	IN
92H	146	IN	IN	OUT	OUT
93H	147	IN	IN	OUT	IN
98H	152	IN	OUT	IN	OUT
99H	153	IN	OUT	IN	IN
9AH	154	IN	IN	IN	OUT
9BH	155	IN	IN	IN	IN

*Port A must be an input. Port B can be inputs or outputs. Each half of Port C is programmable. Port C may be either inputs or outputs. Upper C includes bits 4 through 7 and Lower C includes bits 0 through 3.

For example, if you want ports B and C to be outputs while retaining port A as inputs, use:

```
OUT 103H,90H
```

Port B will now output all “1”s after:

```
OUT 101H,FFH
```

or all “0”s after:

```
OUT 101H,0
```

Output Drive Capabilities

The output drive capacity of the 82C55A is one TTL load. Any output on port A, B, or C can sink or source 2.5 mA.

Recommended Configuration for 82C55A

Port A: all inputs
Port B: all inputs unless you intend to control all the gates on CTC2
Port C: inputs, outputs, or both.

NOTE: Port A must be set as inputs.

COUNTER/TIMER AND DIGITAL I/O INTERFACE

The 24 programmable I/O lines of 82C55A are divided into three addressable ports of 8 bits each plus one port which contains the control register. The control register is used to determine the configuration of the 24 lines. Refer to the Control Register command table in this chapter.

All outputs from CTC1 and CTC2 go to port A of the 82C55A. Port A must be programmed as an input.

5300 Port A Line Signals		
Port A Line	**Direction**	**Counter/Timer**
0	In	CTC1 Output 0
1	In	CTC1 Output 1
2	In	CTC1 Output 2
3	In	CTC2 Output 0
4	In	CTC2 Output 1
5	In	CTC2 Output 2
6	In	CTC1 Gate 0
7	In	CTC1 Gate 1

Ports A and B of 82C55A can be used to read gate and output status on CTC1 and CTC2. If you configure port B of 82C55A as an output, you can use Port B to control the gate inputs on CTC2.

5300 Port B Line Signals		
Port B Line	**Direction**	**Counter/Timer**
0	Out	CTC2 Gate 0
1	Out	CTC2 Gate 1
2	Out	CTC2 Gate 2
3-7		No Connection

Port C of 82C55A provides eight general purpose I/O lines which go to J1 and can be programmed as all inputs, all outputs, or four lines of each (upper and lower C).

5300 Port C		
Port C Line	**Direction**	**J1 Pin #**
0	In/Out	23
1	In/Out	24
2	In/Out	25
3	In/Out	26
4	In/Out	27
5	In/Out	28
6	In/Out	29
7	In/Out	30

EXAMPLE 1: TICK TIMER

You can also configure CTC2 Output 0 as the interrupt source. Connect the output of CTC3 Output 0 (pin 19 on J1) to CTC2 Clock 0 (pin 33 on J1). The output from CTC2 Output 0 goes to jumper block W2, which determines the interrupt select line (IRQ2 – IRQ7). Using your interrupt handler, you can process precisely timed interrupts.

The following example is a demonstration program which sets up the 5300 to produce a 100 Hz tick timer.

NOTE: The hard disk interrupt is used for this demo. If you are using a hard disk, you must change the interrupt to another number.

NOTE: Additional application information is available on the Octagon Bulletin Board: Read AN–0060 (303–427–5368)

```
/* Filename:    5300tick.c

Function name:  5300tick

Edit date:      10/29/90

Function:
          Demonstration program setting up the 5300
          to produce a 100 Hz tick timer.

          NOTE: The hard disk interrupt is used for
          this demo. If you are using a hard disk,
          you must change the interrupt to another
          number.

          Compiled using Borland's Turbo–C, version
          2.0.

          Settings for the 5300:
            Connect J1–19 to J1–33. Set jumpers
            W2[15–16] and W2[7–8] to enable inter-
            rupts

Revisions:

*/
```

```
/* Includes */

#include     "stdio.h"
#include     "dos.h"
#include     "time.h"

/* Definitions */
/* set up base address of 5300 */

#define BASE_5300       0x100

/* define offset addresses of components*/

#define PIO_5300        0 + BASE_5300

/* counter/timers 1 through 3   */

#define CTC1 4+BASE_5300
#define CTC2 8+BASE_5300
#define CTC3 12+BASE_5300

/* Declarations */

/* counter for interrupt                */

int     irqcnt;

/* interrupt number                     */

int     itrnum = 0x0d;

/* pointer to old interrupt             */

void interrupt (*oldint)();

/* flag for interrupt routine to stop initializing
the tick counter */

int     notick;

/* Program */

/* interrupt handler                    */

/* This is the program that processes the inter-
rupt. It must go before main when using Turbo–C and
Quick–C. */
```

```
void interrupt tick() {

/* This example uses IRQ5 which is the hard disk
interrupt. If you are using a hard disk, you may
want to use another unused IRQ port. */

/* Check to see if we should continue to process
this interrupt. If not, simply exit */

if (notick == 0) {

/* send count to port C output, line 0 on the 5300.
This output should toggle at a 50 Hz rate */

        outportb(PIO_5300+2,irqcnt);  irqcnt++;

/* reset counter to 0 if overflow        */

        if(irqcnt == 256)
        irqcnt = 0;

/* re-initialize counter                */

        outportb(CTC2,232);  outportb(CTC2,3);

/* reset interrupt port                 */

        outportb(0x20,0x20);

}

}

main() {

/* use the hard disk interrupt IRQ5 (0DH) transfer
control on each tick. */

/* get & save old interrupt vector      */

oldint = getvect(itrnum);

/* set interrupts                       */
setvect(itrnum,tick);

/* set 5300  for 100 times/second        */
```

```
/* set CTC3O0 to mode 2 ( rate generator) and to
divide by 80 */

outportb(CTC3+3,0x34);

/* Divide by 80.  LSB first, then MSB   */

outportb(CTC3,80);  outportb(CTC3,0);

/* set CTC2O0 for mode 4 (triggered strobe) and to
divide by 1000 (3 * 256 + 232 = 1000)          */

outportb(CTC2+3,0x38);  outportb(CTC2,232);
outportb(CTC2,3);

/* set up 82C55A port on 5300 for A & B as inputs,
C as outputs   */

outportb(PIO_5300+3,0x92);

/* set up for interrupts                */

outportb(0x21,0x98);

/* main program loop.  This example simply waits
for a key press. Your program would continue. */

/* check for a keyboard hit.  If one, then exit
program */

puts("Waiting for keyboard hit.\n"); while (kbhit()
== 0)
            ;

/* tell the interrupt handler not to reset the
timer and get the character */

notick = 1;
getch();

/* restore interrupt handler            */
setvect(itrnum,oldint);
}
```

EXAMPLE 2: PULSE WIDTH MODULATION

This program uses 3 of the 9 counter/timers on the 5300 card to generate a pulse width modulated output. The first stage, CTC3 Counter 3, is used as a simple 'divide by 8' counter in the 'SQUARE WAVE' mode. This produces a 1 MHz square wave output that is used as the CLOCK input for the other two stages. The second stage, CTC 1 Counter 2, determines the output frequency. The third stage, CTC1 Counter 1, is used in the One Shot mode providing the modulating element.

In the QuickBASIC example program, the following variables are used:

- FR = Requested frequency

  FR is the frequency to be modulated. This frequency has a maximum and minimum value. FR can be any frequency multiple of 1 MHz that is in the range of 16 Hz to 500 KHz.

- FD = Frequency Divisor for the requested frequency

  FD is a value calculated in the program that is an integer in the range of 2 to 65,535.

- DU = Requested Duty cycle (that the output signal is true)

  DU represents the Duty Cycle in percent. The time that the output signal is ON verses the total available time, defines the duty cycle.

- DX = Decimal equivalent for duty cycle

  DX is calculated in the program and is based on the input 'DU'.

- PD = Pulse width Divisor.

  PD is a value calculated by the program that is an integer in the range of 2 to 65,535.

Note the relationship between:

- the 1 MHz clock,
- the frequency and
- the resolution of the modulation.

The resolution is expressed as follows:

RESOLUTION(%) = FREQUENCY / 1 MHz

A 100 KHz frequency can only be modulated in 10% steps. A 1 KHz frequency can be modulated in 0.1% percent steps. The lower the value of the output frequency the higher the resolution. Also, note that the frequency of the output is rounded to a value that will equal 1 MHz when multiplied by the integer 'FD'.

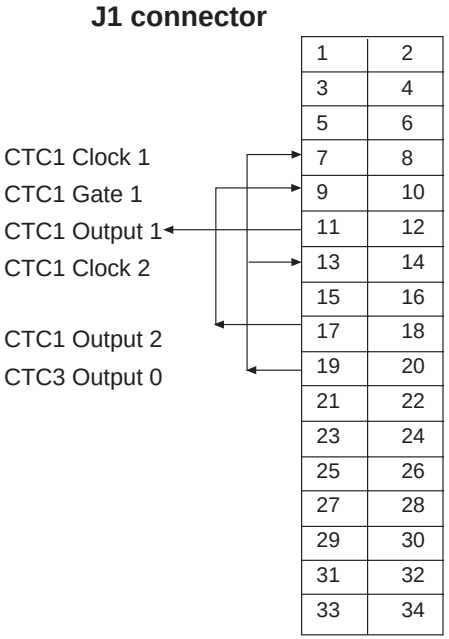


Figure 4–1—Counter Timer Wiring Connection

' These are the connections which must be made in order for the Pulse Width Modulator to work. The connections are as follows:

' CTC3 Ch0 'OUT' (J1-19) to CTC1 Ch1 'CLOCK' (J1-7) & CTC1 Ch2 'CLOCK' (J1-13)
' CTC1 Ch2 'OUT' (J1-17) to CTC1 Ch1 'GATE' (J1-9)
' CTC1 Ch1 'OUT' (J1-11) is the Modulated Signal'

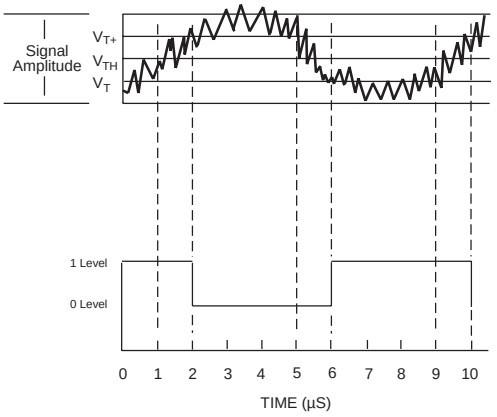
```
'OPTION BASE 1
' Pulse Width Modulation Example using Quick Basic
' Base Address of 5300 card configured to 100 Hex

    ADDBASE = &H100
    CTC1 = ADDBASE + 4
    CTC2 = ADDBASE + 8
    CTC3 = ADDBASE + 12

' Divide on board Master clock by 4.  8 MHz divided by 4 equals 2
MHz

' Set CTC3, Ch0 to mode 3

    OUT CTC3 + 3, &H36
    OUT CTC3 + 0, 8
    OUT CTC3 + 0, 0

' Get Frequency from user

    INPUT "Enter the frequency in 'Hz'"; FR
    INPUT “Enter the Duty Cycle in ‘%’”; DU

' Calculate the divisor for the 82C54 for correct frequency genera-
tion

    FD = 1000000! / FR

' CTC1 Ch 2, set to Mode 3

    OUT CTC1 + 3, &HB6

' Calculate low and high byte counts

    LSB = FD MOD 256
    MSB = INT(FD / 256)
    OUT CTC1 + 2, LSB
    OUT CTC1 + 2, MSB

' Calculate duty cycle in number of clock counts

    DX = 1 - DU / 100
    PD = FD * DX
    LSB = PD MOD 256
    MSB = INT(PD / 256)
```

```
' Set CTC1, CH1 to Mode 1

    OUT CTC1 + 3, &H72
    OUT CTC1 + 1, LSB
    OUT CTC1 + 1, MSB

END
```

This appendix includes some basic information on Schmitt triggers and pulse width modulation.

SCHMITT TRIGGER

A Schmitt trigger provides better noise immunity than a TTL input. The output will turn on when the signal reaches V_{T+} but will not turn off until the signal drops below V_{T-}. With TTL, the signal could be on or off between V_{T-} and V_{T+}.



Received signal from transmission line with thresholds at different amplitudes.

Recovered signal with CMOS Schmitt trigger, V_{T+} and V_T. Restores true waveform.

Recovered signal from Schmitt is same width as comparator with threshold at V_{TH}, and is only delayed in time.

Figure A–1—Schmitt Trigger

PULSE WIDTH MODULATION

Pulse width modulation (PWM) is often used in controlling motor speed, light levels, or damper opening. The width of the pulse—how long it is on—is modulated according to the level needed. Figure A–2 shows a general–case PWM.

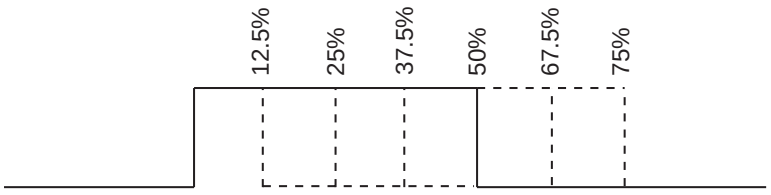


Figure A–2—Pulse Width Modulation

Within a given (programmed) time interval, the pulse signal may be on for less than 100% of the time. How long it is on—the pulse width —is determined by the amount of “ON” you want to achieve.

SPECIFICATIONS

Power Specifications
5V at 50 mA typical

Environmental Specifications
–40° to 85° C operating
RH 5% to 95%, noncondensing

Clock Input
8 MHz max.

Optically Isolated Input Rate
1 MHz max.

Optically Isolated Input Current
15 mA max., 5 mA min.

Digital I/O
8 lines, programmable as 4 or 8 inputs or outputs. TTL compatible logic.

Size
4.5 in. x 4.9 in.

For additional information, please turn to Appendix C, which includes the data sheet for the 82C54 Counter/Timer chip.

JUMPER CONFIGURATIONS

5300 Base Address Select: W1	
Pins Jumpered	**Base I/O Address**
[1-2][3-4][5-6]	100H*
[3-4][5-6]	110H
[1-2][5-6]	120H
[5-6]	130H
[1-2][3-4]	140H
[3-4]	150H
[1-2]	160H
No jumpers	170H

* = default

5300 Interrupt Request Lines: W2	
Pins Jumpered	**IRQ**
[1-2]	IRQ2
[11-12]	IRQ3
[9-10]	IRQ4
[7-8]	IRQ5
[5-6]	IRQ6
[3-4]	IRQ7
[1-3]	No interrupt selected*

* = default

5300 Interrupt Sources: W2	
Pins Jumpered	**Base I/O Address**
[13-14]	PIO 0 Output
[15-16]	CTC2 Output 0
[14-16]	No interrupt selected*

* = default

CONNECTOR CONFIGURATIONS

J1: Counter/Timer & Digital I/O Port					
Pin#	**Function**	**Direction**	**Pin#**	**Function**	**Direction**
1	CTC1 Clock 0	In	18	Gnd	
2	Gnd		19	CTC3 Output 0	Out
3	CTC1 Gate 0	In	20	Gnd	
4	Gnd		21	CTC3 Output 2	Out
5	CTC1 Output 0	Out	22	Gnd	
6	Gnd		23	PIO 0	In/Out
7	CTC1 Clock 1	In	24	PIO 1	In/Out
8	Gnd		25	PIO 2	In/Out
9	CTC1 Gate 1	In	26	PIO 3	In/Out
10	Gnd		27	PIO 4	In/Out
11	CTC1 Output 1	Out	28	PIO 5	In/Out
12	Gnd		29	PIO 6	In/Out
13	CTC1 Clock 2	In	30	PIO 7	In/Out
14	Gnd		31	CTC2 Clock 2	In
15	CTC1 Gate 2	In	32	CTC2 Clock 1	In
16	Gnd		33	CTC2 Clock 0	In
17	CTC1 Output 2	Out	34	+5V	

J2: Counter/Timer Port		
Pin#	**Function**	**Direction**
1	CTC3 Output 2 / 2*	Out
2	CTC3 Output 2 / 2	Out
3	CTC2 Output 0	In
4	CTC2 Gate 0	In
5	CTC2 Output 1	Out
6	CTC2 Gate 1	In
7	CTC2 Output 2	Out
8	CTC2 Gate 2	In
9	Gnd	
10	Gnd	

* = Inverted output of pin 2.

I/O MAPS

5300 I/O Map	
82C55A Digital I/O Chip	
Description	**Address**
Port A	Base + 0x00
Port B	Base + 0x01
Port C	Base + 0x02
Control	Base + 0x03
8254 CTC1	
Description	**Address**
Counter 0	Base + 0x04
Counter 1	Base + 0x05
Counter 2	Base + 0x06
Control CTC1	Base + 0x07
8254 CTC2	
Description	**Address**
Counter 0	Base + 0x08
Counter 1	Base + 0x09
Counter 2	Base + 0x0A
Control CTC2	Base + 0x0B
8254 CTC3	
Description	**Address**
Counter 0	Base + 0x0C
Counter 1	Base + 0x0D
Counter 2	Base + 0x0E
Control CTC3	Base + 0x0F

This page intentionally left blank.

CHMOS PROGRAMMABLE INTERVAL TIMER

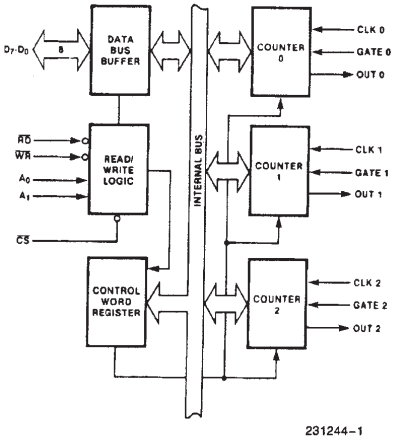
82C54
CHMOS PROGRAMMABLE INTERVAL TIMER

- **Compatible with all Intel and most other microprocessors**
- **High Speed, "Zero Wait State" Operation with 8 MHz 8086/88 and 80186/188**
- **Handles Inputs from DC to 8 MHz — 10 MHz for 82C54-2**
- **Available in EXPRESS — Standard Temperature Range — Extended Temperature Range**
- **Three independent 16-bit counters**
- **Low Power CHMOS — I_{CC} = 10 mA @ 8 MHz Count frequency**
- **Completely TTL Compatible**
- **Six Programmable Counter Modes**
- **Binary or BCD counting**
- **Status Read Back Command**
- **Available in 24-Pin DIP and 28-Pin PLCC**

The Intel 82C54 is a high-performance, CHMOS version of the industry standard 8254 counter/timer which is designed to solve the timing control problems common in microcomputer system design. It provides three independent 16-bit counters, each capable of handling clock inputs up to 10 MHz. All modes are software programmable. The 82C54 is pin compatible with the HMOS 8254, and is a superset of the 8253.

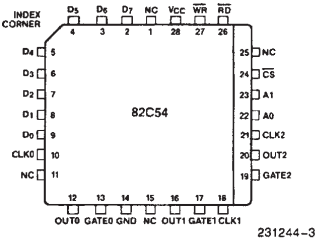
Six programmable timer modes allow the 82C54 to be used as an event counter, elapsed time indicator, programmable one-shot, and in many other applications.

The 82C54 is fabricated on Intel's advanced CHMOS III technology which provides low power consumption with performance equal to or greater than the equivalent HMOS product. The 82C54 is available in 24-pin DIP and 28-pin plastic leaded chip carrier (PLCC) packages.



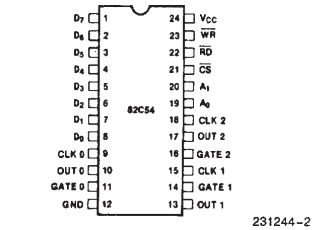
231244-1

Figure 1. 82C54 Block Diagram



231244-3

PLASTIC LEADED CHIP CARRIER



231244-2

Diagrams are for pin reference only.
Package sizes are not to scale.

Figure 2. 82C54 Pinout

Order Number: 231244-005

Table 1. Pin Description

Symbol	Pin Number		Type	Function
	DIP	PLCC		
D_7-D_0	1-8	2-9	I/O	Data: Bidirectional tri-state data bus lines, connected to system data bus.
CLK 0	9	10	I	Clock 0: Clock input of Counter 0.
OUT 0	10	12	O	Output 0: Output of Counter 0.
GATE 0	11	13	I	Gate 0: Gate input of Counter 0.
GND	12	14		Ground: Power supply connection.
OUT 1	13	16	O	Out 1: Output of Counter 1.
GATE 1	14	17	I	Gate 1: Gate input of Counter 1.
CLK 1	15	18	I	Clock 1: Clock input of Counter 1.
GATE 2	16	19	I	Gate 2: Gate input of Counter 2.
OUT 2	17	20	O	Out 2: Output of Counter 2.
CLK 2	18	21	I	Clock 2: Clock input of Counter 2.
A_1, A_0	20-19	23-22	I	Address: Used to select one of the three Counters or the Control Word Register for read or write operations. Normally connected to the system address bus.
				A_1 | A_0 | Selects 0 | 0 | Counter 0 0 | 1 | Counter 1 1 | 0 | Counter 2 1 | 1 | Control Word Register
$\overline{CS}$	21	24	I	Chip Select: A low on this input enables the 82C54 to respond to $\overline{RD}$ and $\overline{WR}$ signals. $\overline{RD}$ and $\overline{WR}$ are ignored otherwise.
$\overline{RD}$	22	26	I	Read Control: This input is low during CPU read operations.
$\overline{WR}$	23	27	I	Write Control: This input is low during CPU write operations.
V_{CC}	24	28		Power: +5V power supply connection.
NC		1, 11, 15, 25		No Connect

FUNCTIONAL DESCRIPTION

General

The 82C54 is a programmable interval timer/counter designed for use with Intel microcomputer systems. It is a general purpose, multi-timing element that can be treated as an array of I/O ports in the system software.

The 82C54 solves one of the most common problems in any microcomputer system, the generation of accurate time delays under software control. Instead of setting up timing loops in software, the programmer configures the 82C54 to match his requirements and programs one of the counters for the desired delay. After the desired delay, the 82C54 will interrupt the CPU. Software overhead is minimal and variable length delays can easily be accommodated.

Some of the other counter/timer functions common to microcomputers which can be implemented with the 82C54 are:

- Real time clock
- Even counter
- Digital one-shot
- Programmable rate generator
- Square wave generator
- Binary rate multiplier
- Complex waveform generator
- Complex motor controller

Block Diagram

DATA BUS BUFFER

This 3-state, bi-directional, 8-bit buffer is used to interface the 82C54 to the system bus (see Figure 3).

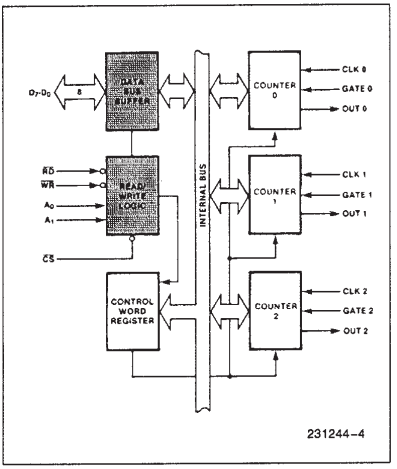


Figure 3. Block Diagram Showing Data Bus Buffer and Read/Write Logic Functions

READ/WRITE LOGIC

The Read/Write Logic accepts inputs from the system bus and generates control signals for the other functional blocks of the 82C54. A_1 and A_0 select one of the three counters or the Control Word Register to be read from/written into. A "low" on the $\overline{RD}$ input tells the 82C54 that the CPU is reading one of the counters. A "low" on the $\overline{WR}$ input tells the 82C54 that the CPU is writing either a Control Word or an initial count. Both $\overline{RD}$ and $\overline{WR}$ are qualified by $\overline{CS}$; $\overline{RD}$ and $\overline{WR}$ are ignored unless the 82C54 has been selected by holding $\overline{CS}$ low.

CONTROL WORD REGISTER

The Control Word Register (see Figure 4) is selected by the Read/Write Logic when A_1, A_0 = 11. If the CPU then does a write operation to the 82C54, the data is stored in the Control Word Register and is interpreted as a Control Word used to define the operation of the Counters.

The Control Word Register can only be written to; status information is available with the Read-Back Command.

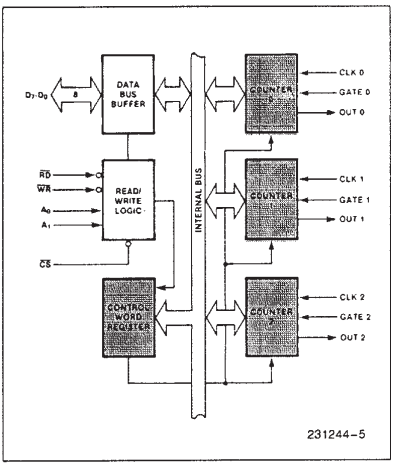


Figure 4. Block Diagram Showing Control Word Register and Counter Functions

COUNTER 0, COUNTER 1, COUNTER 2

These three functional blocks are identical in operation, so only a single Counter will be described. The internal block diagram of a single counter is shown in Figure 5.

The Counters are fully independent. Each Counter may operate in a different Mode.

The Control Word Register is shown in the figure; it is not part of the Counter itself, but its contents determine how the Counter operates.

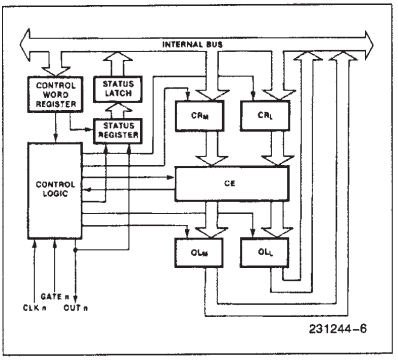


Figure 5. Internal Block Diagram of a Counter

The status register, shown in the Figure, when latched, contains the current contents of the Control Word Register and status of the output and null count flag. (See detailed explanation of the Read-Back command.)

The actual counter is labelled CE (for "Counting Element"). It is a 16-bit presettable synchronous down counter.

OL_M and OL_L are two 8-bit latches. OL stands for "Output Latch"; the subscripts M and L stand for "Most significant byte" and "Least significant byte" respectively. Both are normally referred to as one unit and called just OL. These latches normally "follow" the CE, but if a suitable Counter Latch Command is sent to the 82C54, the latches "latch" the present count until read by the CPU and then return to "following" the CE. One latch at a time is enabled by the counter's Control Logic to drive the internal bus. This is how the 16-bit Counter communicates over the 8-bit internal bus. Note that the CE itself cannot be read; whenever you read the count, it is the OL that is being read.

Similarly, there are two 8-bit registers called CR_M and CR_L (for "Count Register"). Both are normally referred to as one unit and called just CR. When a new count is written to the Counter, the count is stored in the CR and later transferred to the CE. The Control Logic allows one register at a time to be loaded from the internal bus. Both bytes are transferred to the CE simultaneously. CR_M and CR_L are cleared when the Counter is programmed. In this way, if the Counter has been programmed for one byte counts (either most significant byte only or least significant byte only) the other byte will be zero. Note that the CE cannot be written into; whenever a count is written, it is written into the CR.

The Control Logic is also shown in the diagram. CLK n, GATE n, and OUT n are all connected to the outside world through the Control Logic.

82C54 SYSTEM INTERFACE

The 82C54 is treated by the systems software as an array of peripheral I/O ports; three are counters and the fourth is a control register for MODE programming.

Basically, the select inputs A_0, A_1 connect to the A_0, A_1 address bus signals of the CPU. The $\overline{CS}$ can be derived directly from the address bus using a linear select method. Or it can be connected to the output of a decoder, such as an Intel 8205 for larger systems.

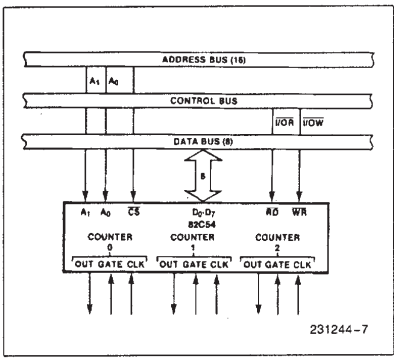


Figure 6. 82C54 System Interface

OPERATIONAL DESCRIPTION

General

After power-up, the state of the 82C54 is undefined. The Mode, count value, and output of all Counters are undefined.

How each Counter operates is determined when it is programmed. Each Counter must be programmed before it can be used. Unused counters need not be programmed.

Programming the 82C54

Counters are programmed by writing a Control Word and then an initial count. The control word format is shown in Figure 7.

All Control Words are written into the Control Word Register, which is selected when A_1, A_0 = 11. The Control Word itself specifies which Counter is being programmed.

By contrast, initial counts are written into the Counters, not the Control Word Register. The A_1, A_0 inputs are used to select the Counter to be written into. The format of the initial count is determined by the Control Word used.

Control Word Format

A_1, A_0 = 11 $\overline{CS}$ = 0 $\overline{RD}$ = 1 $\overline{WR}$ = 0

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC1	SC0	RW1	RW0	M2	M1	M0	BCD

SC — Select Counter:

SC1	SC0	
0	0	Select Counter 0
0	1	Select Counter 1
1	0	Select Counter 2
1	1	Read-Back Command (See Read Operations)

RW — Read/Write:

RW1	RW0	
0	0	Counter Latch Command (see Read Operations)
0	1	Read/Write least significant byte only.
1	0	Read/Write most significant byte only.
1	1	Read/Write least significant byte first, then most significant byte.

NOTE: Don't care bits (X) should be 0 to insure compatibility with future Intel products.

M — MODE:

M2	M1	M0	
0	0	0	Mode 0
0	0	1	Mode 1
X	1	0	Mode 2
X	1	1	Mode 3
1	0	0	Mode 4
1	0	1	Mode 5

BCD:

0	Binary Counter 16-bits
1	Binary Coded Decimal (BCD) Counter (4 Decades)

Figure 7. Control Word Format

Write Operations

The programming procedure for the 82C54 is very flexible. Only two conventions need to be remembered:

1) For each Counter, the Control Word must be written before the initial count is written.
2) The initial count must follow the count format specified in the Control Word (least significant byte only, most significant byte only, or least significant byte and then most significant byte).

Since the Control Word Register and the three Counters have separate addresses (selected by the A_1, A_0 inputs), and each Control Word specifies the Counter it applies to (SC0, SC1 bits), no special instruction sequence is required. Any programming sequence that follows the conventions above is acceptable.

A new initial count may be written to a Counter at any time without affecting the Counter's programmed Mode in any way. Counting will be affected as described in the Mode definitions. The new count must follow the programmed count format.

If a Counter is programmed to read/write two-byte counts, the following precaution applies: A program must not transfer control between writing the first and second byte to another routine which also writes into that same Counter. Otherwise, the Counter will be loaded with an incorrect count.

		A_1	A_0
Control Word —	Counter 0	1	1
LSB of count —	Counter 0	0	0
MSB of count —	Counter 0	0	0
Control Word —	Counter 1	1	1
LSB of count —	Counter 1	0	1
MSB of count —	Counter 1	0	1
Control Word —	Counter 2	1	1
LSB of count —	Counter 2	1	0
MSB of count —	Counter 2	1	0

		A_1	A_0
Control Word —	Counter 2	1	1
Control Word —	Counter 1	1	1
Control Word —	Counter 0	1	1
LSB of count —	Counter 2	1	0
MSB of count —	Counter 2	1	0
LSB of count —	Counter 1	0	1
MSB of count —	Counter 1	0	1
LSB of count —	Counter 0	0	0
MSB of count —	Counter 0	0	0

		A_1	A_0
Control Word —	Counter 0	1	1
Counter Word —	Counter 1	1	1
Control Word —	Counter 2	1	1
LSB of count —	Counter 2	1	0
LSB of count —	Counter 1	0	1
LSB of count —	Counter 0	0	0
MSB of count —	Counter 0	0	0
MSB of count —	Counter 1	0	1
MSB of count —	Counter 2	1	0

		A_1	A_0
Control Word —	Counter 1	1	1
Control Word —	Counter 0	1	1
LSB of count —	Counter 1	0	1
Control Word —	Counter 2	1	1
LSB of count —	Counter 0	0	0
MSB of count —	Counter 1	0	1
LSB of count —	Counter 2	1	0
MSB of count —	Counter 0	0	0
MSB of count —	Counter 2	1	0

NOTE:
In all four examples, all counters are programmed to read/write two-byte counts.
These are only four of many possible programming sequences.

Figure 8. A Few Possible Programming Sequences

Read Operations

It is often desirable to read the value of a Counter without disturbing the count in progress. This is easily done in the 82C54.

There are three possible methods for reading the counters: a simple read operation, the Counter Latch Command, and the Read-Back Command. Each is explained below. The first method is to perform a simple read operation. To read the Counter, which is selected with the A1, A0 inputs, the CLK input of the selected Counter must be inhibited by using either the GATE input or external logic. Otherwise, the count may be in the process of changing when it is read, giving an undefined result.

COUNTER LATCH COMMAND

The second method uses the "Counter Latch Command". Like a Control Word, this command is written to the Control Word Register, which is selected when A_1, A_0 = 11. Also like a Control Word, the SC0, SC1 bits select one of the three Counters, but two other bits, D5 and D4, distinguish this command from a Control Word.

$A_1, A_0 = 11; \overline{CS} = 0; \overline{RD} = 1; \overline{WR} = 0$

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC1	SC0	0	0	X	X	X	X

SC1, SC0 - specify counter to be latched

SC1	SC0	Counter
0	0	0
0	1	1
1	0	2
1	1	Read-Back Command

D5,D4 - 00 designates Counter Latch Command

X - don't care

NOTE:
Don't care bits (X) should be 0 to insure compatibility with future Intel products.

Figure 9. Counter Latching Command Format

The selected Counter's output latch (OL) latches the count at the time the Counter Latch Command is received. This count is held in the latch until it is read by the CPU (or until the Counter is reprogrammed). The count is then unlatched automatically and the OL returns to "following" the counting element (CE). This allows reading the contents of the Counters "on the fly" without affecting counting in progress. Multiple Counter Latch Commands may be used to latch more than one Counter. Each latched Counter's OL holds its count until it is read. Counter Latch Commands do not affect the programmed Mode of the Counter in any way.

If a Counter is latched and then, some time later, latched again before the count is read, the second Counter Latch Command is ignored. The count read will be the count at the time the first Counter Latch Command was issued.

With either method, the count must be read according to the programmed format; specifically, if the Counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other; read or write or programming operations of other Counters may be inserted between them.

Another feature of the 82C54 is that reads and writes of the same Counter may be interleaved; for example, if the Counter is programmed for two byte counts, the following sequence is valid.

1. Read least significant byte.
2. Write new least significant byte.
3. Read most significant byte.
4. Write new most significant byte.

If a Counter is programmed to read/write two-byte counts, the following precaution applies; A program must not transfer control between reading the first and second byte to another routine which also reads from that same Counter. Otherwise, an incorrect count will be read.

READ-BACK COMMAND

The third method uses the Read-Back command. This command allows the user to check the count value, programmed Mode, and current state of the OUT pin and Null Count flag of the selected counter(s).

The command is written into the Control Word Register and has the format shown in Figure 10. The command applies to the counters selected by setting their corresponding bits D3,D2,D1 = 1.

A0, A1 = 11 $\overline{CS}$ = 0 $\overline{RD}$ = 1 $\overline{WR}$ = 0

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	1	$\overline{COUNT}$	$\overline{STATUS}$	CNT 2	CNT 1	CNT 0	0

D_5: 0 = Latch count of selected counter(s)
D_4: 0 = Latch status of selected counter(s)
D_3: 1 = Select counter 2
D_2: 1 = Select counter 1
D_1: 1 = Select counter 0
D_0: Reserved for future expansion; must be 0

Figure 10. Read-Back Command Format

The read-back command may be used to latch multiple counter output latches (OL) by setting the $\overline{COUNT}$ bit D5 = 0 and selecting the desired counter(s). This single command is functionally equivalent to several counter latch commands, one for each counter latched. Each counter's latched count is held until it is read (or the counter is reprogrammed). That counter is automatically unlatched when read, but other counters remain latched until they are read. If multiple count read-back commands are issued to the same counter without reading the

count, all but the first are ignored; i.e., the count which will be read is the count at the time the first read-back command was issued.

The read-back command may also be used to latch status information of selected counter(s) by setting $\overline{\text{STATUS}}$ bit D4 = 0. Status must be latched to be read; status of a counter is accessed by a read from that counter.

The counter status format is shown in Figure 11. Bits D5 through D0 contain the counter's programmed Mode exactly as written in the last Mode Control Word. OUTPUT bit D7 contains the current state of the OUT pin. This allows the user to monitor the counter's output via software, possibly eliminating some hardware from a system.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
OUTPUT	NULL COUNT	RW1	RW0	M2	M1	M0	BCD

D_7 1 = Out Pin is 1
0 = Out Pin is 0
D_6 1 = Null count
0 = Count available for reading
D_5-D_0 Counter Programmed Mode (See Figure 7)

Figure 11. Status Byte

NULL COUNT bit D6 indicates when the last count written to the counter register (CR) has been loaded into the counting element (CE). The exact time this happens depends on the Mode of the counter and is described in the Mode Definitions, but until the count is loaded into the counting element (CE), it can't be read from the counter. If the count is latched or read before this time, the count value will not reflect the new count just written. The operation of Null Count is shown in Figure 12.

THIS ACTION:	CAUSES:
A. Write to the control word register:[1]	Null count = 1
B. Write to the count register (CR);[2]	Null count = 1
C. New count is loaded into CE (CR → CE);	Null count = 0

[1] Only the counter specified by the control word will have its null count set to 1. Null count bits of other counters are unaffected.
[2] If the counter is programmed for two-byte counts (least significant byte then most significant byte) null count goes to 1 when the second byte is written.

Figure 12. Null Count Operation

If multiple status latch operations of the counter(s) are performed without reading the status, all but the first are ignored; i.e., the status that will be read is the status of the counter at the time the first status read-back command was issued.

Both count and status of the selected counter(s) may be latched simultaneously by setting both $\overline{\text{COUNT}}$ and $\overline{\text{STATUS}}$ bits D5,D4 = 0. This is functionally the same as issuing two separate read-back commands at once, and the above discussions apply here also. Specifically, if multiple count and/or status read-back commands are issued to the same counter(s) without any intervening reads, all but the first are ignored. This is illustrated in Figure 13.

If both count and status of a counter are latched, the first read operation of that counter will return latched status, regardless of which was latched first. The next one or two reads (depending on whether the counter is programmed for one or two type counts) return latched count. Subsequent reads return unlatched count.

Command								Description	Results
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0		
1	1	0	0	0	0	1	0	Read back count and status of Counter 0	Count and status latched for Counter 0
1	1	1	0	0	1	0	0	Read back status of Counter 1	Status latched for Counter 1
1	1	1	0	1	1	0	0	Read back status of Counters 2, 1	Status latched for Counter 2, but not Counter 1
1	1	0	1	1	0	0	0	Read back count of Counter 2	Count latched for Counter 2
1	1	0	0	0	1	0	0	Read back count and status of Counter 1	Count latched for Counter 1, but not status
1	1	1	0	0	0	1	0	Read back status of Counter 1	Command ignored, status already latched for Counter 1

Figure 13. Read-Back Command Example

$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A_1	A_0	
0	1	0	0	0	Write into Counter 0
0	1	0	0	1	Write into Counter 1
0	1	0	1	0	Write into Counter 2
0	1	0	1	1	Write Control Word
0	0	1	0	0	Read from Counter 0
0	0	1	0	1	Read from Counter 1
0	0	1	1	0	Read from Counter 2
0	0	1	1	1	No-Operation (3-State)
1	X	X	X	X	No-Operation (3-State)
0	1	1	X	X	No-Operation (3-State)

Figure 14. Read/Write Operations Summary

Mode Definitions

The following are defined for use in describing the operation of the 82C54.

CLK PULSE: a rising edge, then a falling edge, in that order, of a Counter's CLK input.

TRIGGER: a rising edge of a Counter's GATE input.

COUNTER LOADING: the transfer of a count from the CR to the CE (refer to the "Functional Description")

MODE 0: INTERRUPT ON TERMINAL COUNT

Mode 0 is typically used for event counting. After the Control Word is written, OUT is initially low, and will remain low until the Counter reaches zero. OUT then goes high and remains high until a new count or a new Mode 0 Control Word is written into the Counter.

GATE = 1 enables counting; GATE = 0 disables counting. GATE has no effect on OUT.

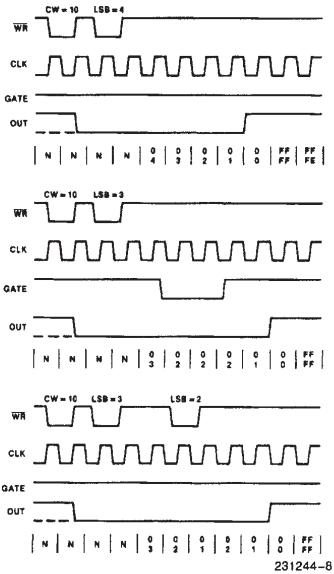
After the Control Word and initial count are written to a Counter, the initial count will be loaded on the next CLK pulse. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not go high until N + 1 CLK pulses after the initial count is written.

If a new count is written to the Counter, it will be loaded on the next CLK pulse and counting will continue from the new count. If a two-byte count is written, the following happens:

1) Writing the first byte disables counting. OUT is set low immediately (no clock pulse required).
2) Writing the second byte allows the new count to be loaded on the next CLK pulse.

This allows the counting sequence to be synchronized by software. Again, OUT does not go high until N + 1 CLK pulses after the new count of N is written.

If an initial count is written while GATE = 0, it will still be loaded on the next CLK pulse. When GATE goes high, OUT will go high N CLK pulses later; no CLK pulse is needed to load the Counter as this has already been done.



NOTE:
The Following Conventions Apply To All Mode Timing Diagrams:
1. Counters are programmed for binary (not BCD) counting and for Reading/Writing least significant byte (LSB) only.
2. The counter is always selected ($\overline{CS}$ always low).
3. CW stands for "Control Word"; CW = 10 means a control word of 10, hex is written to the counter.
4. LSB stands for "Least Significant Byte" of count.
5. Numbers below diagrams are count values.
The lower number is the least significant byte.
The upper number is the most significant byte. Since the counter is programmed to Read/Write LSB only, the most significant byte cannot be read.
N stands for an undefined count.
Vertical lines show transitions between count values.

Figure 15. Mode 0

MODE 1: HARDWARE RETRIGGERABLE ONE-SHOT

OUT will be initially high. OUT will go low on the CLK pulse following a trigger to begin the one-shot pulse, and will remain low until the Counter reaches zero. OUT will then go high and remain high until the CLK pulse after the next trigger.

After writing the Control Word and initial count, the Counter is armed. A trigger results in loading the Counter and setting OUT low on the next CLK pulse, thus starting the one-shot pulse. An initial count of N will result in a one-shot pulse N CLK cycles in duration. The one-shot is retriggerable, hence OUT will remain low for N CLK pulses after any trigger. The one-shot pulse can be repeated without rewriting the same count into the counter. GATE has no effect on OUT.

If a new count is written to the Counter during a one-shot pulse, the current one-shot is not affected unless the Counter is retriggered. In that case, the Counter is loaded with the new count and the one-shot pulse continues until the new count expires.

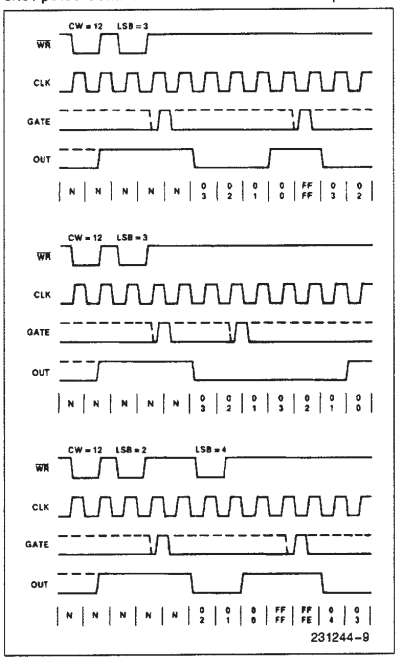


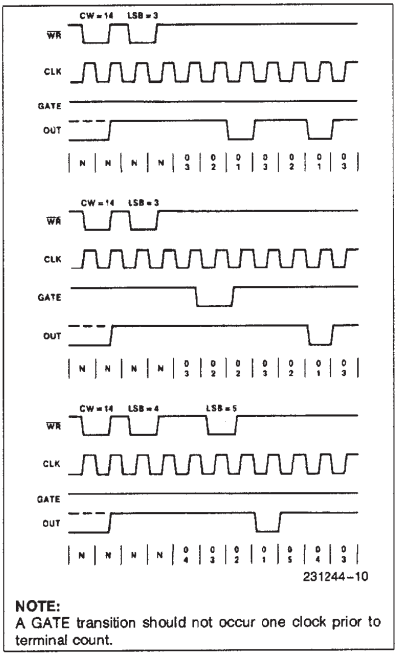
Figure 16. Mode 1

MODE 2: RATE GENERATOR

This Mode functions like a divide-by-N counter. It is typically used to generate a Real Time Clock interrupt. OUT will initially be high. When the initial count has decremented to 1, OUT goes low for one CLK pulse. OUT then goes high again, the Counter reloads the initial count and the process is repeated. Mode 2 is periodic; the same sequence is repeated indefinitely. For an initial count of N, the sequence repeats every N CLK cycles.

GATE = 1 enables counting; GATE = 0 disables counting. If GATE goes low during an output pulse, OUT is set high immediately. A trigger reloads the Counter with the initial count on the next CLK pulse; OUT goes low N CLK pulses after the trigger. Thus the GATE input can be used to synchronize the Counter.

After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. OUT goes low N CLK Pulses after the initial count is written. This allows the Counter to be synchronized by software also.



NOTE:
A GATE transition should not occur one clock prior to terminal count.

Figure 17. Mode 2

Writing a new count while counting does not affect the current counting sequence. If a trigger is received after writing a new count but before the end of the current period, the Counter will be loaded with the new count on the next CLK pulse and counting will continue from the new count. Otherwise, the new count will be loaded at the end of the current counting cycle. In mode 2, a COUNT of 1 is illegal.

MODE 3: SQUARE WAVE MODE

Mode 3 is typically used for Baud rate generation. Mode 3 is similar to Mode 2 except for the duty cycle of OUT. OUT will initially be high. When half the initial count has expired, OUT goes low for the remainder of the count. Mode 3 is periodic; the sequence above is repeated indefinitely. An initial count of N results in a square wave with a period of N CLK cycles.

GATE = 1 enables counting; GATE = 0 disables counting. If GATE goes low while OUT is low, OUT is set high immediately; no CLK pulse is required. A trigger reloads the Counter with the initial count on the next CLK pulse. Thus the GATE input can be used to synchronize the Counter.

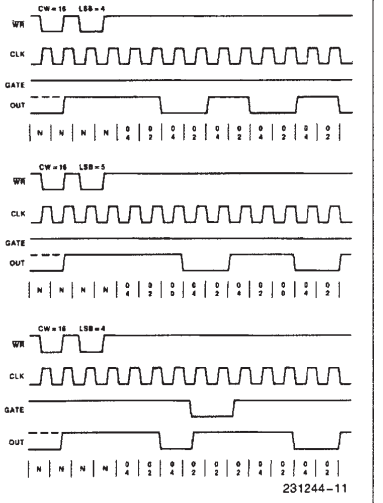
After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. This allows the Counter to be synchronized by software also.

Writing a new count while counting does not affect the current counting sequence. If a trigger is received after writing a new count but before the end of the current half-cycle of the square wave, the Counter will be loaded with the new count on the next CLK pulse and counting will continue from the new count. Otherwise, the new count will be loaded at the end of the current half-cycle.

Mode 3 is implemented as follows:

Even counts: OUT is initially high. The initial count is loaded on one CLK pulse and then is decremented by two on succeeding CLK pulses. When the count expires OUT changes value and the Counter is reloaded with the initial count. The above process is repeated indefinitely.

Odd counts: OUT is initially high. The initial count minus one (an even number) is loaded on one CLK pulse and then is decremented by two on succeeding CLK pulses. One CLK pulse *after* the count expires, OUT goes low and the Counter is reloaded with the initial count minus one. Succeeding CLK pulses decrement the count by two. When the count expires, OUT goes high again and the Counter is reloaded with the initial count minus one. The above process is repeated indefinitely. So for odd counts, OUT will be high for (N + 1)/2 counts and low for (N − 1)/2 counts.



NOTE:
A GATE transition should not occur one clock prior to terminal count.

Figure 18. Mode 3

MODE 4: SOFTWARE TRIGGERED STROBE

OUT will be initially high. When the initial count expires, OUT will go low for one CLK pulse and then go high again. The counting sequence is "triggered" by writing the initial count.

GATE = 1 enables counting; GATE = 0 disables counting. GATE has no effect on OUT.

After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not strobe low until N + 1 CLK pulses after the initial count is written.

If a new count is written during counting, it will be loaded on the next CLK pulse and counting will continue from the new count. If a two-byte count is written, the following happens:

1) Writing the first byte has no effect on counting.
2) Writing the second byte allows the new count to be loaded on the next CLK pulse.

This allows the sequence to be "retriggered" by software. OUT strobes low N+1 CLK pulses after the new count of N is written.

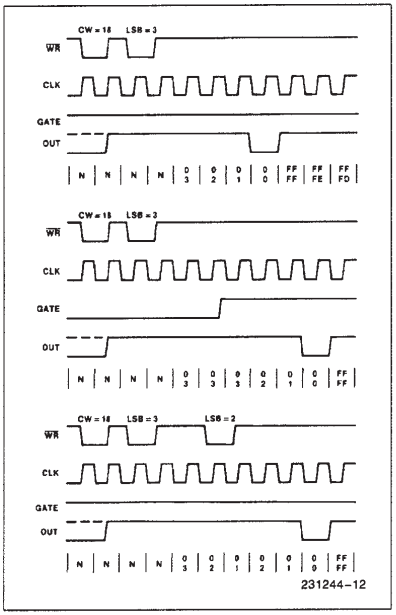


Figure 19. Mode 4

MODE 5: HARDWARE TRIGGERED STROBE (RETRIGGERABLE)

OUT will initially be high. Counting is triggered by a rising edge of GATE. When the initial count has expired, OUT will go low for one CLK pulse and then go high again.

After writing the Control Word and initial count, the counter will not be loaded until the CLK pulse after a trigger. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not strobe low until N+1 CLK pulses after a trigger.

A trigger results in the Counter being loaded with the initial count on the next CLK pulse. The counting sequence is retriggerable. OUT will not strobe low for N + 1 CLK pulses after any trigger. GATE has no effect on OUT.

If a new count is written during counting, the current counting sequence will not be affected. If a trigger occurs after the new count is written but before the current count expires, the Counter will be loaded with the new count on the next CLK pulse and counting will continue from there.

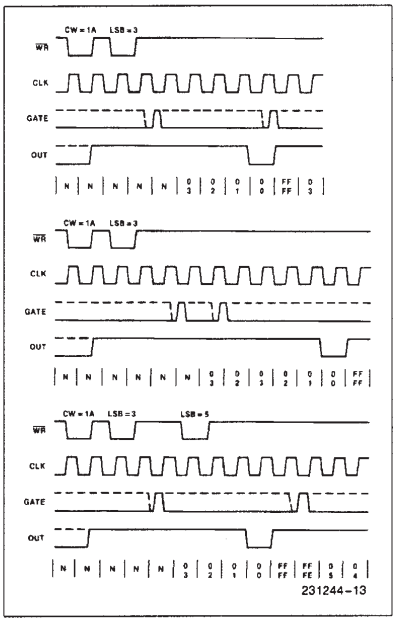


Figure 20. Mode 5

Signal Status Modes	Low Or Going Low	Rising	High
0	Disables counting	—	Enables counting
1	—	1) Initiates counting 2) Resets output after next clock	—
2	1) Disables counting 2) Sets output immediately high	Initiates counting	Enables counting
3	1) Disables counting 2) Sets output immediately high	Initiates counting	Enables counting
4	Disables counting	—	Enables counting
5	—	Initiates counting	—

Figure 21. Gate Pin Operations Summary

MODE	MIN COUNT	MAX COUNT
0	1	0
1	1	0
2	2	0
3	2	0
4	1	0

NOTE:
0 is equivalent to 2^{16} for binary counting and 10^4 for BCD counting

Figure 22. Minimum and Maximum Initial Counts

Operation Common to All Modes

Programming

When a Control Word is written to a Counter, all Control Logic is immediately reset and OUT goes to a known initial state; no CLK pulses are required for this.

GATE

The GATE input is always sampled on the rising edge of CLK. In Modes 0, 2, 3, and 4 the GATE input is level sensitive, and the logic level is sampled on the rising edge of CLK. In Modes 1, 2, 3, and 5 the GATE input is rising-edge sensitive. In these Modes, a rising edge of GATE (trigger) sets an edge-sensitive flip-flop in the Counter. This flip-flop is then sampled on the next rising edge of CLK; the flip-flop is reset immediately after it is sampled. In this way, a trigger will be detected no matter when it occurs—a high logic level does not have to be maintained until the next rising edge of CLK. Note that in Modes 2 and 3, the GATE input is both edge- and level-sensitive. In Modes 2 and 3, if a CLK source other than the system clock is used, GATE should be pulsed immediately following $\overline{WR}$ of a new count value.

COUNTER

New counts are loaded and Counters are decremented on the falling edge of CLK.

The largest possible initial count is 0; this is equivalent to 2^{16} for binary counting and 10^4 for BCD counting.

The Counter does not stop when it reaches zero. In Modes 0, 1, 4, and 5 the Counter "wraps around" to the highest count, either FFFF hex for binary counting or 9999 for BCD counting, and continues counting. Modes 2 and 3 are periodic; the Counter reloads itself with the initial count and continues counting from there.

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias.....0°C to 70°C
Storage Temperature .....−65° to +150°C
Supply Voltage .....−0.5 to +8.0V
Operating Voltage .....+4V to +7V
Voltage on any Input.....GND −2V to +6.5V
Voltage on any Output ..GND−0.5V to V_{CC} + 0.5V
Power Dissipation .....1 Watt

**Notice: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*

D.C. CHARACTERISTICS

(T_A = 0°C to 70°C, V_{CC} = 5V ± 10%, GND = 0V) (T_A = −40°C to +85°C for Extended Temperature)

Symbol	Parameter	Min	Max	Units	Test Conditions
V_{IL}	Input Low Voltage	−0.5	0.8	V	
V_{IH}	Input High Voltage	2.0	V_{CC} + 0.5	V	
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = 2.5 mA
V_{OH}	Output High Voltage	3.0 V_{CC} − 0.4		V V	I_{OH} = −2.5 mA I_{OH} = −100 µA
I_{IL}	Input Load Current		±2.0	µA	V_{IN} = V_{CC} to 0V
I_{OFL}	Output Float Leakage Current		±10	µA	V_{OUT} = V_{CC} to 0.0V
I_{CC}	V_{CC} Supply Current		20	mA	Clk Freq = 8MHz 82C54 10MHz 82C54-2
I_{CCSB}	V_{CC} Supply Current-Standby		10	µA	CLK Freq = DC $\overline{CS}$ = V_{CC}. All Inputs/Data Bus V_{CC} All Outputs Floating
I_{CCSB1}	V_{CC} Supply Current-Standby		150	µA	CLK Freq = DC $\overline{CS}$ = V_{CC}. All Other Inputs, I/O Pins = V_{GND}, Outputs Open
C_{IN}	Input Capacitance		10	pF	f_c = 1 MHz
$C_{I/O}$	I/O Capacitance		20	pF	Unmeasured pins returned to GND(5)
C_{OUT}	Output Capacitance		20	pF	

A.C. CHARACTERISTICS

(T_A = 0°C to 70°C, V_{CC} = 5V ± 10%, GND = 0V) (T_A = −40°C to +85°C for Extended Temperature)

BUS PARAMETERS (Note 1)

READ CYCLE

Symbol	Parameter	82C54		82C54-2		Units
		Min	Max	Min	Max	
t_{AR}	Address Stable Before $\overline{RD}$ ↓	45		30		ns
t_{SR}	$\overline{CS}$ Stable Before $\overline{RD}$ ↓	0		0		ns
t_{RA}	Address Hold Time After $\overline{RD}$ ↑	0		0		ns
t_{RR}	$\overline{RD}$ Pulse Width	150		95		ns
t_{RD}	Data Delay from $\overline{RD}$ ↓		120		85	ns
t_{AD}	Data Delay from Address		220		185	ns
t_{DF}	$\overline{RD}$ ↑ to Data Floating	5	90	5	65	ns
t_{RV}	Command Recovery Time	200		165		ns

NOTE:
1. AC timings measured at V_{OH} = 2.0V, V_{OL} = 0.8V.

A.C. CHARACTERISTICS (Continued)

WRITE CYCLE

Symbol	Parameter	82C54		82C54-2		Units
		Min	Max	Min	Max	
t_{AW}	Address Stable Before $\overline{WR}\downarrow$	0		0		ns
t_{SW}	$\overline{CS}$ Stable Before $\overline{WR}\downarrow$	0		0		ns
t_{WA}	Address Hold Time After $\overline{WR}\uparrow$	0		0		ns
t_{WW}	$\overline{WR}$ Pulse Width	150		95		ns
t_{DW}	Data Setup Time Before $\overline{WR}\uparrow$	120		95		ns
t_{WD}	Data Hold Time After $\overline{WR}\uparrow$	0		0		ns
t_{RV}	Command Recovery Time	200		165		ns

CLOCK AND GATE

Symbol	Parameter	82C54		82C54-2		Units
		Min	Max	Min	Max	
t_{CLK}	Clock Period	125	DC	100	DC	ns
t_{PWH}	High Pulse Width	60(3)		30(3)		ns
t_{PWL}	Low Pulse Width	60(3)		50(3)		ns
T_R	Clock Rise Time		25		25	ns
t_F	Clock Fall Time		25		25	ns
t_{GW}	Gate Width High	50		50		ns
t_{GL}	Gate Width Low	50		50		ns
t_{GS}	Gate Setup Time to CLK $\uparrow$	50		40		ns
t_{GH}	Gate Hold Time After CLK $\uparrow$	50(2)		50(2)		ns
T_{OD}	Output Delay from CLK $\downarrow$		150		100	ns
t_{ODG}	Output Delay from Gate $\downarrow$		120		100	ns
t_{WC}	CLK Delay for Loading(4)	0	55	0	55	ns
t_{WG}	Gate Delay for Sampling(4)	−5	50	−5	40	ns
t_{WO}	OUT Delay from Mode Write		260		240	ns
t_{CL}	CLK Set Up for Count Latch	−40	45	−40	40	ns

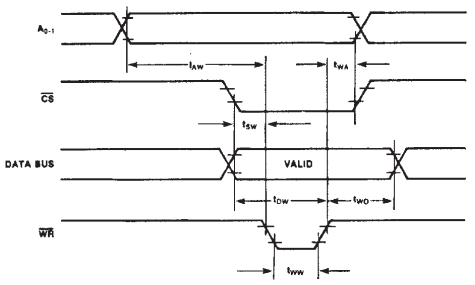
NOTES:
2. In Modes 1 and 5 triggers are sampled on each rising clock edge. A second trigger within 120 ns (70 ns for the 82C54-2) of the rising clock edge may not be detected.
3. Low-going glitches that violate t_{PWH}, t_{PWL} may cause errors requiring counter reprogramming.
4. Except for Extended Temp., See Extended Temp. A.C. Characteristics below.
5. Sampled not 100% tested. T_A = 25°C.
6. If CLK present at T_{WC} min then Count equals N+2 CLK pulses, T_{WC} max equals Count N+1 CLK pulse. T_{WC} min to T_{WC} max, count will be either N+1 or N+2 CLK pulses.
7. In Modes 1 and 5, if GATE is present when writing a new Count value, at T_{WG} min Counter will not be triggered, at T_{WG} max Counter will be triggered.
8. If CLK present when writing a Counter Latch or ReadBack Command, at T_{CL} min CLK will be reflected in count value latched, at T_{CL} max CLK will not be reflected in the count value latched. Writing a Counter Latch or ReadBack Command between T_{CL} min and T_{WL} max will result in a latched count vallue which is ± one least significant bit.

EXTENDED TEMPERATURE (T_A = −40°C to +85°C for Extended Temperature)

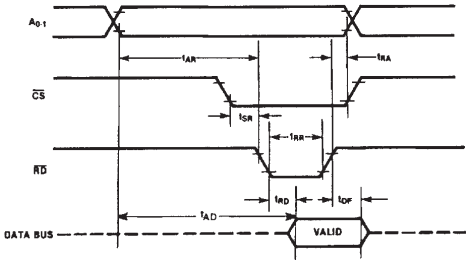
Symbol	Parameter	82C54		82C54-2		Units
		Min	Max	Min	Max	
t_{WC}	CLK Delay for Loading	−25	25	−25	25	ns
t_{WG}	Gate Delay for Sampling	−25	25	−25	25	ns

WAVEFORMS

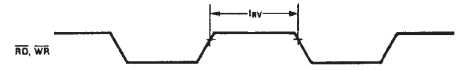
WRITE



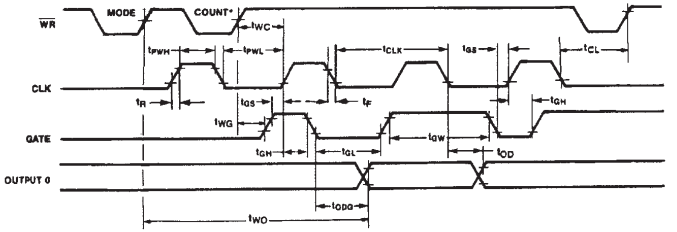
READ



RECOVERY



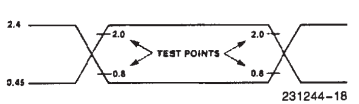
CLOCK AND GATE



231244-17

* Last byte of count being written

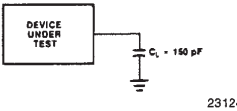
A.C. TESTING INPUT, OUTPUT WAVEFORM



231244-18

A.C. Testing: Inputs are driven at 2.4V for a logic "1" and 0.45V for a logic "0." Timing measurements are made at 2.0V for a logic "1" and 0.8V for a logic "0."

A.C. TESTING LOAD CIRCUIT



231244-19

C_L = 150 pF
C_L includes jig capacitance

WARRANTY

Octagon Systems Corporation (Octagon), warrants that its standard hardware products will be free from defects in materials and workmanship under normal use and service for the current established warranty period. Octagon's obligation under this warranty shall not arise until Buyer returns the defective product, freight prepaid to Octagon's facility or another specified location. Octagon's only responsibility under this warranty is, at its option, to replace or repair, free of charge, any defective component part of such products.

LIMITATIONS ON WARRANTY

The warranty set forth above does not extend to and shall not apply to:

1. Products, including software, which have been repaired or altered by other than Octagon personnel, unless Buyer has properly altered or repaired the products in accordance with procedures previously approved in writing by Octagon.
2. Products which have been subject to power supply reversal, misuse, neglect, accident, or improper installation.
3. The design, capability, capacity, or suitability for use of the Software. Software is licensed on an "AS IS" basis without warranty.

The warranty and remedies set forth above are in lieu of all other warranties expressed or implied, oral or written, either in fact or by operation of law, statutory or otherwise, including warranties of merchantability and fitness for a particular purpose, which Octagon specifically disclaims. Octagon neither assumes nor authorizes any other liability in connection with the sale, installation or use of its products. Octagon shall have no liability for incidental or consequential damages of any kind arising out of the sale, delay in delivery, installation, or use of its products.

SERVICE POLICY

1. Octagon's goal is to ship your product within 5 working days of receipt.
2. If a product should fail during the warranty period, it will be repaired free of charge. For out of warranty repairs, the customer will be invoiced for repair charges at current standard labor and materials rates.
3. Customers that return products for repairs, within the warranty period, and the product is found to be free of defect, may be liable for the minimum current repair charge.

RETURNING A PRODUCT FOR REPAIR

Upon determining that repair services are required, the customer must:

1. Obtain an RMA (Return Material Authorization) number from the Customer Service Department, 303-430–1500.
2. If the request is for an out of warranty repair, a purchase order number or other acceptable information must be supplied by the customer.
3. Include a list of problems encountered along with your name, address, telephone, and RMA number.
4. Carefully package the product in an antistatic bag. (Failure to package in antistatic material will VOID all warranties.) Then package in a safe container for shipping.
5. Write RMA number on the outside of the box.
6. For products under warranty, the customer pays for shipping to Octagon. Octagon pays for shipping back to customer.
7. Other conditions and limitations may apply to international shipments.

NOTE: PRODUCTS RETURNED TO OCTAGON FREIGHT COLLECT OR WITHOUT AN RMA NUMBER CANNOT BE ACCEPTED AND WILL BE RETURNED FREIGHT COLLECT.

RETURNS

There will be a 15% restocking charge on returned product that is unopened and unused, if Octagon accepts such a return. Returns will not be accepted 30 days after purchase. Opened and/or used products, non-standard products, software and printed materials are not returnable without prior written agreement.

GOVERNING LAW

This agreement is made in, governed by and shall be construed in accordance with the laws of the State of Colorado.

The information in this manual is provided for reference only. Octagon does not assume any liability arising out of the application or use of the information or products described in this manual. This manual may contain or reference information and products protected by copyrights or patents. No license is conveyed under the rights of Octagon or others.