



# UM8253 / UM8253-5

## Programmable Interval Timer

### Features

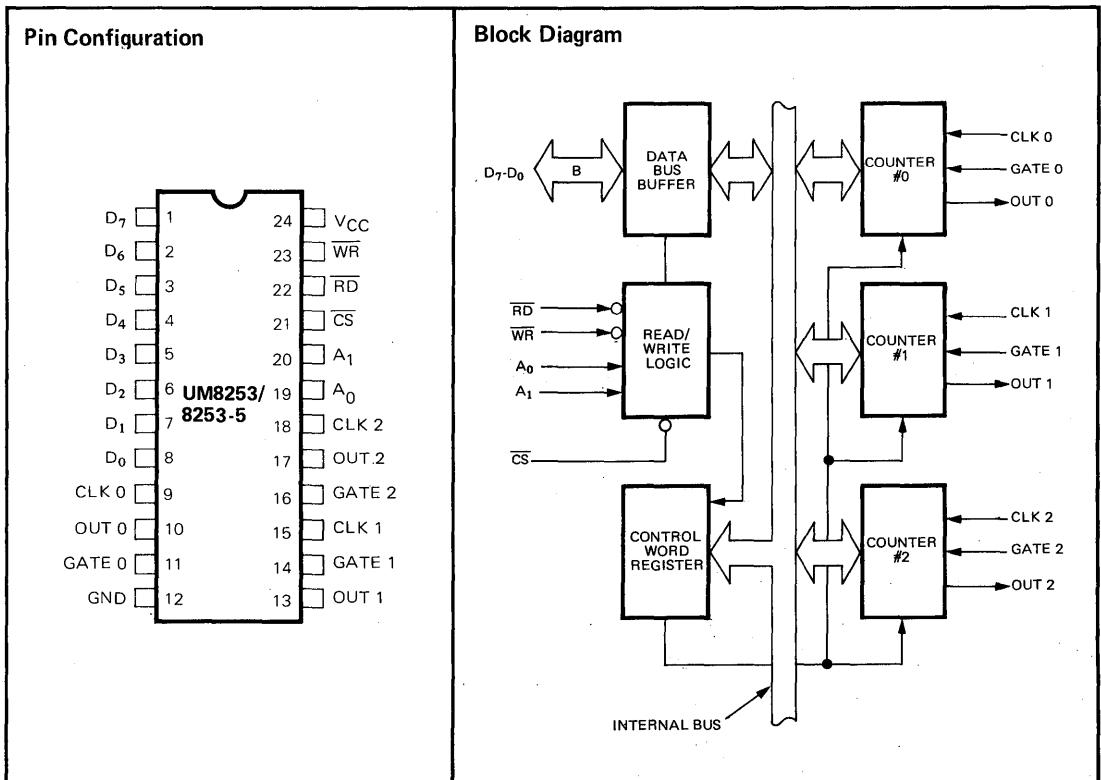
- MCS-85™ compatible UM8253-5
- 3 independent 16-bit counters
- DC to 2.6 MHz
- Programmable counter modes
- Count binary or BCD
- Single +5V supply

### General Description

The UM8253 is a programmable counter timer device designed for use as an microcomputer peripheral. It uses NMOS technology with a single +5V supply and is packaged in a 24-pin plastic DIP.

It is organized as 3 independent 16-bit counters, each with a count rate of up to 2.6 MHz. All modes of operation are software programmable.

\*MCS-85™ is the trademark of Intel microsystem.



Peripheral IC

**Absolute Maximum Ratings\***

Ambient Temperature Under Bias  $T_A$  . . . . .  $0^{\circ}\text{C}$  to  $70^{\circ}\text{C}$   
 Storage Temperature  $T_{\text{STG}}$  . . . . .  $-65^{\circ}\text{C}$  to  $+150^{\circ}\text{C}$   
 Voltage on Any Pin with Respect to  
     Ground . . . . .  $-0.5\text{V}$  to  $+7\text{V}$   
 Power Dissipation . . . . . 1 Watt

**\*Comments**

Stress above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only. Functional operation of this device at these or any other conditions above those indicated in the operational sections of this specification is not implied and exposure to absolute maximum rating conditions for extended periods may affect device reliability.

**D.C. Characteristics**

( $T_A = 0^{\circ}\text{C}$  to  $70^{\circ}\text{C}$ ,  $V_{\text{CC}} = 5\text{V} \pm 10\%$ )

Symbol	Parameter	Min	Max.	Units	Conditions
$V_{\text{IL}}$	Input Low Voltage	-0.5	0.8	V	
$V_{\text{IH}}$	Input High Voltage	2.2	$V_{\text{CC}}+0.5\text{V}$	V	
$V_{\text{OL}}$	Output Low Voltage		0.45	V	Note 1
$V_{\text{OH}}$	Output High Voltage	2.4		V	Note 2
$I_{\text{IL}}$	Input Load Current		$\pm 10$	$\mu\text{A}$	$V_{\text{IN}} = V_{\text{CC}}$ to $0\text{V}$
$I_{\text{OFL}}$	Output Float Leakage		$\pm 10$	$\mu\text{A}$	$V_{\text{OUT}} = V_{\text{CC}}$ to $0.45\text{V}$
$I_{\text{CC}}$	$V_{\text{CC}}$ Supply Current		140	mA	

**Capacitance**

( $T_A = 25^{\circ}\text{C}$ ,  $V_{\text{CC}} = \text{GND} = 0\text{V}$ )

Symbol	Parameter	Min.	Typ.	Max.	Units	Conditions
$C_{\text{IN}}$	Input Capacitance			10	pF	$f_c = 1\text{ MHz}$
$C_{\text{I/O}}$	I/O Capacitance			20	pF	Unmeasured pins returned to $V_{\text{SS}}$

**A.C. Characteristics**

( $T_A = 0^{\circ}\text{C}$  to  $70^{\circ}\text{C}$ ,  $V_{\text{CC}} = 5.0\text{V} \pm 10\%$ ,  $\text{GND} = 0\text{V}$ )

**Bus Parameters (Note 3)**
**READ CYCLE**

Symbol	Parameter	UM8253		UM8253-5		Units
		Min.	Max.	Min.	Max.	
$t_{\text{AR}}$	Address Stable Before $\overline{\text{READ}}$	50		30		ns
$t_{\text{RA}}$	Address Hold Time for $\overline{\text{READ}}$	5		5		ns
$t_{\text{RR}}$	$\overline{\text{READ}}$ Pulse Width	400		300		ns
$t_{\text{RD}}$	Data Delay From $\overline{\text{READ}}$ (4)		300		200	ns
$t_{\text{DF}}$	$\overline{\text{READ}}$ to Data Floating	25	125	25	100	ns
$t_{\text{RV}}$	Recovery Time Between $\overline{\text{READ}}$ and Any Other Control Signal	1		1		$\mu\text{s}$

**A.C. Characteristics (Continued)**
**WRITE CYCLE**

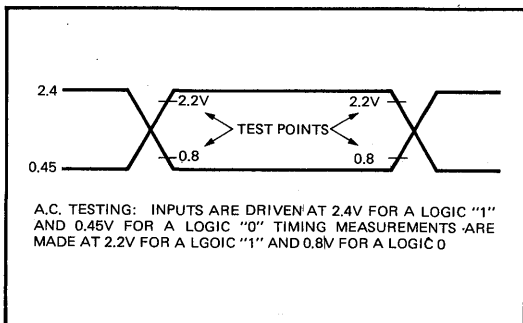
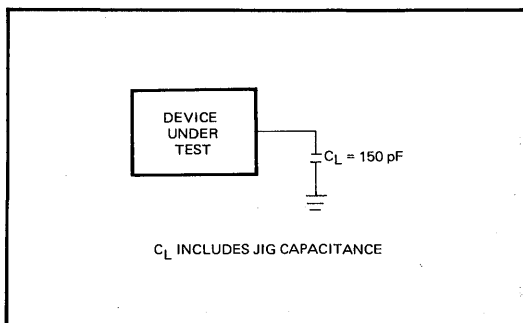
Symbol	Parameter	UM8253		UM8253-5		Units
		Min.	Max.	Min.	Max.	
$t_{AW}$	Address Stable Before $\overline{WRITE}$	50		30		ns
$t_{WA}$	Address Hold Time for $\overline{WRITE}$	30		30		ns
$t_{WW}$	$\overline{WRITE}$ Pulse Width	400		300		ns
$t_{DW}$	Data Set Up Time for $\overline{WRITE}$	300		250		ns
$t_{WD}$	Data Hold Time for $\overline{WRITE}$	40		30		ns
$t_{RV}$	Recovery time Between $\overline{WRITE}$ and Any Other Control Signal	1		1		$\mu$ s

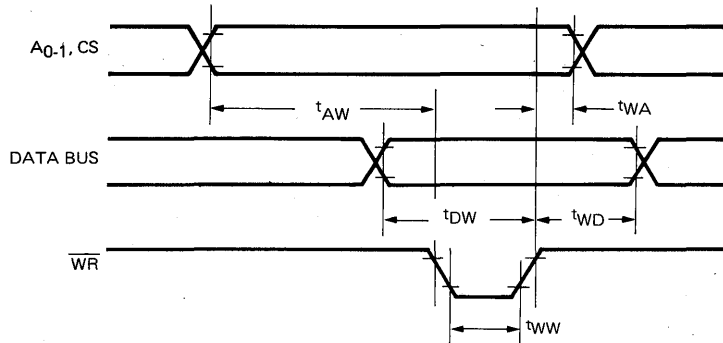
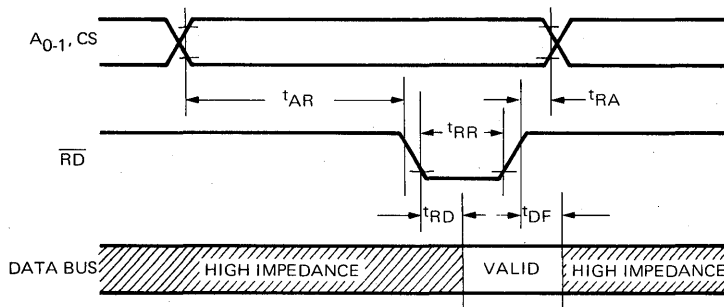
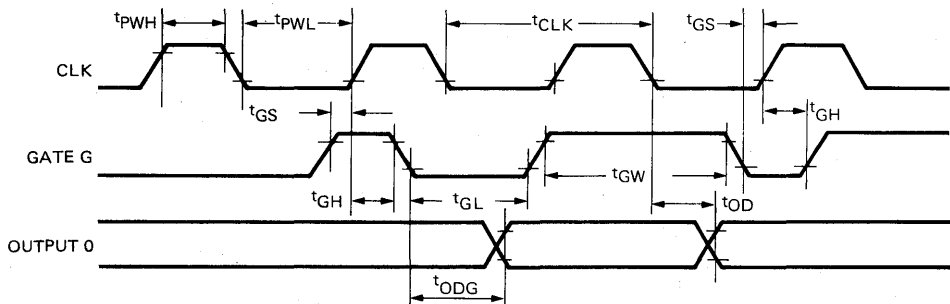
**Clock and Gate Timing**

Symbol	Parameter	UM8253		UM8253-5		Unit
		Min.	Max.	Min.	Max.	
$t_{CLK}$	Clock Period	380	dc	380	dc	ns
$t_{PWH}$	High Pulse Width	230		230		ns
$t_{PWL}$	Low Pulse Width	150		150		ns
$t_{GW}$	Gate Width High	150		150		ns
$t_{GL}$	Gate Width Low	100		100		ns
$t_{GS}$	Gate Set Up Time to $CLK\uparrow$	100		100		ns
$t_{GH}$	Gate Hold Time After $CLK\uparrow$	50		50		ns
$t_{OD}$	Output Delay From $CLK\downarrow$ [4]		400		400	ns
$t_{ODG}$	Output Delay From $Gate\downarrow$ [4]		300		300	ns

**Notes:**

- $I_{OL} = 2.2$  mA.
- $I_{OH} = -400$   $\mu$ A.
- AC timings measured at  $V_{OH} 2.2$ ,  $V_{OL} = 0.8$ .
- $C_L = 150$  pF.

**A.C. Testing Input, Output Waveform**

**A.C. Testing Load Circuit**


**Waveforms**
**WRITE TIMING**

**READ TIMING**

**CLOCK AND GATE TIMING**


## Functional Description

### General

The UM8253 is a programmable interval timer/counter specifically designed for use with microcomputer systems. Its function is that of a general purpose, multiming element that can be treated as an array of I/O ports in the system software.

The UM8253 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 systems software, the programmer configures the UM8253 to match his requirements, initializes one of the counters of the UM8253 with the desired quantity, then upon command the UM8253 will count out the delay and interrupt the UPU when it has completed its tasks. It is easy to see that the software overhead is minimal and that multiple delays can easily be maintained by assignment of priority levels.

Other counter/timer functions that are non-delay in nature but also common to most microcomputers can be implemented with the UM8253.

- Programmable Rate Generator
- Event Counter
- Binary Rate Multiplier
- Real Time Clock
- Digital One-Shot
- Complex Motor Controller

### Data Bus Buffer

This 3-state, bi-directional, 8-bit buffer is used to interface the UM8253 to the system data bus. Data is transmitted or received by the buffer upon execution of input or OUT put CPU instructions. The Data Bus Buffer has three basic functions.

1. Programming the MODES of the UM8253.
2. Loading the count registers.
3. Reading the count values.

### Read/Write Logic

The Read/Write Logic accepts inputs from the system bus and in turn generates control signals for overall device operation. It is enable or disabled by CS so that no operation can occur to change the function unless the device has been selected by the system logic.

### $\overline{RD}$ (Read)

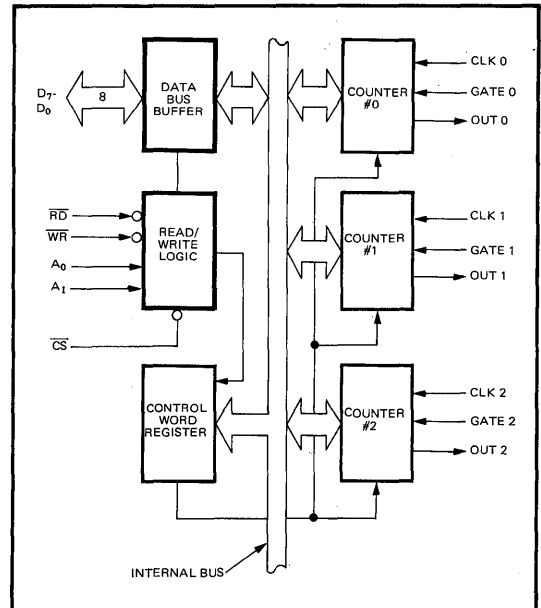
A "low" on this input informs the UM8253 that the CPU is inputting data in the form of a counters value.

### $\overline{WR}$ (Write)

A "low" on this input informs the UM8253 that the CPU is outputting data in the form of mode information or loading counters.

### $A_0, A_1$

These inputs are normally connected to the address bus. Their function is to select one of the three counters to be operated on and to address the control word register for mode selection.



**Figure 1. Block Diagram Showing Data Bus Buffer and Read/Write Logic Functions**

CS	RD	WR	$A_1$	$A_0$	
0	1	0	0	0	Load Counter No. 0
0	1	0	0	1	Load Counter No. 1
0	1	0	1	0	Load Counter No. 2
0	1	0	1	1	Write Mode Word
0	0	1	0	0	Read Counter No. 0
0	0	1	0	1	Read Counter No. 1
0	0	1	1	0	Read Counter No. 2
0	0	1	1	1	No-Operation 3-State
1	X	X	X	X	Disable 3-State
0	1	1	X	X	No-Operation 3-State

### $\overline{CS}$ (Chip Select)

A "low" on this input enables the UM8253. No reading or writing will occur unless the device is selected. The  $\overline{CS}$  input has no effect upon the actual operation of the counters.

### Control Word Register

The Control Word Register is selected when  $A_0, A_1$  are 11. It then accepts information from the data bus buffer and stores it in a register. The information stored in this register controls the operational MODE of each counter, selection of binary or BCD counting and the loading of each count register.

The Control Word Register can only be written into; no read operation of its contents is available.

### Counter #0, Counter #1, Counter #2

These three functional blocks are identical in operation so only a single Counter will be described. Each Counter consists of a single, 16-bit, pre-settable, DOWN counter. The counter can operate in either binary or BCD and its input, gate and output are configured by the selection of MODES stored in the Control Word Register.

The counters are fully independent and each can have separate Mode configuration and counting operation, binary or BCD. Also, there are special features in the control word that handle the loading of the count value so that software overhead can be minimized for these functions.

The reading of the contents of each counter is available to the programmer with simple READ operations for event counting applications and special commands and logic are included in the UM8253 so that the contents of each counter can be read "on the fly" without having to inhibit the clock input.

### UM8253 System Interface

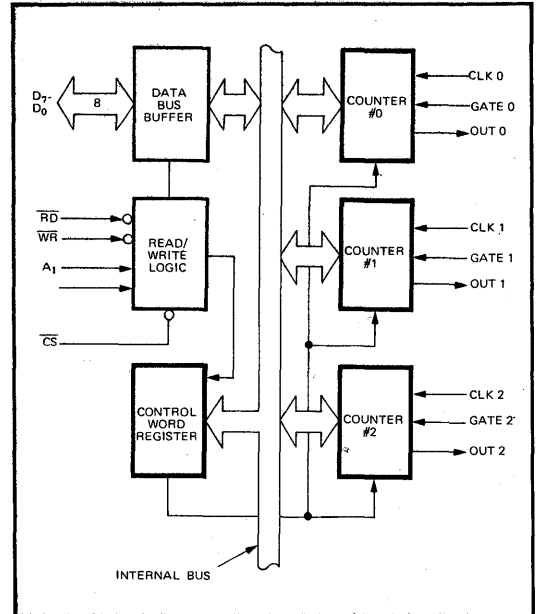
The UM8253 is a component of the UMC Microcomputer Systems and interfaces in the same manner as all other peripherals of the family. It is treated by the systems software as an array of peripheral I/O ports; three are counters and the fourth is a control register for MODE programming.

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 8205 for larger systems.

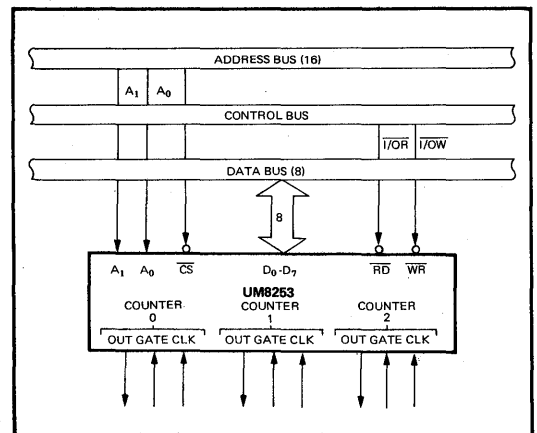
### Operational Description

#### General

The complete functional definition of the UM8253 is programmed by the systems software. A set of control words must be sent out by the CPU to initialize each counter of the UM8253 with the desired MODE and



**Figure 2. Block Diagram Showing Control Word Register and Counter Functions**



**Figure 3. UM8253 System Interface**

quantity information, prior to initialization, the MODE, count, and output of all counters is undefined. These control words program the MODE, Loading sequence and selection of binary or BCD counting.

Once programmed, the UM8253 is ready to perform whatever timing tasks it is assigned to accomplish.

The actual counting operation of each counter is completely independent and additional logic is provided on-chip so that the usual problems associated with efficient monitoring and management of external, asynchronous events or rates to the microcomputer system have been eliminated.

### Programming the UM8253

All of the MODES for each counter are programmed by the systems software by simple I/O operations.

Each counter of the UM8253 is individually programmed by writing a control word into the Control Word Register. ( $A_1, A_1 = 11$ )

### Control Word Format

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
SC1	SC0	RL1	RL0	M2	M1	M0	BCD

### Definition of Control

#### SC – Select Counter:

SC1	SC0	
0	0	Select Counter 0
0	1	Select Counter 1
1	0	Select Counter 2
1	1	Illegal

#### RL – Read/Load:

RL1	RL0	
0	0	Counter Latching operation (see READ/WRITE procedure Section)
1	0	Read/Load most significant byte only.
0	1	Read/Load least significant byte only.
1	1	Read/Load least significant byte first, then most significant byte.

#### 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)

### Counter Loading

The count register is not loaded until the count value is written (one or two bytes, depending on the mode selected by the RL bits), followed by a rising edge and a falling edge of the clock. Any read of the counter prior to that falling clock edge may yield invalid data.

### MODE Definition

**MODE 0: Interrupt on Terminal Count.** The output will be initially low after the mode set operation. After the count is loaded into the selected count register, the output will remain low and the counter will count. When terminal count is reached the output will go high and remain high until the selected count register is reloaded with the mode or a new count is loaded. The counter continues to decrement after terminal count has been reached.

Rewriting a counter register during counting results in the following:

- (1) Write 1st byte stops the current counting.
- (2) Write 2nd byte starts the new count.

**MODE 1: Programmable One-Shot.** The output will go low on the count following the rising edge of the gate input.

The output will go high on the terminal count. If a new count value is loaded while the output is low it will not affect the duration of the one-shot pulse until the succeeding trigger. The current count can be read at any time without affecting the one-shot pulse.

The one-shot is retriggerable, hence the output will remain low for the full count after any rising edge of the gate input.

**MODE 2: Rate Generator.** Divide by N counter. The output will be low for one period of the input clock. The period from one output pulse to the next equals the number of input counts in the count register. If the count register is reloaded between output pulses the present period will not be affected, but the subsequent period will reflect the new value.

The gate input, when low, will force the output high. When the gate input goes high, the counter will start from the initial count. Thus, the gate input can be used to synchronize the counter.

When this mode is set, the output will remain high until after the count register is loaded. The output then can also be synchronized by software.

MODE 3: Square Wave Rate Generator. Similar to MODE 2 except that the output will remain high until one half the count has been completed (for even numbers) and go low for the other half of the count. This is accomplished by decrementing the counter by two on the falling edge of each clock pulse. When the counter reaches terminal count, the state of the output is changed and the counter is reloaded with the full count and the whole process is repeated.

If the count is odd and the output is high, the first clock pulse (after the count is loaded) decrements the count by 1. Subsequent clock pulses decrement the clock by 2. After timeout, the output goes low and the full count is reloaded. The first clock pulse (following the reload) decrements the counter by 3. Subsequent clock pulses decrement the count by 2 until timeout. Then the whole process is repeated. In this way, if the count is odd, the output will be high for  $(N+1)/2$  counts and low for  $(N-1)/2$  counts.

In Modes 2 and 3, if a CLK source other than the system clock is used, GATE should be pulsed immediately following WR of a new count value.

MODE 4: Software Triggered Strobe. After the mode is set, the output will be high. When the count is loaded, the counter will begin counting. On terminal count, the output will go low for one input clock period, then will go high again.

If the count register is reloaded during counting, the new count will be loaded on the next CLK pulse. The count will be inhibited while the GATE input is low.

MODE 5: Hardware Triggered Strobe. The counter will start counting after the rising edge of the trigger input and will go low for one clock period when the terminal count is reached. The counter is retriggerable. The output will not go low until the full count after the rising edge of any trigger.

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	1) Reloads counter 2) Initiates counting	Enables counting
3	1) Disables counting 2) Sets output immediately high	1) Reloads counter 2) Initiates counting	Enables counting
4	Disables counting	---	Enables counting
5	---	Initiates counting	---

Figure 4. Gate Pin Operations Summary

## UM8253 Read/Write Procedure

### Write Operations

The systems software must program each counter of the UM8253 with the mode and quantity desired. The programmer must write out to the UM8253 a MODE control word and the programmed number of count register bytes (1 or 2) prior to actually using the selected counter.

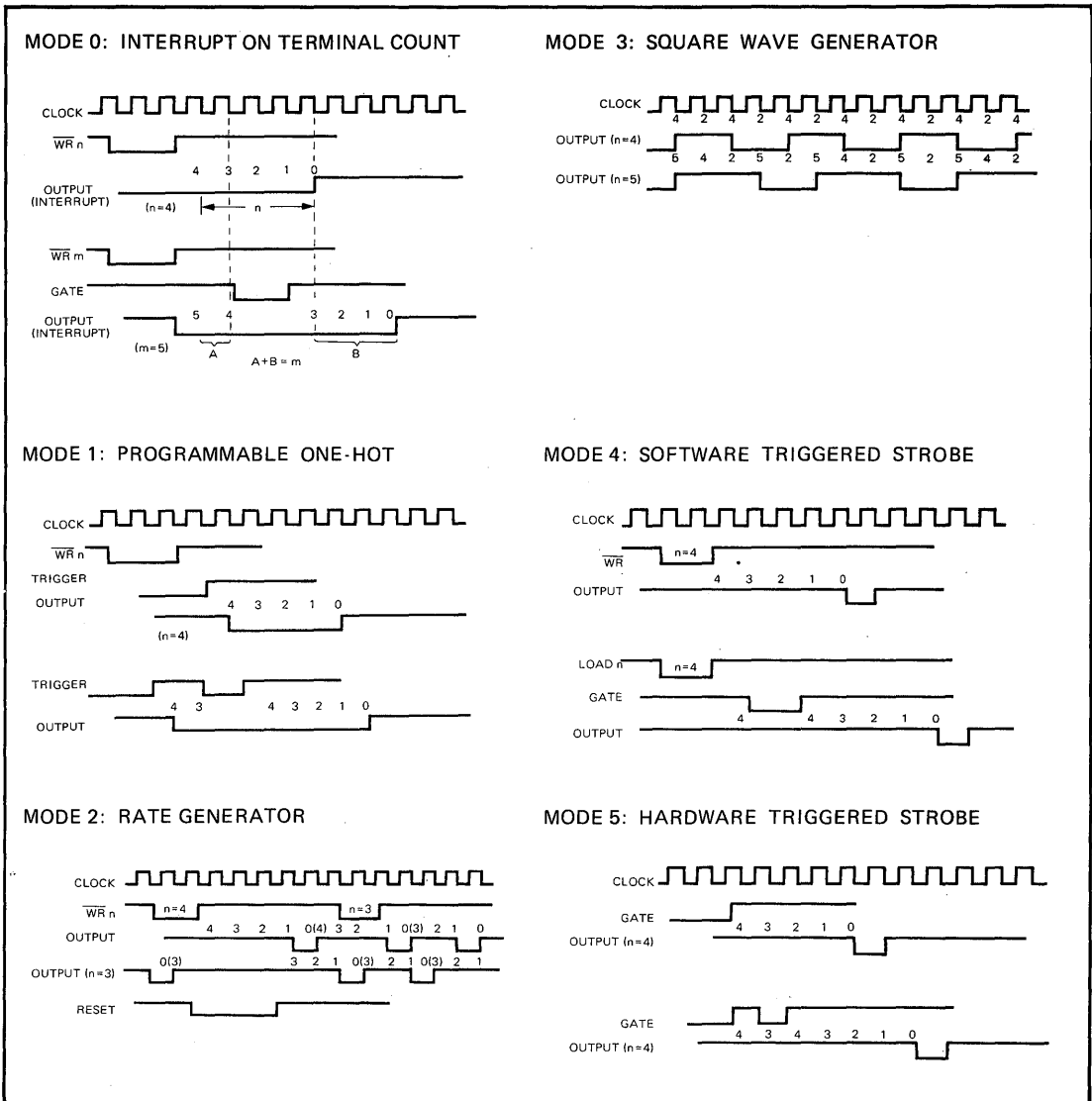
The actual order of the programming is quite flexible. Writing out of the MODE control word can be in any sequence of counter selection, e.g., counter #0 does not have to be first or counter #2 last. Each counter's MODE control word register has a separate address so that its loading is completely sequence independent. (SCO, SC1)

The loading of the Count Register with the actual count value, however, must be done in exactly the sequence

programmed in the MODE control word (RL0, RL1). This loading of the counter's count register is still sequence independent like the MODE control word loading, but when a selected count register is to be loaded it *must* be loaded with the number of bytes programmed in the MODE control word (RL0, RL1). The one or two bytes to be loaded in the count register do not have to follow the associated MODE control word. They can be programmed at any time following the MODE control word loading as long as the correct number of bytes is loaded in order.

All counters are down counters. Thus, the value loaded into the count register will actually be decremented. Loading all zeroes into a count register will result in the maximum count ( $2^{16}$  for Binary or  $10^4$  for BCD). In





**Figure 5. UM8253 Timing Diagrams**

MODE 0 the new count will not restart until the load has been completed. It will accept one of two bytes depending on how the MODE control words (RL0, RL1) are programmed. Then proceed with the restart operation.

**Read Operations**

In most counter applications it becomes necessary to read the value of the count in progress and make a computational decision based on this quantity. Event counters

are probably the most common application that uses this function. The UM8253 contains logic that will allow the programmer to easily read the contents of any of the three counters without disturbing the actual count in progress.

There are two methods that the programmer can use to read the value of the counters. The first method involves the use of simple I/O read operations of the selected counter. By controlling the A0, A1 inputs to the UM8253 the programmer can select the counter to be read

(remember that no read operation of the mode register is allowed A0, A1=11). The only requirement with this method is that in order to assure a stable count reading the actual operation of the selected counter *must be inhibited* either by controlling the Gate input or by external logic that inhibits the clock input. The contents of the counter selected will be available as follows:

first I/O Read contains the least significant byte (LSB).

second I/O Read contains the most significant byte (MSB).

Due to the internal logic of the UM8253 it is absolutely necessary to complete the entire reading procedure. If two bytes are programmed to be read then two bytes must be read before any loading WR command can be sent to the same counter.

### Read Operation Chart

A1	A0	RD	
0	0	0	Read Counter No. 0
0	1	0	Read Counter No. 1
1	0	0	Read Counter No. 2
1	1	0	Illegal

### Reading While Counting

In order for the programmer to read the contents of any counter without effecting or disturbing the counting operation the UM8253 has special internal logic that can be accessed using simple WR commands to the MODE register. Basically, when the programmer wishes to read the contents of a selected counter "on the fly" he loads

the MODE register with a special code which latches the present count value into a storage register so that its contents contain an accurate, stable quantity. The programmer then issues a normal read command to the selected counter and the contents of the latched register is available.

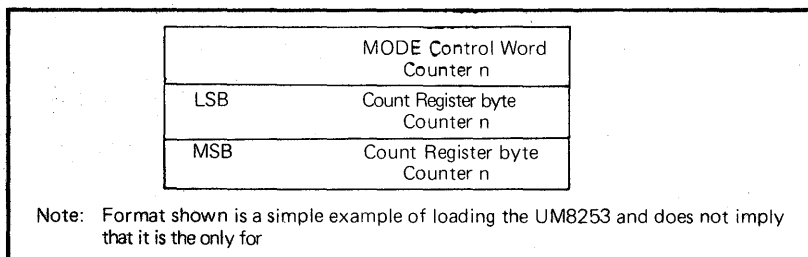


Figure 6. Programming Format

### MODE Register for Latching Count

A0, A1 = 11

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
SC1	SC0	0	0	X	X	X	X

SC1, SC0 – specify counter to be latched.

D5, D4 – 00 designates counter latching operation.

X – don't care.

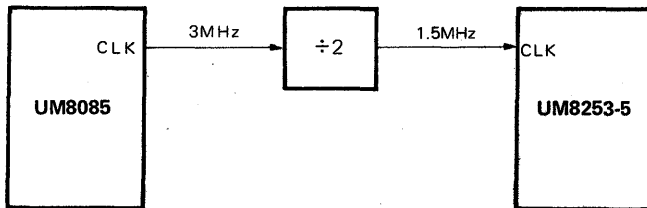
The same limitation applies to this mode of reading the counter as the previous method. That is, it is mandatory to complete the entire read operation as programmed.

This command has no effect on the counter's mode.

		A <sub>1</sub>	A <sub>0</sub>
No. 1	MODE Control Word Counter 0	1	1
No. 2	MODE Control Word Counter 1	1	1
No. 3	MODE Control Word Counter 2	1	1
No. 4	LSB Count Register Byte Counter 1	0	1
No. 5	MSB Count Register Byte Counter 1	0	1
No. 6	LSB Count Register Byte Counter 2	1	0
No. 7	MSB Count Register Byte Counter 2	1	0
No. 8	LSB Count Register Byte Counter 0	0	0
No. 9	MSB Count Register Byte Counter 0	0	0

Note: The exclusive addresses of each counter's count register make the task of programming the UM8253 a very simple matter, and maximum effective use of the device will result if this feature is fully utilized.

**Figure 7. Alternate Programming Formats**



If an UM8085 clock output is to drive an UM8253-5 clock input, it must be reduced to 2MHz or less.

**Figure 8. MCS-85™ Clock Interface\***

**Ordering Information**

Part Number	CLK
UM8253	2.6MHz
UM8253-5	5MHz