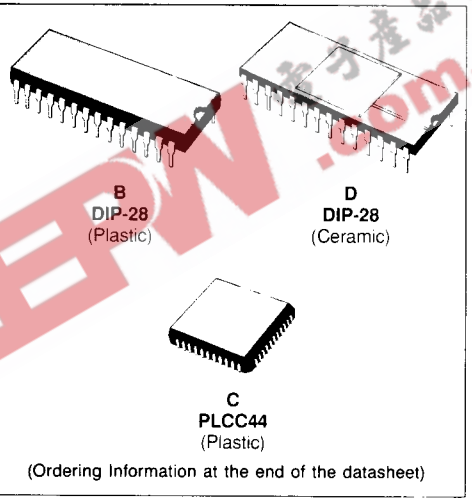
SGS-THOMSON MICROELECTRONICS

Z84C30

Z80C CTC CMOS VERSION

- FOUR INDEPENDENTLY PROGRAMMABLE COUNTER/TIMER CHANNELS, EACH WITH A READABLE DOWNCOUNTER AND A SELECTABLE 16 OR 256 PRESCALER. DOWNCOUNTERS ARE RELOADED AUTOMATICALLY AT ZERO COUNT
- THREE CHANNELS HAVE ZERO COUNT/TIMEOUT OUTPUTS CAPABLE OF DRIVING DARLINGTON TRANSISTORS
- SELECTABLE POSITIVE OR NEGATIVE TRIGGER INITIATES TIMER OPERATION
- STANDARD Z80C FAMILY DAISY-CHAIN INTERRUPT STRUCTURE PROVIDES FULLY VECTORED, PRIORITIZED INTERRUPTS WITHOUT EXTERNAL LOGIC. THE CTC MAY ALSO BE USED AS AN INTERRUPT CONTROLLER
- INTERFACE DIRECTLY TO THE Z80C CPU OR-FOR BAUD RATE GENERATION - TO THE Z80C SIO
- SINGLE 5 V ± 10 % POWER SUPPLY
- LOW POWER CONSUMPTION :
  - 3 mA TYP. AT 4 MHz
  - 4 mA TYP. AT 6 MHz
  - LESS THAN 10 µA IN POWER DOWN MODE
- EXTENDED OPERATING TEMPERATURE :
  - 40 °C TO + 85 °C



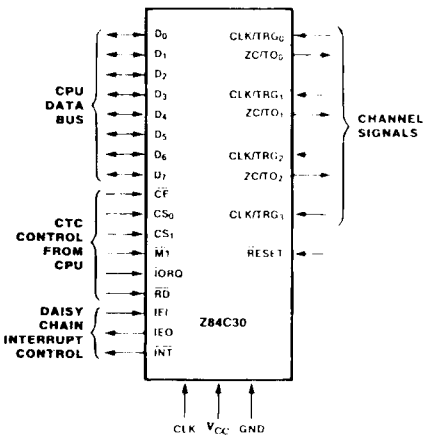
B
DIP-28
(Plastic)

D
DIP-28
(Ceramic)

C
PLCC44
(Plastic)

(Ordering Information at the end of the datasheet)

LOGIC FUNCTIONS



DESCRIPTION

The Z80C CTC four-channel counter/timer can be programmed by system software for a broad range of counting and timing applications. The four independently programmable channels of the CTC satisfy common microcomputer system requirements for event counting, interrupt and interval timing, and general clock rate generation.

System design is simplified because the CTC connects directly to both the CPU and the SIO with no additional logic. In larger systems, address decoders and buffers may be required.

Programming the CTC is straightforward : each channel is programmed with two bytes ; a third is necessary when interrupts are enabled. Once started, the CTC counts down, reloads its time constant automatically, and resumes counting. Software timing loops are completely eliminated.

Interrupt processing is simplified because only one vector need be specified ; the CTC internally generates a unique vector for each channel.

The Z80C CTC requires a single + 5 V power supply and the standard Z80C single-phase system clock. It is fabricated with n-channel silicon-gate depletion-load technology, and packaged in a 28-pin plastic or ceramic DIP.

Figure 1 : Dual in Line Pin Configuration.

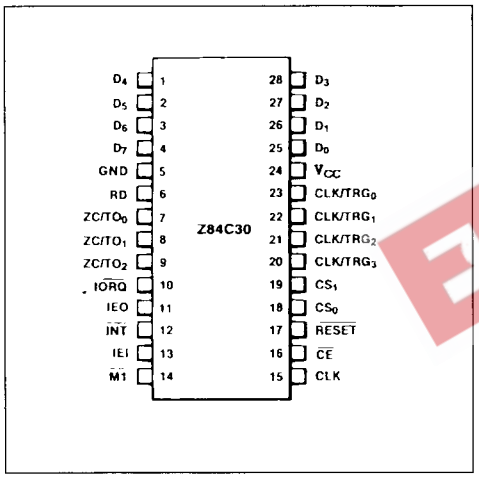
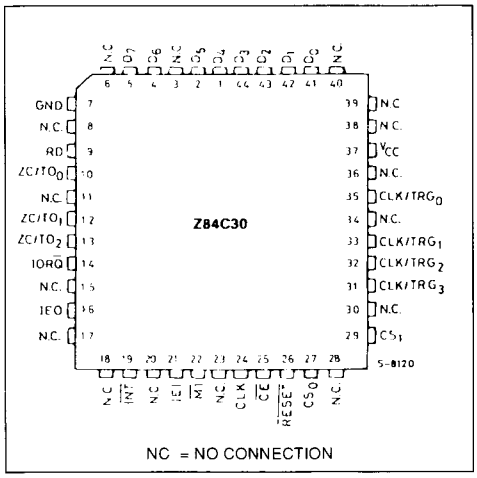


Figure 2 : Chip Carrier Pin Configuration.



NC = NO CONNECTION

FUNCTIONAL DESCRIPTION

The Z80C CTC has four independent counter/timer channels. Each channel is individually programmed with two words : a control word and a time-constant word. The control word selects the operating mode (counter or timer), enables or disables the channel interrupt, and selects certain other operating parameters. If the timing mode is selected, the control word also sets a prescaler, which divides the system clock by either 16 or 256. The time-constant word is a value from 1 to 256.

During operation, the individual counter channel counts down from the preset time constant value. In counter mode operation the counter decrements on each of the CLK/TRG input pulses until zero count is reached. Each decrement is synchronized by the system clock. For counts greater than 256, more than one counter can be cascaded. At zero count, the down-counter is automatically reset with the time constant value.

The timer mode determines time interval as small as 4 µs (Z80CA) without additional logic or software timing loops. Time intervals are generated by dividing the system clock with a prescaler that decrements a preset down-counter.

Thus, the time interval is an integral multiple of the clock period, the prescaler value (16 or 256) and the time constant that is preset in the down-counter. A timer is triggered automatically when its time constant value is programmed, or by an external CLK/TRG input.

Three channels have two outputs that occur at zero count. The first output is a zero-count/timeout pulse at the ZC/TO output. The fourth channel (Channel 3) does not have a ZC/TO output ; interrupt request is the only output available from Channel 3.

The second output is Interrupt Request ($\overline{INT}$), which occurs if the channel has its interrupt enabled during programming. When CPU acknowledges Interrupt Request, the CTC places an interrupt vector on the data bus.

The four channels of the CTC are fully prioritized and fit into four contiguous slots in a standard Z80C daisy-chain interrupt structure. Channel 0 is the highest priority and Channel 3 the lowest. Interrupts can be individually enabled (or disabled) for each of the four channels.

ARCHITECTURE

The CTC has four major elements, as shown in figure 3.

- CPU bus I/O
- Channel control logic
- Interrupt logic
- Counter/timer circuits

CPU BUS I/O

The CPU bus I/O circuit decodes the address inputs, and interfaces the CPU data and control signals to the CTC for distribution on the internal bus.

INTERNAL CONTROL LOGIC

The CTC internal control logic controls overall chip operating functions such as the chip enable, reset, and read/write logic.

INTERRUPT LOGIC

The interrupt control logic ensures that the CTC interrupts interface properly with the Z80C CPU interrupt system. The logic controls the interrupt priority of the CTC as a function of the IEI signal. If IEI is High, the CTC has priority. During interrupt processing, the interrupt logic holds IEO Low, which inhibits the interrupt operation on Lower priority devices. If the IEI input goes Low, priority is relinquished and the interrupt logic drives IEO Low.

If a channel is programmed to request an interrupt, the interrupt logic drives IEO Low at the zero count, and generates an INT signal to the CPU. When the CPU responds with interrupt acknowledge (M1 and IORQ), then the interrupt logic arbitrates the CTC internal priorities, and the interrupt control logic places a unique interrupt vector on the data bus.

If an interrupt is pending, the interrupt logic holds IEO Low. When the CPU issues a Return From Interrupt (RETI) instruction, each peripheral device decodes the first byte (ED_{16}). If the device has a pending interrupt, it raises IEO (High) for one M1 cycle. This ensures that all lower priority devices can decode the entire RETI instruction and reset properly.

COUNTER/TIMER CIRCUITS

The CTC has four independent counter/timer circuits, each containing the logic shown in figure 4.

CHANNEL CONTROL LOGIC

The channel control logic receives the 8-bit channel control word when the counter/timer channel is programmed. The channel control logic decodes the control word and sets the following operating conditions :

- Interrupt enable (or disable)
- Operating mode (timer or counter)
- Timer mode prescaler factor (16 or 256)
- Active slope for CLK/TRG input
- Timer mode trigger (automatic or CLK/TRG input)
- Time constant data word to follow
- Software reset

TIME CONSTANT REGISTER

When the counter/timer channel is programmed, the time constant register receives and stores an 8-bit time constant value, which can be anywhere from 1 to 256 (0 = 256). This constant is automatically

Figure 3 : Functional Block Diagram.

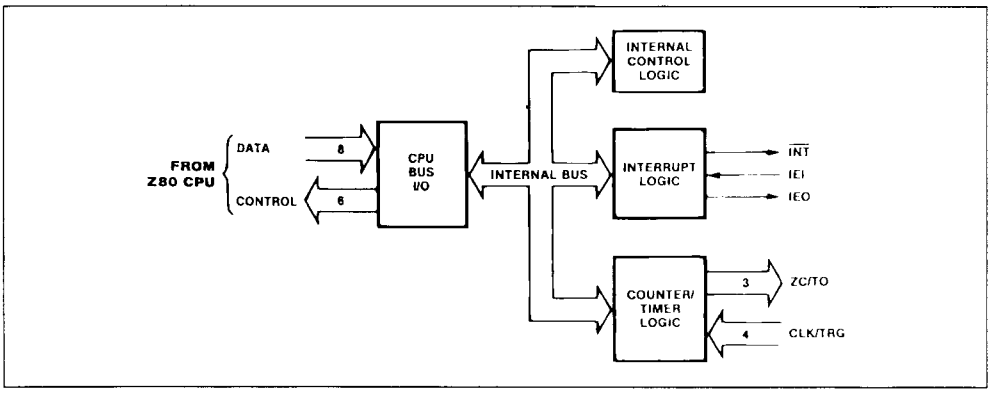
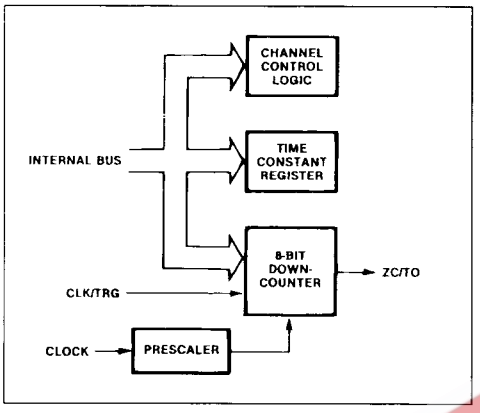


Figure 4 : Counter/ Timer Block Diagram.



loaded into the down-counter when the counter/time channel is initialized, and subsequently after each zero count.

PRESCALER

The prescaler, which is used only in timer mode, divides the system clock frequency by a factor of either 16 or 256. The prescaler output clocks the down-counter during timer operation. The effect of the prescaler on the down-counter is a multiplication of the system clock period by 16 or 256. The prescaler factor is programmed by bit 5 of the channel control word.

DOWN COUNTER

Prior to each count cycle, the down-counter is loaded with the time constant register contents. The counter is then decremented one of two ways, depending on operating mode :

- By the prescaler output (timer mode)
- By the trigger pulses into the CLK/TRG input (counter mode)

Without disturbing the down-count, the CPU can read the count remaining at any time by performing an I/O read operation at the port address assigned to the CTC channel. When the down-counter reaches the zero count, the ZC/TO output generates a positive-going pulse. When the interrupt is enabled, zero count also triggers an interrupt request signal ($\overline{INT}$) from the interrupt logic.

PROGRAMMING

Each Z80C CTC channel must be programmed prior to operation.

Programming consists of writing two words to the I/O port that corresponds to the desired channel. The first word is a control word that selects the operating mode and other parameters ; the second word is a time constant, which is a binary data word with a value from 1 to 256. A time constant word must be preceded by a channel control word.

After initialization, channels may be reprogrammed at any time. If updated control and time constant words are written to a channel during the count operation, the count continues to zero before the news time constant is loaded into the counter.

If the interrupt on any CTC channel is enabled, the programming procedure should also include an interrupt vector. Only one vector is required for all four channels, because the interrupt logic automatically modifies the vector for the channel requesting service.

A control word is identified by a 1 in bit 0. A 1 in bit 2 indicates a time constant word is to follow. Interrupt vectors are always addressed to Channel 0, and identified by a 0 in bit 0.

ADDRESSING

During programming, channels are addressed with the channel select pins CS_1 and CS_2. A 2-bit binary code selects the appropriate channel as shown in the following table

Channel	CS_1	CS_0
0	0	0
1	0	1
2	1	0
3	1	1

RESET

The CTC has both hardware and software resets. The hardware reset terminates all down-counts and disables all CTC interrupts by resetting the interrupt bits in the control registers. In addition, the ZO/TO and Interrupt outputs go inactive, IEO reflects IEI, and D_0-D_7 go to the high-impedance state. All channels must be completely reprogrammed after a hardware reset.

The software reset is controlled by bit 1 in the channel control word. When a channel receives a soft-

ware reset, it stops counting. When a software reset is used, the other bits in the control word also change the contents of the channel control register. After a software reset a new time constant word must be written to the same channel.

If the channel control word has both bits D_1 and D_2 set to 1, the addressed channel stops operating, pending a new time constant word. The channel is ready to resume after the new constant is programmed. In timer mode, if $D_3 = 0$, operation is triggered automatically when the time constant word is loaded.

CHANNEL CONTROL WORD PROGRAMMING

The channel control word is shown in figure 5. It sets the modes and parameters described below.

Interrupt Enable. D_7 enables the interrupt, so that an interrupt output ($\overline{INT}$) is generated at zero count. Interrupts may be programmed in either mode and may be enabled or disabled at any time.

OPERATING MODE

D_6 selects either timer or counter mode.

Prescaler factor (Timer Mode Only). D_5 selects factor–either 16 or 256.

Trigger slope. D_4 selects the active edge or slope of the CLK/TRG input pulses. Note that reprogramming the CLK/TRG slope during operation is equivalent to issuing an active edge. If the trigger slope is changed by a control word update while a channel is pending operation in timer mode, the result is the same as a CLK/TRG pulse and the timer starts. Similarly, if the channel is in counter mode, the counter decrements.

Trigger mode (timer mode only). D_3 selects the trigger mode for timer operation. When D_3 is reset to 0, the timer is triggered automatically. The time constant word is programmed during an I/O write operation, which takes one machine cycle. At the end of the write operation there is a setup delay of one clock period. The timer starts automatically (decrements) on the rising edge of the second clock pulse (T_2) of the machine cycle following the write operation. Once started, the timer runs continuously. At zero count the timer reloads automatically and continues counting without interruption or delay, until stopped by a reset.

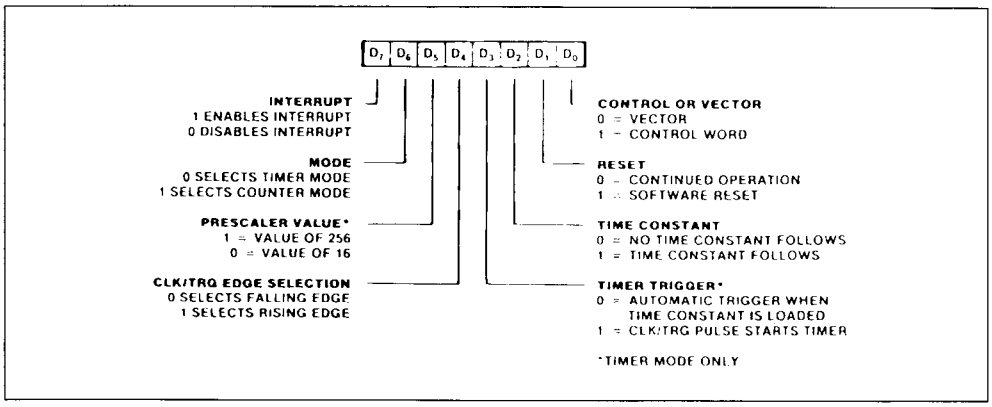
When D_3 is set to 1, the timer is triggered externally through the CLK/TRG input. The time constant word is programmed during an I/O write operation, which takes one machine cycle. The timer is ready for operation on the rising edge of the second clock pulse (T_2) of the following machine cycle. Note that the first timer decrement follows the active edge of the CLK/TRG pulse by a delay time of one clock cycle if a minimum setup time to the rising edge of clock is met. If this minimum is not met, the delay is extended by another clock period. Consequently, for immediate triggering, the CLK/TRG input must precede T_2 by one clock cycle plus its minimum setup time. If the minimum time is not met, the timer will start on the third clock cycle (T_3).

Once started the timer operates continuously, without interruption or delay, until stopped by a reset.

Time constant to follow. A 1 in D_2 indicates that the next word addressed to the selected channel is a time constant data word for the time constant register. The time constant word may be written at any time.

A 0 in D_2 indicates no time constant word is to follow. This is ordinarily used when the channel is already in operation and the new channel control word

Figure 5 : Channel Control Word..



is an update. A channel will not operate without a time constant value. The only way to write a time constant value is to write a control word with D_2 set.

Software reset. Setting D_1 to 1 causes a software reset, which is described in the Reset section.

Control word. Setting D_0 to 1 identifies the word as a control word.

TIME CONSTANT PROGRAMMING

Before a channel can start counting it must receive a time constant word from the CPU. During programming or reprogramming, a channel control word in which bit 2 is set must precede the time constant word to indicate that the next word is a time constant. The time constant word can be any value from 1 to 256 (figure 6). Note that 00_{16} is interpreted as 256.

In timer mode, the time interval is controlled by three factors :

- The system clock period (ϕ)
- The prescaler factor (P), which multiplies the interval by either 16 or 256
- The time constant (T), which is programmed into the time constant register.

Consequently, the time interval is the product of ϕ x P x T. The minimum timer resolution is 16 x ϕ (4 µs with a 4 MHz clock). The maximum timer interval is 256 x ϕ x 256 (16.4 ms with a 4 MHz clock). For longer intervals timers may be cascaded.

Figure 6 : Time Constant Word.

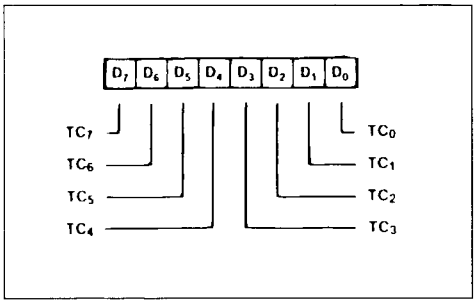
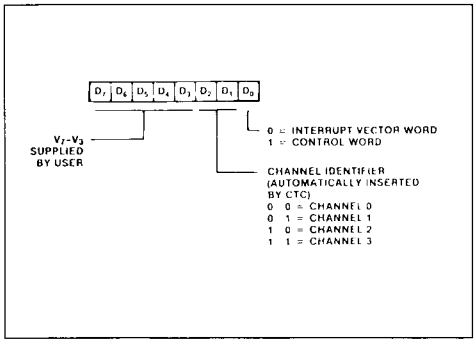


Figure 7 : Interrupt Vector Word.



INTERRUPT VECTOR PROGRAMMING

If the CTC has one or more interrupts enabled, it can supply interrupt vectors to the CPU. To do so, the CTC must be pre-programmed with the most-significant five bits of the interrupt vector. Programming consists of writing a vector word to the I/O port corresponding to the CTC Channel 0. Note that D_0 of the vector word is always zero, to distinguish the vector from a channel control word. D_1 and D_2 are not used in programming the vector word. These bits are supplied by the interrupt logic to identify the channel requesting interrupt service with a unique interrupt vector (figure 7). Channel 0 has the highest priority.

PIN DESCRIPTIONS

$\overline{CE}$. *Chip Enable* (input, active Low). When enabled the CTC accepts control words, interrupt vectors, or time constant data words from the data bus during an I/O write cycle ; or transmits the contents of the down-counter for the CPU during an I/O read cycle. In most applications this signal is decoded from the eight least significant bits of the address bus for any of the four I/Q port addresses that are mapped to the four counter-timer channels.

CLK. *System Clock* (input). Standard singlephase Z80C system clock.

CLK/TRG0-CLK/TRG3. *External Clock/Timer Trigger* (input, user-selectable active High or Low). Four pins corresponding to the four CTC channels. In counter mode, every active edge on this pin decrements the down-counter. In timer mode, an active edge starts the timer.

CS_0-CS_1. *Channel Select* (inputs active High). Two-bit binary address code selects one of the four CTC channels for an I/O write or read (usually connected to A_0 and A_1).

D_0-D_7. *System Data Bus* (bidirectional, 3-state). Transfers all data and commands between the CPU and the CTC.

IEI. *Interrupt Enable In* (input, active High). A high indicates that no other interrupting devices of higher priority in the daisy chain are being serviced by the CPU.

IEO. *Interrupt Enable Out* (output, active High). High only if IEI is High and the CPU is not servicing an interrupt from any CTC channel. IEO blocks lower priority devices from interrupting while a higher priority interrupting device is being serviced.

$\overline{INT}$. *Interrupt Request* (output, open drain, active Low). Low when any CTC channel that has been programmed to enable interrupts has a zero-count condition in its down-counter.

$\overline{IORQ}$. *Input/Output Request* (input from CPU, active Low). Used with $\overline{CE}$ and $\overline{RD}$ to transfer data and channel control words between the CPU and the CTC. During a write cycle $\overline{IORQ}$ and $\overline{CE}$ are active and $\overline{RD}$ inactive. The CTC does not receive a specific write signal ; rather, it internally generates its own from the inverse of an active $\overline{RD}$ signal. In a read cycle, $\overline{IORQ}$, $\overline{CE}$ and $\overline{RD}$ are active ; the contents of the down-counter are read by the CPU. If $\overline{IORQ}$ and M1 are both true, the CPU is acknowledging an interrupt request, and the highest priority interrupting channel places its interrupt vector on the data bus.

$\overline{M1}$. *Machine Cycle One* (input from CPU, active Low). When $\overline{M1}$ and $\overline{IORQ}$ are active, the CPU is acknowledging an interrupt. The CTC then places an interrupt vector on the data bus if it has highest priority, and if a channel has requested an interrupt ($\overline{INT}$).

$\overline{RD}$. *Read Cycle Status* (input, active Low). Used in conjunction with $\overline{IORQ}$ and $\overline{CE}$ to transfer data and channel control words between the CPU and the CTC.

$\overline{RESET}$. *Reset* (input active Low). Terminates all down-counts and disables all interrupts by resetting the interrupt bits in all control registers ; the ZC/TO and the Interrupt outputs go inactive ; IEO reflects IEI ; D0-D7 go to the high-impedance state.

ZC/TO_0-ZC/TO_2. *Zero Count/Timeout* (output, active High). Three ZC/TO pins corresponding to CTC channels 2 through 0 (Channel 3 has no ZC/TO pin). In both counter and timer modes the output is an active High pulse when the down-counter decrements to zero.

TIMING

READ CYCLE TIMING

Figure 9 shows read cycle timing. This cycle reads the contents of a down-counter without disturbing the count. During clock cycle T_2, the CPU initiates a read cycle by driving the following inputs Low : $\overline{RD}$, $\overline{IORQ}$, and $\overline{CE}$. A 2-bit binary code at inputs CS_1 and CS_0 selects the channel to be read. M1 must be High to distinguish this cycle from an interrupt acknowledge. No additional wait states are allowed.

WRITE CYCLE TIMING

Figure 10 shows write cycle timing for loading control, time constant or vector words.

The CTC does not have a write signal input, so it generates one internally when the read ($\overline{RD}$) input is High during T_1. During T_2 $\overline{IORQ}$ and $\overline{CE}$ inputs are Low. M1 must be High to distinguish a write cycle from an interrupt acknowledge. A 2-bit binary code at inputs CS_1 and CS_0 selects the channel to be addressed, and the word being written is placed on the data bus. The data word is latched into the appropriate register with the rising edge of clock cycle T_3.

TIMER OPERATION

In the timer mode, a CLK/TRG pulse input starts the timer (figure 11) on the second succeeding rising

Figure 8 : A Typical Z80C Environment.

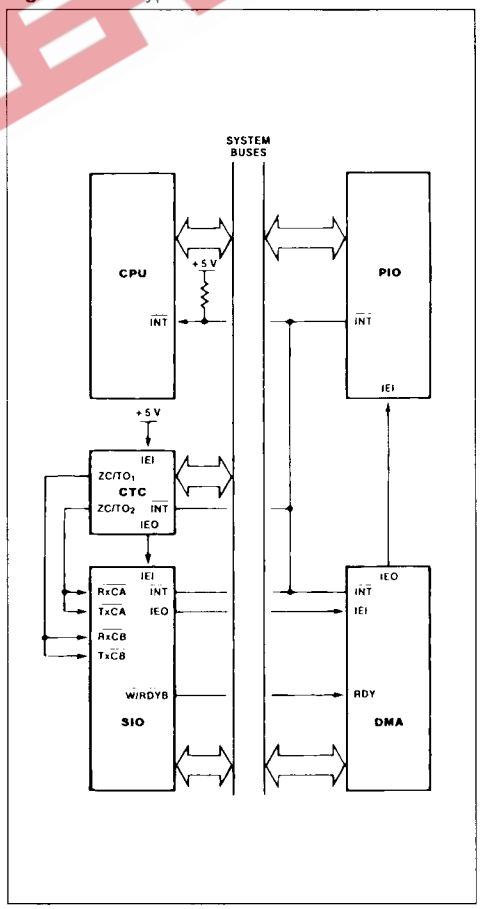


Figure 9 : Read Cycle Timing.

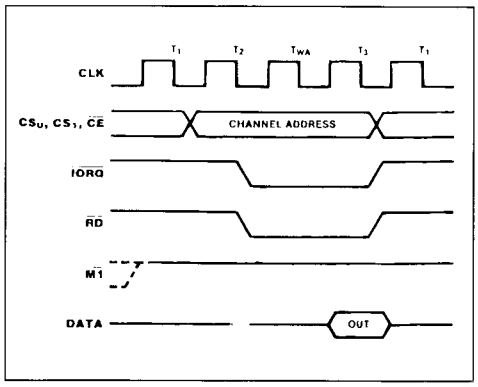
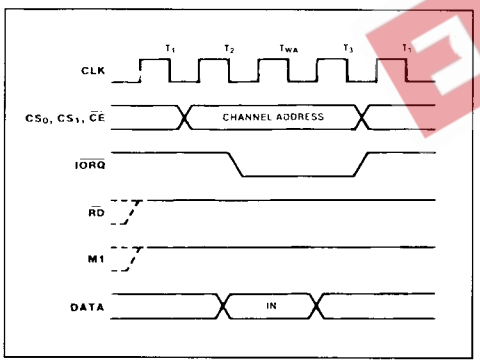


Figure 10 : Write Cycle Timing.



edge of CLK. The trigger pulse is asynchronous and it must have a minimum width. A minimum lead time (210 ns) is required between the active edge of the CLK/TRG and the next rising edge of CLK to enable the prescaler on the following clock edge. If the CLK/TRG edge occurs closer than this, the initiation of the timer function is delayed one clock cycle. This corresponds to the startup timing discussed in the programming section. The timer can also be started automatically if so programmed by the channel control word.

COUNTER OPERATION.

In the counter mode, the CLK/TRG pulse input decrements the downcounter. The trigger is asynchronous, but the count is synchronized with CLK. For the decrement to occur on the next rising edge of CLK, the trigger edge must precede CLK by a minimum lead time as shown in figure 12. If the lead time is less than specified, the count is delayed by one clock cycle. The trigger pulse must have a minimum width, and the trigger period must be at least twice the clock period.

The ZC/TO output occurs immediately after zero count, and follows the rising CLK edge.

Figure 11 : Timer Mode Timing.

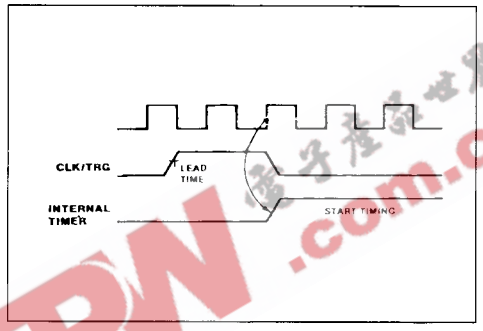
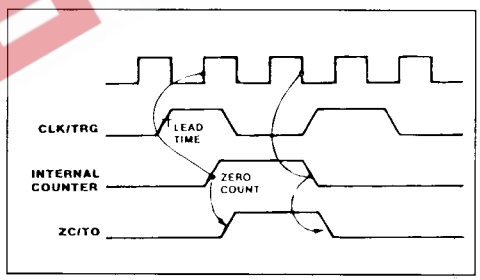


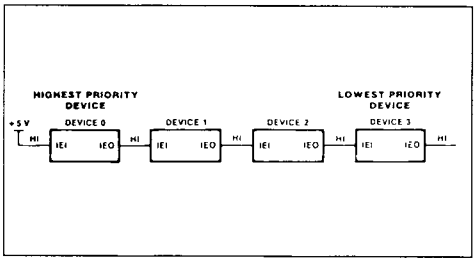
Figure 12 : Counter Mode Timing.



INTERRUPT OPERATION

The CTC follows the Z80C system interrupt protocol for nested priority interrupts and return from interrupt, wherein the interrupt priority of a peripheral is determined by its location in a daisy chain. Two lines–IEI and IEO–in the CTC connect it to the system daisy chain. The device closest to the + 5 V supply has the highest priority (figure 13). For additional information on the Z80C interrupt structure, refer *to the Z80 CPU Technical Manual.*

Figure 13 : Daisy-chain Interrupt Priorities.



INTERRUPT OPERATION

Within the CTC, interrupt priority is predetermined by channel number : Channel 0 has the highest priority, and Channel 3 the lowest. If a device or channel is being serviced with an interrupt routine, it cannot be interrupted by a device or channel with lower priority until service is complete. Higher priority devices or channels may interrupt the servicing of lower priority devices or channels.

A CTC channel may be programmed to request an interrupt every time its down-counter reaches zero. Note that the CPU must be programmed for interrupt mode 2. Some time after the interrupt request, the CPU sends an interrupt acknowledge. The CTC interrupt control logic determines the highest priority channel that is requesting an interrupt. Then, if the CTC IEI input is High (indicating that it has priority within the system daisy chain) it places an 8-bit interrupt vector on the system data bus. The high-order five bits of this vector were written to the CTC during the programming process ; the next two bits are provided by the CTC interrupt control logic as a binary code that identifies the highest priority channel requesting an interrupt ; the low-order bit is always zero.

INTERRUPT ACKNOWLEDGE TIMING

Figure 14 shows interrupt acknowledge timing. After an interrupt request, the CPU sends an interrupt acknowledge ($\overline{M1}$ and $\overline{IORQ}$). All channels are inhibited from changing their interrupt request status when $\overline{M1}$ is active–about two clock cycles earlier than $\overline{IORQ}$. $\overline{RD}$ is High to distinguish this cycle from an instruction fetch.

The CTC interrupt logic determines the highest priority channel requesting an interrupt. If the CTC interrupt enable input (IEI) is High, the highest priority interrupting channel within the CTC places its interrupt vector on the data bus when $\overline{IORQ}$ goes Low. Two wait states (T_{WA}) are automatically inserted at this time to allow the daisy chain to stabilize. Additional wait states may be added.

RETURN FROM INTERRUPT TIMING

At the end of an interrupt service routine the RETI (Return From Interrupt) instruction initializes the daisy chain enable lines for proper control of nested priority interrupt handling. The CTC decodes the 2-byte RETI code internally and determines whether it is intended for a channel being serviced. Figure 15 shows RETI timing.

If several Z80C peripherals are in the daisy chain, IEI settles active (High) on the chip currently being serviced when the opcode ED_{16} is decoded. If the following opcode is $4D_{16}$, the peripheral being serviced is released and its IEO becomes active. Additional wait states are allowed.

Figure 14 : Interrupt Acknowledge Timing.

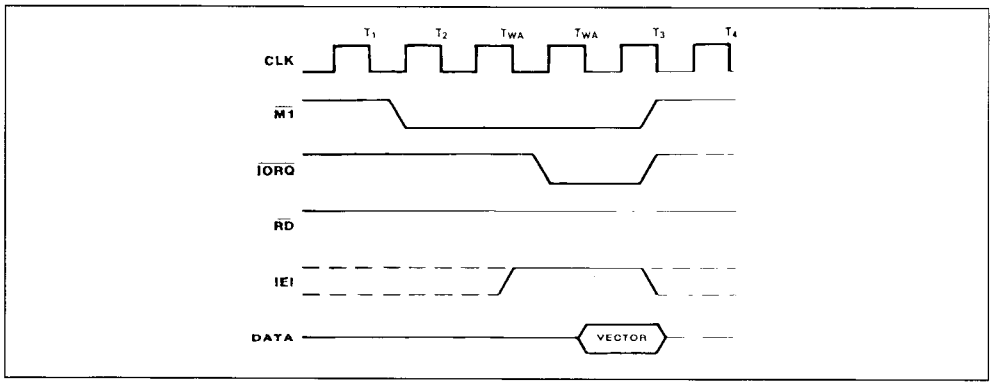
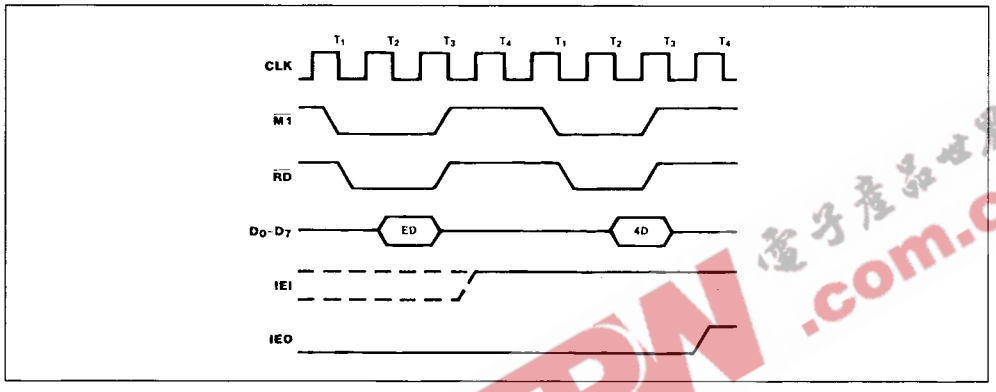
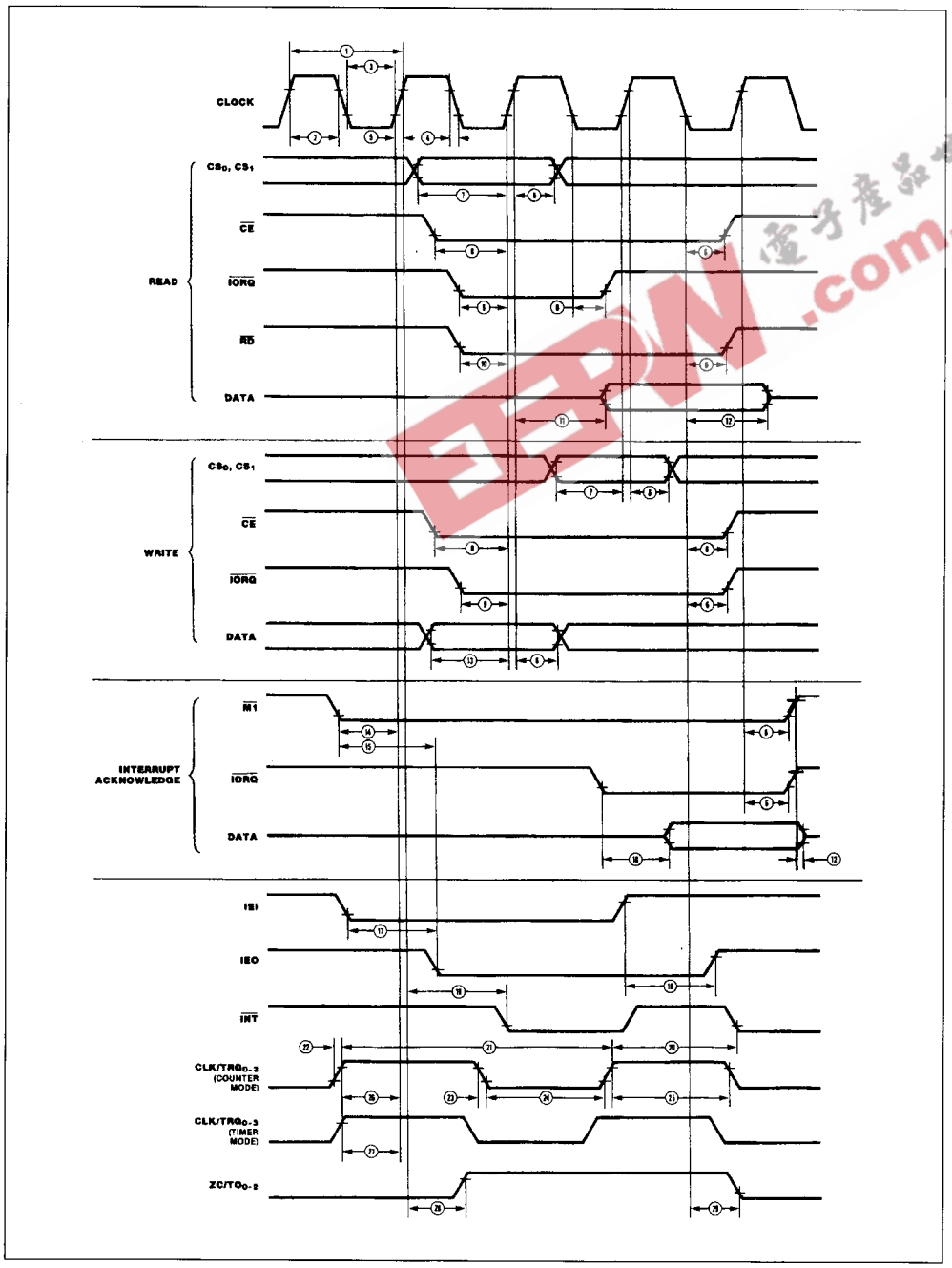


Figure 15 : Return From Interrupt Timing.



AC CHARACTERISTICS



AC CHARACTERISTICS (continued)

N°	Symbol	Parameter	Z84C30A		Z84C30B	
			Min. (ns)	Max. (ns)	Min. (ns)	Max. (ns)
1	TcC	Clock Cycle Time	250		165	
2	TwCh	Clock Width (high)	105		65	
3	TwCl	Clock Width (low)	105		65	
4	TfC	Clock Fall Time		30		20
5	TrC	Clock Rise Time		30		20
6	Th	All Hold Times	0		0	
7	TsCS(C)	CS to Clock ↑ Setup Time	160		100	
8	TsCE(C)	$\overline{CE}$, to Clock ↑ Setup Time	150		100	
9	TsIO(C)	$\overline{IORQ}$ ↓ to Clock ↑ Setup Time	115		100	
10	TsRD(C)	$\overline{RD}$, ↓ to Clock ↑ Setup Time	115		70	
11	TdC(DO)	Clock ↑ to Data Out Delay		200		130
12	TdC(DOz)	$\overline{IORQ}$, $\overline{RD}$ ↑ to Data Float		110		90
13	TsDI(C)	Data in to Clock ↑ Setup Time	50		40	
14	TsMI(C)	$\overline{MI}$ to Clock ↑ Setup Time	90		70	
15	TdMI(IEO)	$\overline{MI}$ ↓ to IEO ↓ Delay (interrupt immediately preceding $\overline{MI}$)		190		130
16	TdIO(DOI)	$\overline{IORQ}$ ↓ to Data Out Delay (INTA cycle)		160		110
17	TdIEI(IEOf)	IEI ↓ to IEO ↓ Delay		130		100
18	TdIEI(IEOr)	IEI ↑ to IEO ↑ Delay (after ED decode)		160		110
19	TdC(INT)	Clock ↑ to $\overline{INT}$ ↓ Delay		(1)TcC+140		(1)TcC+120
20	TdCLK(INT)	CLK/TRG ↑ to $\overline{INT}$ ↓ tsCTR(C) satisfied tsCTR(C) not satisfied		(2) TcC+160 2TcC+370		(2) TcC+160 2TcC+370
21	TcCTR	CLK/TRG Cycle Time	(2) 2TcC			(2) 40
22	TrCTR	CLK/TRG Rise Time		50		40
23	TfCTR	CLK/TRG Fall Time		50		40
24	TwCTRl	CLK/TRG Width (low)	200		120	
25	TwCTRh	CLK/TRG Width (high)	200		120	
26	TsCTR(Cs)	CLK/TRG ↑ to Clock ↑ Setup Time for Immediate Count	(2) 210		(2) 150	
27	TsCTR(Cs)	CLK/TRG ↑ to Clock ↑ Setup Time for enabling of Prescaler on following Clock ↑	(1) 210		(1) 150	
28	TdC(ZC/TOr)	Clock ↑ to ZC/TO ↑ Delay		190		140
29	TdC(ZC/TOf)	Clock ↓ to ZC/TO ↓ Delay		190		140

Notes : 1. Timer mode.
2. Counter mode.

ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_{CC}	V_{CC} Supply Voltage with Respect to V_{SS}	– 0.5 to 7	V
V_{IN}	Input Voltage	– 0.5 to V_{CC} + 0.5	V
P_D	Power Dissipation (T_A = 85 °C)	250	mW
T_{SOLDER}	Soldering Temperature (soldering time 10 sec)	260	°C
T_{stg}	Storage Temperature	– 65 to 150	°C
T_{opr}	Operating Temperature	– 40 to 85	°C

DC CHARACTERISTICS (1)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V_{ILC}	Clock Input Low Voltage		– 0.3	–	0.6	V
V_{IHC}	Clock Input High Voltage		V_{CC} – 0.6	–	V_{CC} + 0.3	V
V_{IL}	Input Low Voltage (except CLK)		– 0.5	–	0.8	V
V_{IH}	Input High Voltage (except CLK)		2.2	–	V_{CC}	V
V_{OL}	Output Low Voltage	I_{OL} = 2.0 mA	–	–	0.4	V
V_{OH1}	Output High Voltage (1)	I_{OH} = – 1.6 mA	2.4	–	–	V
V_{OH2}	Output High Voltage (2)	I_{OH} = – 250 µA	V_{CC} – 0.8	–	–	V
I_{LI}	Input Leakage Current	$V_{SS} \le V_{IN} \le V_{CC}$	–	–	± 10	µA
I_{LO}	3-State Output Leakage Current in Float	V_{SS} + 0.4 ≤ $V_{OUT} \le V_{CC}$	–	–	± 10	µA
I_{CC1}	Power Supply Current 4 MHz 6 MHz	 V_{CC} = 5 V, CLK = 4 MHz $V_{IH} = V_{CC} - 0.2$ V, V_{IL} = 0.2 V	 – –	 2 4	 5 7	 mA mA
I_{CC2}	Stand-by Supply Current	V_{CC} = 5 V, CLK = V_{CC} $V_{IH} = V_{CC} - 0.2$ V V_{IL} = 0.2 V	–	0.5	10	µA
I_{OHD}	Darlington Drive Current (1)	V_{OH} = 1.5 V, R_{EXT} = 1.1 kΩ	– 1.5	–	– 5.0	mA

Note : 1. Applied to $ZC/T0_0$, $ZC/T0_1$ and $ZC/T0_2$.

TEST CONDITIONS

T_A = – 40 °C to + 85 °C

V_{CC} = 5 V ± 10 %

V_{SS} = 0 V.

AC TEST CONDITIONS.

- Inputs except CLK (clock) are driven at 2.4 V for a logic "1" and 0.4 V for a logic "0". Clock input is driven at V_{CC}–0.6 V for a logic "1" and 0.6 V for a logic "0".
- Timing measurements are made at 2.2 V for a logic "1" and 0.8 V for a logic "0".

All AC parameters assume a load capacitance of 100 pF.

ORDERING INFORMATION

Type	Package	Temp.	Clock	Description
Z84C30AB6	DIP-28 (plastic)	−40/+ 85°C	4 MHz	Z80C Counter Timer Control CMOS Version
Z84C30AD6	DIP-28 (ceramic)	−40/+ 85°C		
Z84C30AD2	DIP-28 (ceramic)	−55/+125°C		
Z84C30AC6	PLCC44 (plastic chip-carrier)	−40/+ 85°C		
Z84C30BB6	DIP-28 (plastic)	−40/+ 85°C	6 MHz	
Z84C30BD6	DIP-28 (ceramic)	−40/+ 85°C		
Z84C30BD2	DIP-28 (ceramic)	−55/+125°C		
Z84C30BC6	PLCC44 (plastic chip-carrier)	−40/+ 85°C		