Z86D990/Z86D991 OTP and Z86L99X ROM

Low-Voltage Micro-controllers with ADC

Preliminary Product Specification

PS003807-1002

ZiLOG Worldwide Headquarters • 532 Race Street • San Jose, CA 95126-3432
Telephone: 408.558.8500 • Fax: 408.558.8300 • www.ZiLOG.com

This publication is subject to replacement by a later edition. To determine whether a later edition exists, or to request copies of publications, contact:

ZiLOG Worldwide Headquarters
532 Race Street
San Jose, CA 95126-3432
Telephone: 408.558.8500
Fax: 408.558.8300
www.ZiLOG.com

Document Disclaimer

Table of Contents

List of Figures

List of Tables

Architectural Overview

The Z86D99 is a low-voltage general-purpose one-time programmable (OTP) Z8® microcontroller with an integrated four-channel 8-bit sigma delta analog-to-digital converter. The Z86L99 is the read-only memory (ROM) version of this controller.

The Z86D99/Z86L99 family is designed to be used in a wide variety of embedded control applications including battery chargers, home appliances, infrared (IR) remote controls, security systems, and wireless keyboards.

It has three counter/timers, a general-purpose 8-bit counter/timer with a 6-bit prescaler and an 8-bit/16-bit counter/timer pair that can be used individually for general-purpose timing or as a pair to automate the generation and reception of complex pulses or signals. Unique features of the Z86D99/Z86L99 family of products include 489 bytes of general-purpose random-access memory (RAM), 256 bytes of which are mapped into the program memory space and can be used to store data variables or as executable RAM, a low-battery detection flag, and a controlled current output pin, which is a regulated current source that sinks a predefined current (I_{CCO}). Table 1 highlights the basic product features of these microcontrollers.

Table 1. Z86L99/Z86D99 Feature Comparison

	Pins	I/O	Memory (Bytes)	Operating Voltage (V)	ADC	Timers	Watch-Dog Timer
Z86D990	40/48	32	32K OTP	3.0–5.5	4 channel	3	Yes
Z86D991	28	24	32K OTP	3.0–5.5	—	3	Yes
Z86L990	40/48	32	16K ROM	2.3–5.5	4 channel	3	Yes
Z86L991	28	24	16K ROM	2.3–5.5	—	3	Yes
Z86L996	28	24	4K ROM	2.3–5.5	—	3	Yes
Z86L997	28	24	8K ROM	2.3–5.5	—	3	Yes

The Z8 microcontroller core offers more flexibility and performance than accumulator-based microcontrollers. All 256 general-purpose registers, including dedicated input/output (I/O) port registers, can be used as accumulators. This unique register-to-register architecture avoids accumulator bottlenecks for high code efficiency. The registers can be used as address pointers for indirect addressing, as index registers, or for implementing an on-chip stack.

The Z8 has a sophisticated interrupt structure and automatically saves the program counter and status flags on the stack for fast context-switching. Speed of execution and smooth programming are also supported by a “working register area” with short 4-bit register addresses.

The Z8 instruction set, consisting of 43 basic instructions, is optimized for high-code density and reduced execution time. It is similar in form to the ZiLOG Z80 instruction set. The eight instruction types and six addressing modes together with the ability to operate on bits, 4-bit nibbles or binary coded decimal (BCD) digits, 8-bit bytes, and 16-bit words, make for a code-efficient, flexible microcontroller.

Features

- Four-channel, 8-bit sigma delta analog-to-digital (A/D) converter with external voltage references (not available in the 28-pin configuration)
- Two independent analog comparators
- Controlled current output
- 489 bytes of RAM
  - 233 bytes of general-purpose register-based RAM
  - 256 bytes of RAM mapped into the program memory space that can be used as data RAM or executable RAM
- 32 Kbytes of OTP memory (Z86D99X)
- 16 Kbytes of ROM (Z86L99X)

Counter/Timers

- Special architecture to automate generation and reception of complex pulses or signals:
  - Programmable 8-bit counter/timer (T8) with two 8-bit capture registers and two 8-bit load registers
  - Programmable 16-bit counter/timer (T16) with one 16-bit capture register pair and one 16-bit load register pair
  - Programmable input glitch filter for pulse reception
- One general-purpose 8-bit counter/timer (T1) with 6-bit prescaler

Input/Output and Interrupts

- Thirty-two I/Os, twenty-nine of which are bidirectional I/Os with programmable resistive pull-up transistors (24 I/Os are available in the 28-pin configuration)
- Sixteen I/Os are selectable as stop-mode recovery sources
- Six interrupt vectors with nine interrupt sources
  - Three external sources
  - Two comparator interrupts

  - Three timer interrupts
  - One low-battery detector flag

Operating Characteristics

- 8-MHz operation
- 3.0 V to 5.5 V operating voltage (Z86D990/Z86D991)
- 2.3 V to 5.5 V operating voltage (Z86L990/Z86L991)
- Low power consumption with three standby modes:
  - Stop
  - Halt
  - Low Voltage Standby
- Low-battery detection flag
- Low-voltage protection circuit (also known as V_{BO}, or voltage brownout, circuit)
- Watch-dog timer and power-on reset circuits

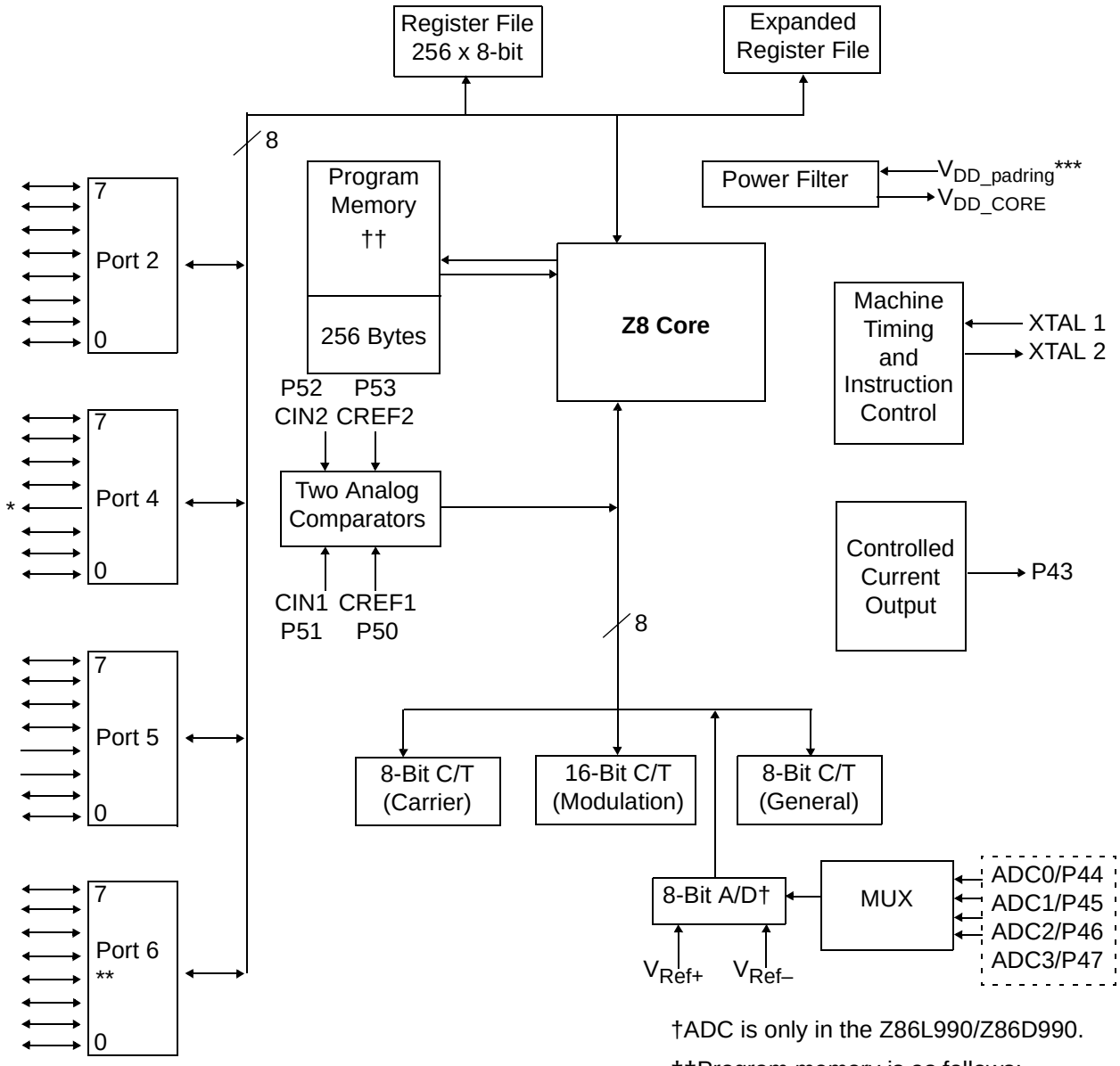
User-Programmable Option Bits

- Clock source—RC/other (LC, resonator, or crystal)
- Watch-dog timer permanently enable
- 32-kHz crystal
- Port 20–27 pull-up resistive transistor
- Port 40–42 pull-up resistive transistor
- Port 44–47 pull-up resistive transistor
- Port 50–51 pull-up resistive transistor
- Port 54–57 pull-up resistive transistor
- Port 60–63 pull-up resistive transistor (not available in Z86D991/Z86L991)
- Port 64–67 pull-up resistive transistor (not available in Z86D991/Z86L991)
- P43 high impedance in STOP mode (available in OTP only)

  Force P43 to output a 1 in the open-drain configuration

Functional Block Diagram

Figure 1 shows the functional block diagram for the microcontrollers.



*Controlled Current Output

**P6 is only in the Z86L990/Z86D990.

***In the 28-pin package, $V_{DD\_padring}$ and V_{DD_CORE} are bonded together.

†ADC is only in the Z86L990/Z86D990.

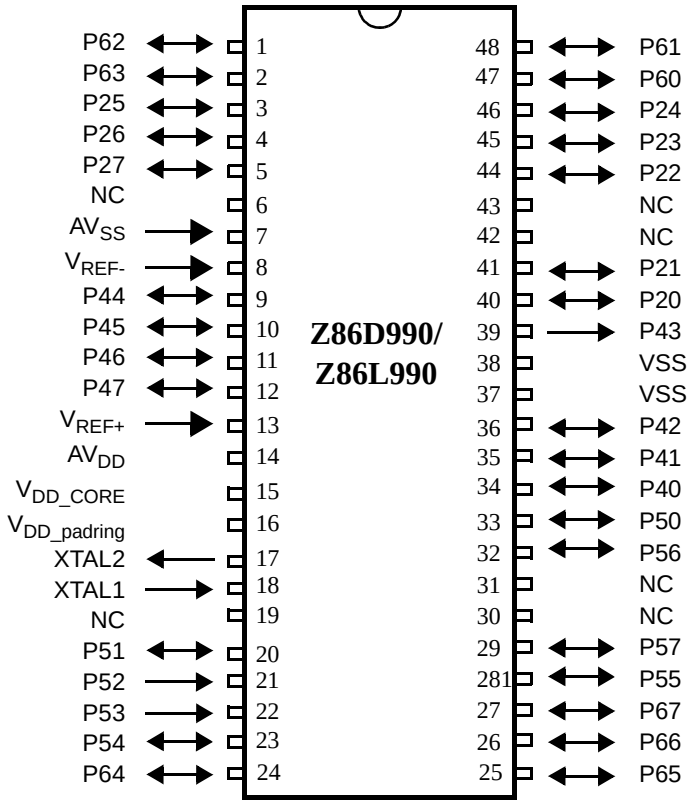
††Program memory is as follows:

Z86D990	32K OTP
Z86D991	32K OTP
Z86L990	16K ROM
Z86L991	16K ROM

Figure 1. Functional Block Diagram

Pin Descriptions

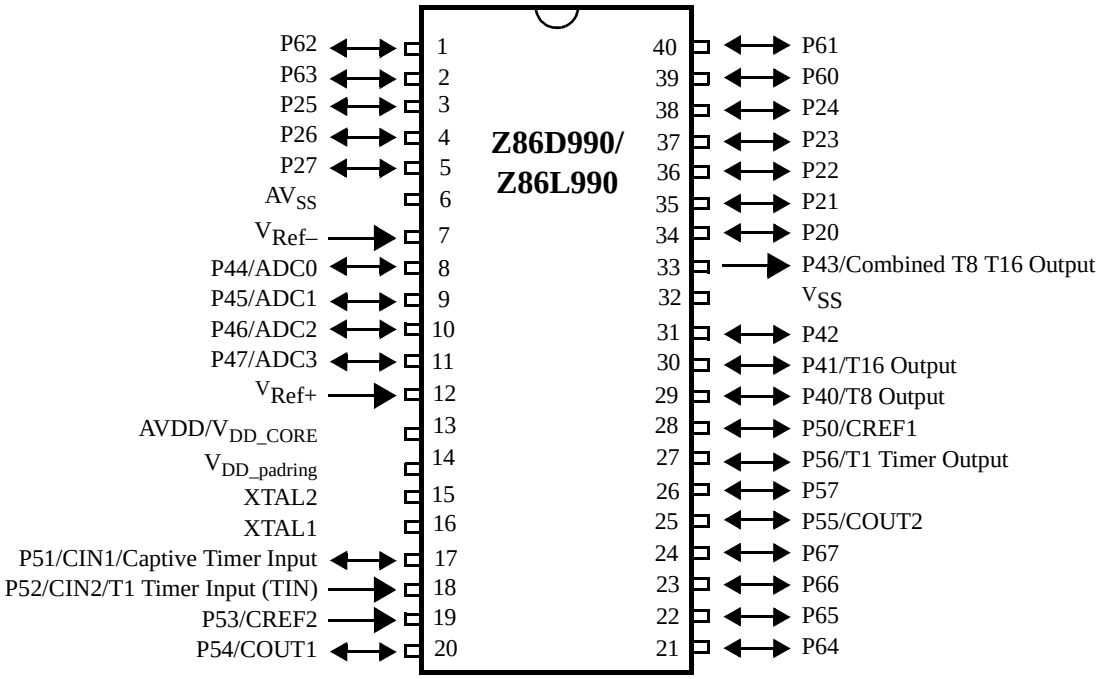
Figure 2 through Figure 4 show the pin names and locations.



Notes:

1. Both V_{SS} pins must be connected to ground.
2. NC is no connection to the die.
3. AV_{DD} must be connected to V_{DD_CORE} and a 10-µF capacitor for good A/D conversion.
4. Power must be connected to $V_{DD\_padring}$. Current passes to V_{DD_CORE} through the internal power filter.

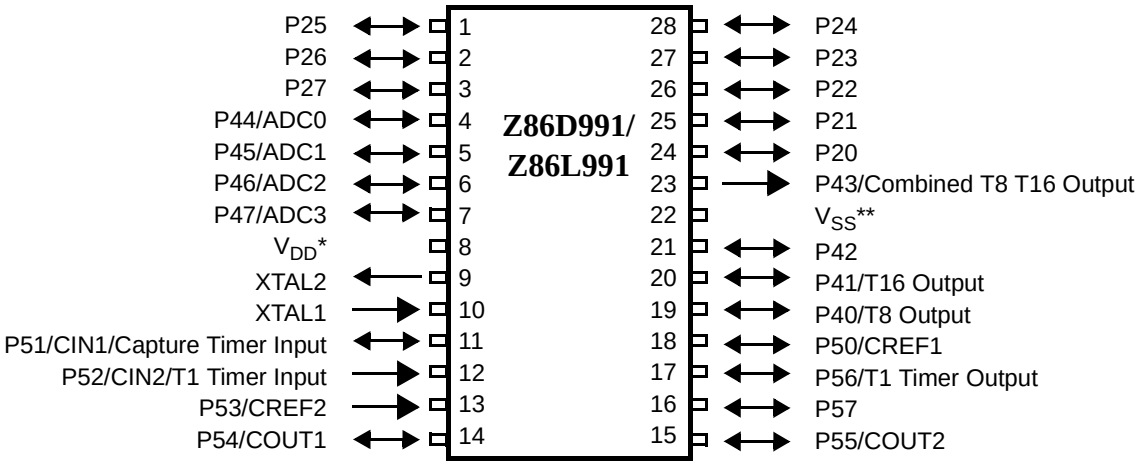
Figure 2. 48-Pin SSOP Pin Assignments



Notes:

1. AV_{DD} must be connected to V_{DD_CORE} and a 10-µF capacitor for good A/D conversion.
2. Power must be connected to $V_{DD\_padring}$. Current passes to V_{DD_CORE} through the internal power filter.

Figure 3. 40-Pin DIP Pin Assignment



Notes:

1. P43 is a controlled current output.
2. P54, P55, P56, and P57 are high drive outputs.

* $V_{DD} = V_{DD\_CORE} + V_{DD\_padring} + AV_{DD}$

Figure 4. 28-Pin SOIC/DIP Pin Assignment—User Mode

Pins Configuration

Table 2 describes the pins.

Table 2. Pin Descriptions

Symbol	Pin # 28 PDIP/SOIC	40 PDIP	48 SSOP	Direction	Description
P20	24	34	40	I/O	Port 2 Bit 0
P21	25	35	41	I/O	Port 2 Bit 1
P22	26	36	44	I/O	Port 2 Bit 2
P23	27	37	45	I/O	Port 2 Bit 3
P24	28	38	46	I/O	Port 2 Bit 4
P25	1	3	3	I/O	Port 2 Bit 5
P26	2	4	4	I/O	Port 2 Bit 6
P27	3	5	5	I/O	Port 2 Bit 7
P40	19	29	34	I/O	Port 4 Bit 0, T8 Output

Table 2. Pin Descriptions (Continued)

Symbol	Pin # 28 PDIP/SOIC	40 PDIP	48 SSOP	Direction	Description
P41	20	30	35	I/O	Port 4 Bit 1, T16 Output
P42	21	31	36	I/O	Port 4 Bit 2
P43	23	33	39	Output	T8/T16 Output, Controlled current output
P44	4	8	9	I/O	Port 4 Bit 4, A/D Channel 0*
P45	5	9	10	I/O	Port 4 Bit 5, A/D Channel 1*
P46	6	10	11	I/O	Port 4 Bit 6, A/D Channel 2*
P47	7	11	12	I/O	Port 4 Bit 7, A/D Channel 3*
P50, CREF1	18	28	33	I/O	Port 5 Bit 0, Comparator 1 reference
P51, CIN1	11	17	20	I/O	Port 5 Bit 1, Capture timer input, IRQ_2
P52, CIN2	12	18	21	Input	Port 5 Bit 2, Timer 1 timer input, IRQ_0
P53, CREF2	13	19	22	Input	Port 5 Bit 3, Comparator 2 reference, IRQ_1
P54	14	20	23	I/O	Port 5 Bit 4, High drive output
P55	15	25	28	I/O	Port 5 Bit 5, High drive output
P56	17	27	32	I/O	Port 5 Bit 6, Timer 1 output, High drive output
P57	16	26	29	I/O	Port 5 Bit 7, High drive output
P60		39	47	I/O	Port 6 Bit 0
P61		40	48	I/O	Port 6 Bit 1
P62		1	1	I/O	Port 6 Bit 2
P63		2	2	I/O	Port 6 Bit 3
P64		21	24	I/O	Port 6 Bit 4
P65		22	25	I/O	Port 6 Bit 5
P66		23	26	I/O	Port 6 Bit 6
P67		24	27	I/O	Port 6 Bit 7
XTAL1	10	16	18	Input	Crystal, Oscillator clock
XTAL2	9	15	17	Output	Crystal, Oscillator clock
AV_{DD}		13	14		Analog power supply
V_{DD_CORE}		13	15		Z8 core power supply
AV_{SS}		6	7		Analog ground
V_{Ref-}		7	8	Input	A/D converter lower reference
V_{Ref+}		12	13	Input	A/D converter upper reference
$V_{DD\_padring}$	8**	14	16		Power supply (pad ring)
V_{SS}	22**	32	37, 38		Ground

Notes: *A/D converter is not available in the 28-pin configuration.
**In the 28-pin configuration, all three (core, pad ring, and analog) powers are tied together.

Operational Description

Central Processing Unit (CPU) Description

The Z8 architecture is characterized by a flexible I/O scheme, an efficient register and address space structure and a number of ancillary features for cost-sensitive, high-volume embedded control applications. ROM-based products are geared for high-volume production (where the software is stable) and one-time programmable equivalents for prototyping as well as volume production where time to market or code flexibility is critical.

Architecture Type

The Z8 register-oriented architecture centers around an internal register file composed of 256 consecutive bytes, known as the standard register file. The standard register file consists of 4 I/O port registers (R2, R4, R5, and R6), 12 control and status registers, 233 general-purpose registers, and 7 registers reserved for future expansion. In addition to the standard register file, the Z86D99/Z86L99 family uses 21 control and status registers located in the Z8 expanded register file. Any general-purpose register can be used as an accumulator and address pointer or an index, data, or stack register.

All active registers can be referenced or modified by any instruction that accesses an 8-bit register, without the requirement for special instructions. Registers accessed as 16 bits are treated as even-odd register pairs. In this case, the data's most significant byte (MSB) is stored in the even-numbered register, while the least significant byte (LSB) goes into the next higher odd-numbered register.

The Z8 CPU has an instruction set designed for the large register file. The instruction set provides a full compliment of 8-bit arithmetic and logical operations. BCD operations are supported using a decimal adjustment of binary values, and 16-bit quantities for addresses and counters can be incremented and decremented. Bit manipulation and Rotate and Shift instructions complete the data-manipulation capabilities of the Z8 CPU. No special I/O instructions are necessary because the I/O is mapped into the register file.

CPU Control Registers

The standard Z8 control registers govern the operation of the CPU. Any instruction which references the register file can access these control registers. The following are available control registers:

- Register Pointer (RP)
- Stack Pointer (SP)
- Program Control Flags (FLAGS)

- Interrupt Control (IPR, IMR, and IRQ)
- Stop Mode Recovery (SMR, P2SMR, and P5SMR)
- Low-Battery Detect (LB) Flag

The Z8 uses a 16-bit Program Counter (PC) to determine the sequence of current program instructions. The PC is not an addressable register.

Peripheral registers are used to transfer data, configure the operating mode, and control the operation of the on-chip peripherals. Any instruction that references the register file can access the peripheral registers. The following are peripheral control registers:

- Analog/Digital Converter (ADCCTRL and ADCDATA)
- T1 Timer/Counter (TMR, T1, and PRE1)
- T8 Timer/Counter (CTR0, HI8, LO8, TC8H, and TC8L)
- T16 Timer/Counter (CTR2, HI16, LO16, TC16H, and TC16L)
- T8/T16 Control Registers (CTR1and CTR3)

In addition, the four port registers are considered to be peripheral registers. The following are port control registers:

- Port Configuration Registers (P456CON and P3M)
- Port 2 Control and Mode Registers (P2 and P2M)
- Port 4 Control and Mode Registers (P4 and P4M)
- Port 5 Control and Mode Registers (P5 and P5M)
- Port 6 Control and Mode Registers (P6 and P6M)

The functions and applications of the control and peripheral registers are explained in “Control and Status Registers” on page 52.

Memory (ROM/OTP and RAM)

There are four basic address spaces available to support a wide range of configurations:

- Program memory (on-chip)
- Standard register file
- Expanded register file
- Executable RAM

The Z8 standard register file totals up to 256 consecutive bytes organized as 16 groups of 16 eight-bit registers. These registers consist of I/O port registers,

general-purpose RAM registers, and control and status registers. Every RAM register acts like an accumulator, speeding instruction execution and maximizing coding efficiency. Working register groups allow fast context switching.

The standard register file of the Z8 (known as Bank 0) has been expanded to form 16 expanded register file (ERF) banks. The expanded register file allows for additional system control registers and for the mapping of additional peripheral devices into the register area. Each ERF bank can potentially consist of up to 256 registers (the same amount as in the standard register file) that can then be divided into 16 working register groups. Currently, only Group 0 of ERF Banks F and D (0Fh and 0Dh) has been implemented.

In addition to the standard program memory and the RAM register files, the Z86D99/Z86L99 family also has 256 bytes of executable RAM that has been mapped into the upper 256 bytes of the program memory address space (FF00h–FFFFh). Data can be written to the executable RAM by using the LDC instruction.

Program Memory Structure

The first 12 bytes of program memory are reserved for the interrupt vectors. These locations contain six 16-bit vectors that correspond to the six available interrupts (IRQ_0 through IRQ_5.) Address 12 (0Ch) up to 32,767 (7FFFh) consists of on-chip one-time programmable memory. The Z86L99X only has the 4K/8K/16K ROM size.

After any reset operation (power-on reset, watch-dog timer time out, and stop mode recovery), program execution resumes with the initial instruction fetch from location 000Ch. After a reset, the first routine executed must be one that initializes the control registers to the required system configuration.

A unique feature of the Z86D99/Z86L99 family is the presence of 256 bytes of on-chip executable RAM. This random-access memory is in addition to the standard Z8 register file memory available on all Z8 microcontrollers. As illustrated in Figure 5, the executable RAM is mapped into the upper 256 bytes of the 64K program memory address space (FF00h–FFFFh). Data can be written to the executable RAM by using the LDC instruction.

Memory locations between 8000h and FEFFh have not been implemented on the Z86D99X microcontrollers.

The Z86D99/Z86L99 family does not have the capability of accessing external memory.

Location (Hex)	
FFFF FF00	256 bytes **Executable RAM**
	Not Implemented
3FFF/7FFF (ROM)/(OTP) 000C	**PROGRAM MEMORY** Location of the first byte of the initial instruction executed after RESET
000B	IRQ_5 (lower byte)
000A	IRQ_5 (upper byte)
0009	IRQ_4 (lower byte)
0008	IRQ_4 (upper byte)
0007	IRQ_3 (lower byte)
0006	IRQ_3 (upper byte)
0005	IRQ_2 (lower byte)
0004	IRQ_2 (upper byte)
0003	IRQ_1 (lower byte)
0002	IRQ_1 (upper byte)
0001	IRQ_0 (lower byte)
0000	IRQ_0 (upper byte)

Figure 5. Program Memory Map

Z8 Standard Register File (Bank 0)

Bank 0 of the Z8 expanded register file architecture is known as the standard register file of the Z8. As shown in Figure 6, the Z8 standard register file consists of 16 groups of sixteen 8-bit registers known as Working Register (WR) groups. Working Register Group F contains various control and status registers. The lower half of Working Register Group 0 consists of I/O port registers (R0 to R7), the upper eight registers are available for use as general-purpose RAM registers. Working Register Group 1 through Group E of the standard register file are available to be used as general-purpose RAM registers. The user can use 233 bytes of general-purpose RAM registers in the standard Z8 register file (Bank 0).

Grp/Bnk	Reg	Working Register Group Function
(F0h)	r0 to 15	**Control and Status Registers**
(E0h)	r0 to 15	General-purpose RAM registers
(D0h)	r0 to 15	General-purpose RAM registers
(C0h)	r0 to 15	General-purpose RAM registers
(B0h)	r0 to 15	General-purpose RAM registers
(A0h)	r0 to 15	General-purpose RAM registers
(90h)	r0 to 15	General-purpose RAM registers
(80h)	r0 to 15	General-purpose RAM registers
(70h)	r0 to 15	General-purpose RAM registers
(60h)	r0 to 15	General-purpose RAM registers
(50h)	r0 to 15	General-purpose RAM registers
(40h)	r0 to 15	General-purpose RAM registers
(30h)	r0 to 15	General-purpose RAM registers
(20h)	r0 to 15	General-purpose RAM registers
(10h)	r0 to 15	General-purpose RAM registers
	r8 to 15	General-purpose RAM registers
(00h)	r0 to 7	**I/O Port Registers**

Figure 6. Standard Z8 Register File (Working Reg. Groups 0–F, Bank 0)

Z8 Expanded Register File

In addition to the Standard Z8 Register File (Bank 0), Expanded Register File Banks F and D of Working Register Group 0 have been implemented on the Z86D99/Z86L99. Figure 7 illustrates the Z8 Expanded Register File architecture. These two expanded register file banks of Working Register Group 0 provide a total of 32 additional RAM control and status registers. The Z86D99/Z86L99 family has implemented 21 of the 32 available registers.

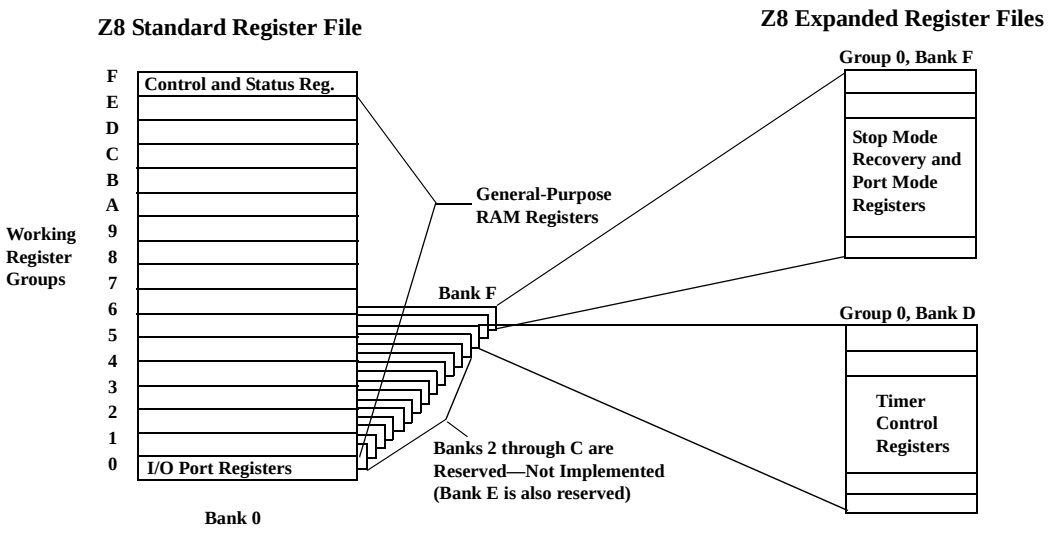


Figure 7. Z8 Expanded Register File Architecture

Clock Circuit Description

The Z8 derives its timing from on-board clock circuitry connected to pins XTAL1 and XTAL2. The clock circuitry consists of an oscillator, a divide-by-two shaping circuit, and a clock buffer. The oscillator's input is XTAL1, and the oscillator's output is XTAL2. The clock can be driven by a crystal, a ceramic resonator, LC clock, RC, or an external clock source.

Clock Control

The Z8 offers software control of the internal system clock using programming register bits in the SMR register. This register selects the clock divide value and determines the mode of STOP Mode Recovery.

The default setting is external clock divide-by-two. When bits 1 and 0 of the SMR register are set to 0, the System Clock (SCLK) and Timer Clock (TCLK) are equal to the external clock frequency divided by two.

When bit 1 of the SMR register is set to 1, then SCLK and TCLK equal the external clock frequency. Refer to Table 53 on page 85 for the maximum clock frequency.

A divide-by-16 prescaler of SCLK and TCLK allows the user to selectively reduce device power consumption during normal processor execution (under SCLK control) and/or HALT mode, where TCLK sources counter/timers and interrupt logic. Combining the divide-by-two circuitry with the divide-by-16 prescaler allows the external clock to be divided by 32.

Interrupts

The Z86D99/Z86L99 family allows up to six different interrupts, three external and three internal, from nine possible sources. The six interrupts are assigned as follows:

- Three edge-triggered external interrupts (P51, P52, and P53), two of which are shared with the two analog comparators
- One internal interrupt assigned to the T8 Timer
- One internal interrupt assigned to the T16 Timer
- One internal interrupt shared between the Low-Battery Detect flag and the T1 Timer

Table 3 presents the interrupt types, the interrupt sources, and the location of the specific interrupt vectors.

Table 3. Interrupt Types, Sources, and Vectors

Name	Source	Vector Location	Comments
IRQ_0	P52 (F/R), Comparator 2	0,1	External interrupt (P52) is triggered by either rising or falling edge; internal interrupt generated by Comparator 2 is mapped into IRQ_0
IRQ_1	P53 (F)	2,3	External interrupt (P53) is triggered by a falling edge
IRQ_2	P51 (R/F), Comparator 1	4,5	External interrupt (P51) is triggered by either a rising or falling edge; internal interrupt generated by Comparator 1 is mapped into IRQ_2
IRQ_3	T16 Timer	6,7	Internal interrupt
IRQ_4	T8 Timer	8,9	Internal interrupt
IRQ_5	LVD, T1 Timer	10,11	Internal interrupt, LVD flag is multiplexed with T1 Timer End-of-Count interrupt
Notes:	F = Falling-edge triggered; R = Rising-edge triggered. When LVD is enabled, IRQ5 is triggered only by low-voltage detection. Timer 1 does not generate an interrupt.		

These interrupts can be masked and their priorities set by using the Interrupt Mask Register (IMR) and Interrupt Priority Register (IPR) (Figure 8.) When more than one interrupt is pending, priorities are resolved by a priority encoder, controlled by the IPR.

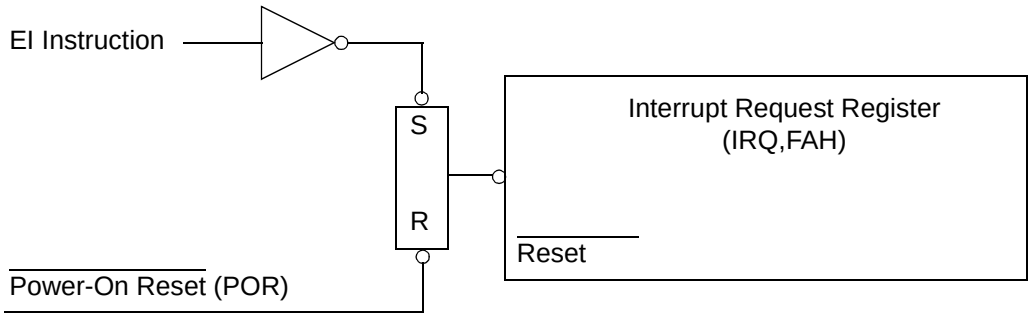


Figure 8. Interrupt Block Diagram

Interrupt requests are stored in the Interrupt Request Register (IRQ), which can also be used for polling. When an interrupt request is granted, the Z8 enters an "interrupt machine cycle" that globally disables all other interrupts, saves the program counter (the address of the next instruction to be executed) and status flags, and finally branches to the vector location for the interrupt granted. It is only at this point that control passes to the interrupt service routine for the specific interrupt.

All six interrupts can be globally disabled by resetting the master Interrupt Enable (bit 7 of the IMR) with a Disable Interrupts (DI) instruction. Interrupts are globally enabled by setting the same bit with an Enable Interrupts (EI) instruction.

Descriptions of three interrupt control registers—the Interrupt Request Register, the Interrupt Mask Register, and the Interrupt Priority Register—are provided in "Register Summary" on page 52. The Z8 family supports both vectored and polled interrupt handling.

External Interrupt Sources

External sources involve interrupt request lines P51, P52, and P53 (IRQ_2, IRQ_0, and IRQ_1, respectively.) IRQ_0, IRQ_1, and IRQ_2 are generated by a transition on the corresponding port pin. As shown in Figure 9, when the appropriate port pin (P51, P52, or P53) transitions, the first flip-flop is set. The next two flip-flops synchronize the request to the internal clock and delay it by two internal clock periods. The output of the most recent flip-flop (IRQ_0, IRQ_1, or IRQ_2) sets the corresponding Interrupt Request Register bit.

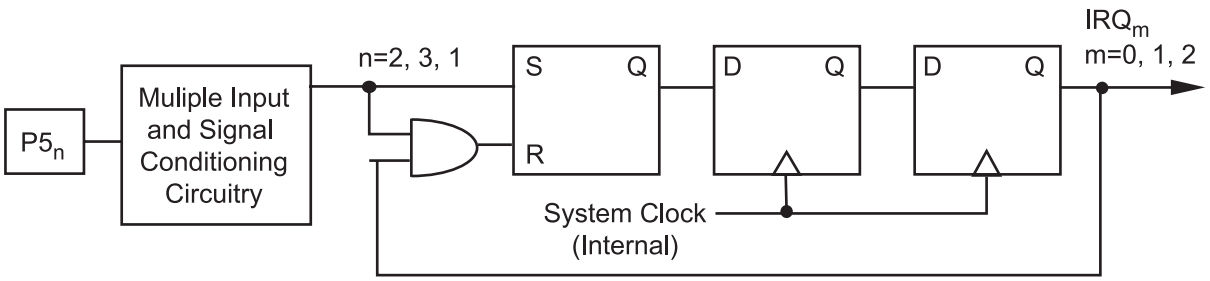


Figure 9. External Interrupt Sources IRQ_0–IRQ_2 Block Diagram

The programming bits for the Interrupt Edge Select function are located in the IRQ register, bits 6 and 7. The configuration of these bits and the resulting interrupt edge is shown in Table 4.

Table 4. Interrupt Edge Select for External Interrupts

Interrupt Request Register		Interrupt Edge	
Bit 7	Bit 6	IRQ_2 (P51)	IRQ_0 (P52)
0	0	Falling	Falling
0	1	Falling	Rising
1	0	Rising	Falling
1	1	Rising/Falling	Rising/Falling

Note: Although interrupts are edge triggered, minimum interrupt request Low and High times must be observed for proper operation. See “Electrical Characteristics” on page 85 for exact timing requirements (T_WIL, T_WIH) on external interrupt requests.

Internal Interrupt Sources

Internal sources are ORed with the external sources, so that either an internal or external source can trigger the interrupt.

Interrupt Request Register Logic and Timing

Figure 10 shows the logic diagram for the Interrupt Request Register. The leading edge of an interrupt request sets the first flip-flop. It remains set until the interrupt requests are sampled.

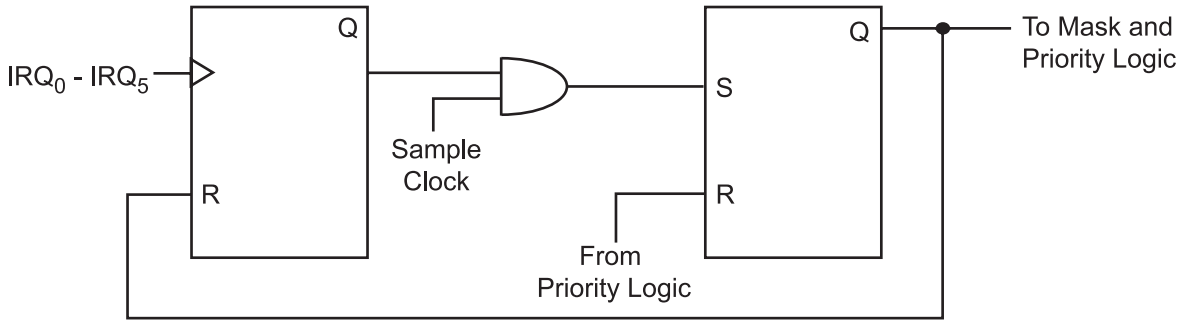


Figure 10. IRQ Logic

Internal interrupt requests are sampled during the most recent clock cycle before an Op Code fetch (see Figure 11.) External interrupt requests are sampled two internal clocks earlier than internal interrupt requests because of the synchronizing flip-flops shown in Figure 9.

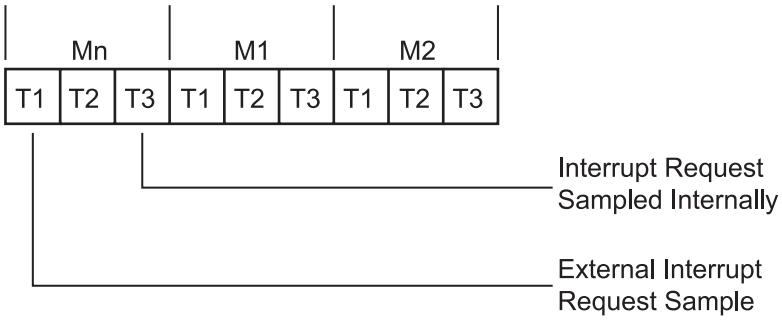


Figure 11. Interrupt Request Timing

At sample time, the interrupt request is transferred to the second flip-flop shown in Figure 10, which drives the interrupt mask and priority logic. When an interrupt cycle occurs, this flip-flop is reset only for the highest priority level that is enabled.

The user has direct access to the second flip-flop by reading and writing to the IRQ. The IRQ is read by specifying it as the source register of an instruction, and the IRQ is written by specifying it as the destination register.

Interrupt Initialization

After RESET, all interrupts are disabled and must be re-initialized before vectored or polled interrupt processing can begin. The Interrupt Priority Register, Interrupt Mask Register, and Interrupt Request Register must be initialized, in that order, to

start the interrupt process. However, the IPR does not have to be initialized for polled processing.

Interrupts must be globally enabled using the EI instruction. Setting bit 7 of the IMR is not sufficient. Subsequent to this EI instruction, interrupts can be enabled either by IMR manipulation or by use of the EI instruction, with equivalent effects.

Additionally, interrupts must be disabled by executing a DI instruction before the IPRs or IMRs can be modified. Interrupts can then be enabled by executing an EI instruction.

IRQ Software Interrupt Generation

IRQ can be used to generate software interrupts by specifying IRQ as the destination of any instruction referencing the Z8 Standard Register File. These Software Interrupts (SWIs) are controlled in the same manner as hardware-generated requests (the IPR and the IMR control the priority and enabling of each SWI level).

To generate a SWI, the request bit in the IRQ is set as follows:

```
OR    IRQ, #NUMBER
```

where the immediate data, *NUMBER*, has a 1 in the bit position corresponding to the appropriate level of the SWI.

For example, for an SWI on IRQ5, *NUMBER* has a 1 in bit 5. With this instruction, if the interrupt system is globally enabled, IRQ5 is enabled, and there are no higher priority pending requests, control is transferred to the service routine pointed to by the IRQ5 vector.

Reset Conditions

A system reset overrides all other operating conditions and puts the Z8 into a known state. The control and status registers are reset to their default conditions after a power-on reset (POR) or a Watch-Dog Timer (WDT) time-out while in RUN mode. The control and status registers are not reset to their default conditions after Stop Mode Recovery (SMR) while in HALT or STOP mode.

General-purpose registers are undefined after the device is powered up. Resetting the Z8 does not affect the contents of the general-purpose registers. The registers keep their most recent value after any reset, as long as the reset occurs in the specified V_{CC} operating range. Registers do not keep their most recent state from a V_{LV} reset, if V_{CC} drops below V_{RAM} (see Table 54 on page 87).

Following a reset (see Table 5), the first routine executed must be one that initializes the control registers to the required system configuration.

Table 5. Control and Status Register Reset Conditions

Register Function	Address Grp/Bnk	Register	Symbol	R/W	Reset Value 7	6	5	4	3	2	1	0
Register Pointer	F0h	r13 (R253)	RP	R/W	0	0	0	0	0	0	0	0
Stack Pointer	F0h	r15 (R255)	SP	R/W	X	X	X	X	X	X	X	X
Program Control Flags	F0h	r12 (R252)	Flags	R/W	X	X	X	X	X	X	X	X
Low Battery Detect	0Dh	r12	LB	R/W	1	1	1	1	1	X	0	0
ADC Control	0Fh	r8	ADCCTRL	R/W	0	0	0	0	0	0	0	0
ADC Data	00h	r7 (R7)	ADCDATA	R	0	0	0	0	0	0	0	0
Interrupt Mask	F0h	r11 (R251)	IMR	R/W	0	0	0	0	0	0	0	0
Interrupt Priority	F0h	r9 (R249)	IPR	W	0	0	0	0	0	0	0	0
Interrupt Request	F0h	r10 (R250)	IRQ	R/W	0	0	0	0	0	0	0	0
Port Configuration (A)	0Fh	r0	P456CON	R/W	0	0	0	0	0	1	1	1
Port Configuration (B)	F0h	r7 (R247)	P3M	W	1	1	1	1	1	1	1	1
Port 2 Data	00h	r2 (R2)	P2	R/W	X	X	X	X	X	X	X	X
Port 2 Mode	F0h	r6 (R246)	P2M	W	1	1	1	1	1	1	1	1
Port 4 Data	00h	r4 (R4)	P4	R/W	X	X	X	X	X	X	X	X
Port 4 Mode	0Fh	r2	P4M	R/W	1	1	1	1	1**	1	1	1
Port 5 Data	00h	r5 (R5)	P5	R/W	X	X	X	X	X	X	X	X
Port 5 Mode	0Fh	r4	P5M	R/W	1	1	1	1	1	1	1	1
Port 6 Data	00h	r6 (R6)	P6	R/W	X	X	X	X	X	X	X	X
Port 6 Mode	0Fh	r6	P6M	R/W	1	1	1	1	1	1	1	1
T1 Timer Data	F0h	r2 (R242)	T1	R/W	0	0	0	0	0	0	0	0
T1 Timer Mode	F0h	r1 (R241)	TMR	R/W	0	0	0	0	0	0	1	1
T1 Timer Prescale	F0h	r3 (R243)	PRE1	R/W	0	0	0	0	0	0	0	0
T8/T16 Control (A)	0Dh	r1	CTR1	R/W	0	0	0*	0*	0	0	0	0
T8/T16 Control (B)	0Dh	r3	CTR3	R/W	0	0	0*	X	X	X	X	X
T8 Timer Control	0Dh	r0	CTR0	R/W	0	0	0*	0*	0*	0*	0*	0
T8 High Capture	0Dh	r11	HI8†	RW	0	0	0	0	0	0	0	0
T8 Low Capture	0Dh	r10	LO8†	R/W	0	0	0	0	0	0	0	0
T8 High Load	0Dh	r5	TC8H†	R/W	0	0	0	0	0	0	0	0
T8 Low Load	0Dh	r4	TC8L†	R/W	0	0	0	0	0	0	0	0
T16 Timer Control	0Dh	r2	CTR2	R/W	0	0	0	0	0	0	0	0
T16 High Capture	0Dh	r9	HI16†	R/W	0	0	0	0	0	0	0	0
T16 Low Capture	0Dh	r8	LO16†	R/W	0	0	0	0	0	0	0	0
T16 High Load	0Dh	r7	TC16H†	R/W	0	0	0	0	0	0	0	0
T16 Low Load	0Dh	r6	TC16L†	R/W	0	0	0	0	0	0	0	0

Table 5. Control and Status Register Reset Conditions (Continued)

Register Function	Address Grp/Bnk Register	Symbol	R/W	Reset Value 7	6	5	4	3	2	1	0
Stop Mode Recovery	0Fh r11	SMR	R/W	0	0	1	0	0	0	0	0
Port 2 SMR Source	0Fh r1	P2SMR	R/W	0	0	0	0	0	0	0	0
Port 5 SMR Source	0Fh r5	P5SMR	R/W	0	0	0	0	0	0	0	0

Notes:
†This register is not reset following Stop Mode Recovery (SMR).
*This bit is not reset following SMR.
X means this bit is undefined at POR and is not reset following SMR.
**In OTP, the default for P43 is open-drain output at power up; you need to initialize the P43 data. In the mask part, the P43 output is disabled until it is configured as output.

Power-On Reset

A POR (cold start) always resets the Z8 control and status registers to their default conditions. A POR sets bit 7 of the Stop Mode Recovery register to 0 to indicate that a cold start has occurred.

A timer circuit clocked by a dedicated on-board RC oscillator is used for the Power-On Reset Timer (TPOR) function. The POR time is specified as T_{POR}. T_{POR} time allows V_{CC} and the oscillator circuit to stabilize before instruction execution begins.

The POR delay timer circuit is a one-shot timer triggered by one of three conditions:

- Power Fail to Power OK status including recovery from Low Voltage ($V_{LV)}$ Standby mode
- STOP-Mode Recovery (when bit 5 of the SMR register = 1)
- WDT time-out

Under normal operating conditions, a stop mode recovery event always triggers the POR delay timer. This delay is necessary to allow the external oscillator time to stabilize. When using an RC or LC oscillator (with a low Q factor), the shorter wake-up time means the delay can be eliminated.

Bit 5 of the SMR register selects whether the POR timer delay is used after Stop-Mode Recovery or is bypassed. If bit 5 =1, then the POR timer delay is used. If bit 5 = 0, then the POR timer delay is bypassed. In this case, the SMR source must be held in the recovery state for 5 TpC to pass the Reset signal internally.

Watch-Dog Timer (WDT)

The WDT is a retriggerable one-shot timer that resets the Z8 if it reaches its terminal count. When operating in the RUN modes, a WDT reset is functionally

equivalent to a hardware POR reset. If the mask option of the permanently enabled watch-dog timer is selected, it runs when power up. If the option is not selected, the WDT is initially enabled by executing the WDT instruction and refreshed on subsequent executions of the WDT instruction.

The WDT instruction does not affect the Zero (Z), Sign (S), and Overflow (V) flags. Permanently enabled WDTs are always enabled, and the WDT instruction is used to refresh it. The WDT cannot be disabled after it has been initially enabled. The WDT is off during both HALT and STOP modes.

The WDT circuit is driven by an on-board RC oscillator. The time-out period for the WDT is fixed to a typical value (see Table 57 on page 90).

Power Management

In addition to the standard RUN mode, the Z8 supports three power-down modes to minimize device current consumption. The following three modes are supported:

- HALT
- STOP
- Low-Voltage Standby

Table 6 shows the status of the internal CPU clock (SCLK), the internal Timer clock (TCLK), the external oscillator, and the Watch-Dog Timer during the RUN mode and three low-power modes.

Table 6. Clock Status in Operating Modes

Operating Mode	SCLK	TCLK	External OSC	WDT*
RUN	On	On	On	On
HALT	Off	On	On	Off
STOP	Off	Off	Off	Off
Low-Voltage Standby	Off	Off	Off	Off
Note: * When WDT is enabled by the mask option bit				

Using the Power-Down Modes

In order to enter HALT or STOP mode, it is necessary to first flush the instruction pipeline to avoid suspending execution in mid-instruction. You can flush the

instruction pipeline by executing a NOP (Op Code = FFh) immediately before the appropriate sleep instruction. For example:

Mnemonic	Comment	Op Code
NOP	; clear the pipeline	FFh
STOP	; enter STOP mode	6Fh

or

Mnemonic	Comment	Op Code
NOP	; clear the pipeline	FFh
HALT	; enter HALT mode	7Fh

HALT

HALT mode suspends instruction execution and turns off the internal CPU clock (SCLK). The on-chip oscillator circuit remains active, so the internal Timer clock (TCLK) continues to run and is applied to the counter/timers and interrupt logic.

An interrupt request, either internally or externally generated, must be executed (enabled) to exit HALT mode. After the interrupt service routine, the program continues from the instruction immediately following the HALT.

The HALT mode can also be exited by a POR. In this case, the program execution restarts at the reset address 000Ch.

STOP

STOP mode provides the lowest possible device standby current. This instruction turns off both the internal CPU clock (SCLK) and internal Timer clock (TCLK) and reduces the standby current to the minimum.

The STOP mode is terminated by a POR or SMR source. Terminating the STOP mode causes the processor to restart the application program at address 000Ch.

Note: When the STOP instruction is executed, the microcontroller goes into the STOP mode despite any state/change of the state of the port. The ports need to be checked immediately before the NOP and STOP instructions to ensure the right input logic before waiting for the change of the ports.

Stop Mode Recovery Sources

Exiting STOP mode using an SMR source is greatly simplified in the Z86D99/Z86L99 family. The Z86D99/Z86L99 family of products allows 16 individual I/O

pins (Ports 2 and 5) to be used as stop-mode recovery sources. The STOP mode is exited when one of these SMR sources is toggled. A transition from either low to high or high to low on any pin of Port 2 or Port 5 if the pin is identified as an SMR source will effect an SMR.

There are three registers that control STOP mode recovery:

- Stop Mode Recovery
- Port 2 Stop Mode Recovery (P2SMR)
- Port 5 Stop Mode Recovery (P5SMR)

The functions and applications of these registers are explained in “Stop-Mode Recovery Control Registers” on page 82.

Low-Voltage Standby

An on-chip voltage comparator checks that the V_{CC} level is at the required level for correct operation of the Z8. When V_{CC} falls below the low-voltage trip voltage (V_{LV}), reset is globally driven, and then the device is put in a low-current standby mode with the external oscillator stopped. If the V_{CC} remains above V_{RAM}, the RAM content is preserved.

When the power level rises above the V_{LV} level, the device performs a POR and functions normally.

The minimum operating voltage varies with temperature and operating frequency, while V_{LV} varies with temperature only.

I/O Ports

The Z86D99/Z86L99 family has up to 32 lines dedicated to input and output in the 40-pin configuration. These lines are grouped into four 8-bit ports known as Port 2, Port 4, Port 5, and Port 6. All four ports are bit programmable as either inputs or outputs with the exception of P52, P53, and P43. P52 and P53 are input only as they are used in OTP programming. P43 is the controlled current output and is therefore output only.

All ports have push-pull CMOS outputs. In addition, the push-pull outputs can be turned off for open-drain operation using the P456CON register.

Internal resistive pull-up transistors are available as a user-defined OTP/mask option on all ports. For Ports 4, 5, and 6, the pull-ups are nibble selectable. For Port 2, the pull-up option applies to all eight I/O lines.

Note: Internal pull-ups are disabled on any given pin or group of port pins when those pins are programmed as outputs.

Mode Registers

Each port has an associated Mode Register that determines the port's functions and allows dynamic change in port functions during program execution. Port and Mode Registers are mapped into the Standard Register File. Because of their close association, Port and Mode Registers are treated like any other general-purpose register. There are no special instructions for port manipulation. Any instruction that addresses a register can address the ports. Data can be directly accessed in the Port Register, with no extra moves.

Input and Output Registers

Each of the four ports (Ports 2, 4, 5, and 6) has an input register, an output register, and associated buffer and control logic. Because there are separate input and output registers associated with each port, writing bits defined as inputs store the data in the output register. This data cannot be read as long as the bits are defined as inputs. However, if the bits are reconfigured as output, the data stored in the output register is reflected on the output pins and can then be read. This mechanism allows the user to initialize the outputs before driving their loads.

Because port inputs are asynchronous to the Z8 internal clock, a READ operation could occur during an input transition. In this case, the logic level might be uncertain (somewhere between a logic 1 and 0).

General Port I/O

The eight I/O lines of each port (except P43, P52, and P53) can be configured under software control to be either input or output, independently. Bits programmed as outputs can be globally programmed as either push-pull or open-drain. See Figure 12.

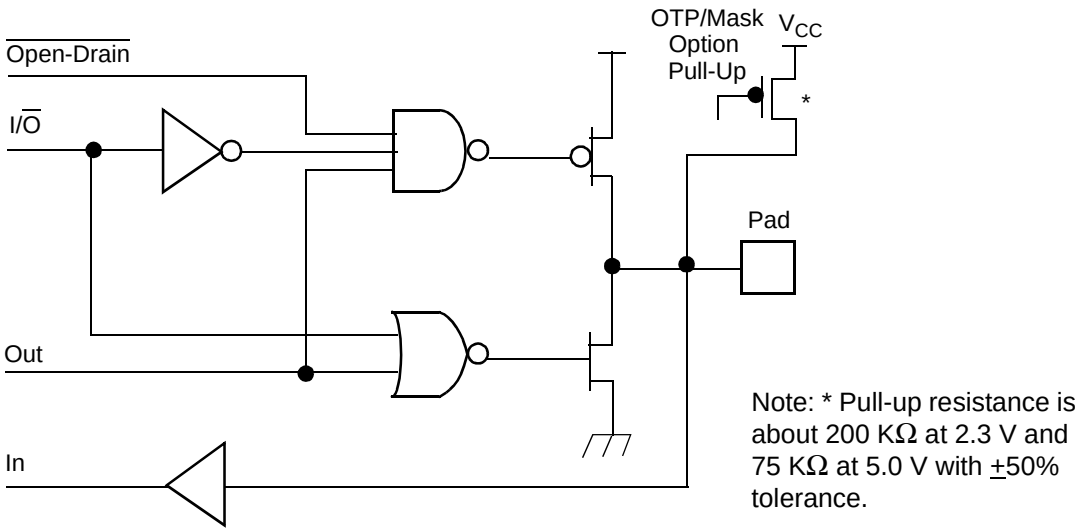


Figure 12. General Input/Output Pin

Read/Write Operations

The ports are accessed as general-purpose registers. Port registers are written by specifying the port register as an instruction's destination register. Writing to a port causes data to be stored in the output register of the port, and reflected externally on any bit configured as an output.

Ports are read by specifying the port register as the source register of an instruction. When an output bit is read, data on the external pin is returned. Under normal loading conditions, returning data on the external pin is equivalent to reading the output register. However, if a bit is defined as an open-drain output, the data returned is the value forced on the output pin by the external system. This value might not be the same as the data in the output register. Reading input bits also returns data on the external pins.

Special Functions

Table 7 defines the special functions of Ports 4 and 5.

Table 7. Special Port Pin Functions

Function	Pin	Signal	Configuration Register
Analog Comparator Inputs	P51	CIN1	P456CON
	P52	CIN2	P456CON
Analog Comparator References	P50	CREF1	
	P53	CREF2	
Analog Comparator Outputs	P54	COUT1	
	P55	COUT2	
ADC Channels	P44	ADC0	ADCCTRL
	P45	ADC1	ADCCTRL
	P46	ADC2	ADCCTRL
	P47	ADC3	ADCCTRL
External Interrupts	P52	IRQ_0	IMR and IRQ
	P53	IRQ_1	IMR and IRQ
	P51	IRQ_2	IMR and IRQ
T_{IN} External Clock Input	P52	T_{IN}	TMR and PRE1
Capture Timer Input	P51	Demodulator_Input	CTR1
T1 Timer Output	P56	T1OUT	TMR
T8 Output	P40	P40_Out	CTR0
T16 Output	P41	P41_Out	CTR2
Combined T8/T16 Output Controlled Current Output	P43	P43_Out	CTR1
ZiLOG Test Mode	P41	DSn Enable	P456CON
	P42	ASn Enable	P456CON

Peripherals

Analog Comparators

The Z86D99/Z86L99 family includes two independent on-chip general-purpose analog comparators as shown in Figure 13. The comparators are multiplexed with a digital input signal by the P456CON register. They can also be used to generate interrupts IRQ0 and IRQ2. The comparators are turned off in STOP mode.

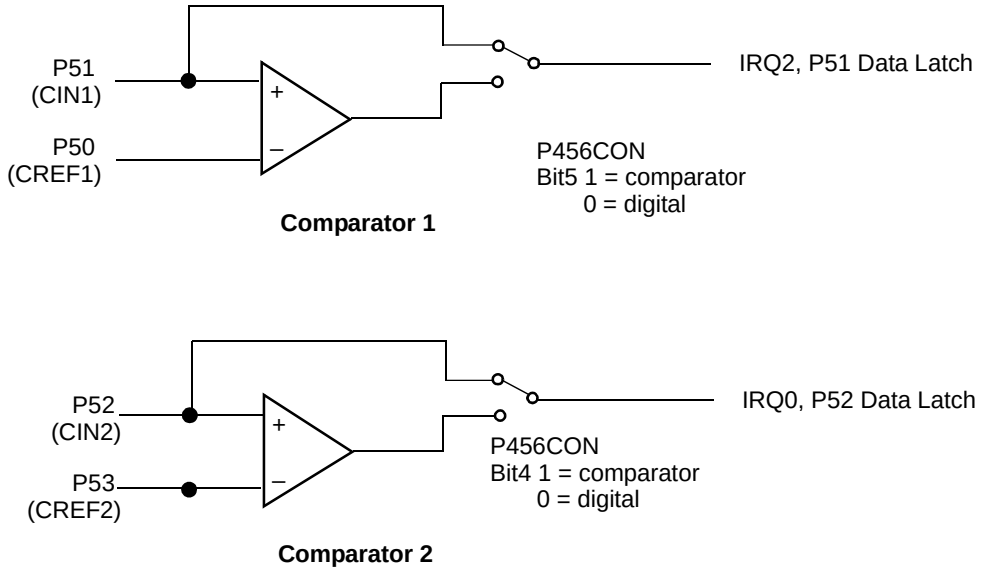


Figure 13. Analog Comparators

Analog/Digital Converter (ADC)

The Z86D99/Z86L99 family incorporates an 8-bit ADC that uses a sigma delta architecture (Figure 14) comprised of a modulator and a digital filter. The input is selected (bit 3,2 from ADCCTRL) with an analog mux from 4 (P47–P44) pins that can be configured as analog inputs (bit 7–4 from ADCCTRL).

Note: Whenever an input pin has an analog value, the digital input buffer has to be disabled in order to reduce the current through the device.

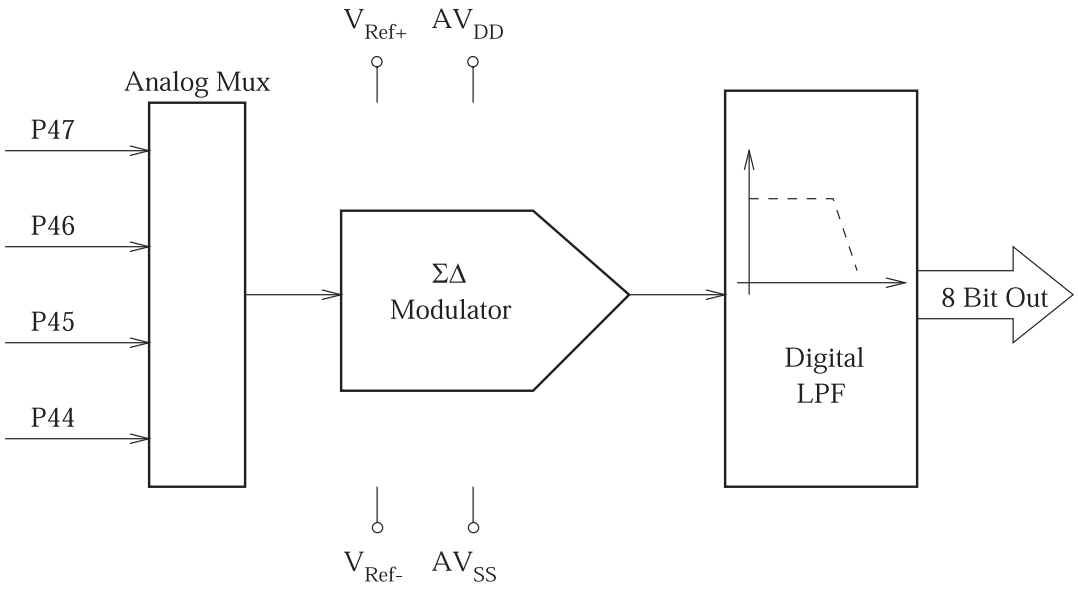


Figure 14. ADC Block Diagram

The low-pass filter transfer function is presented in Figure 15 with the –3-dB frequency given by the formula:

$$f_{3db} = 0.0021 \cdot f_{ADC}$$

where f_{ADC} is the sampling frequency of the modulator.

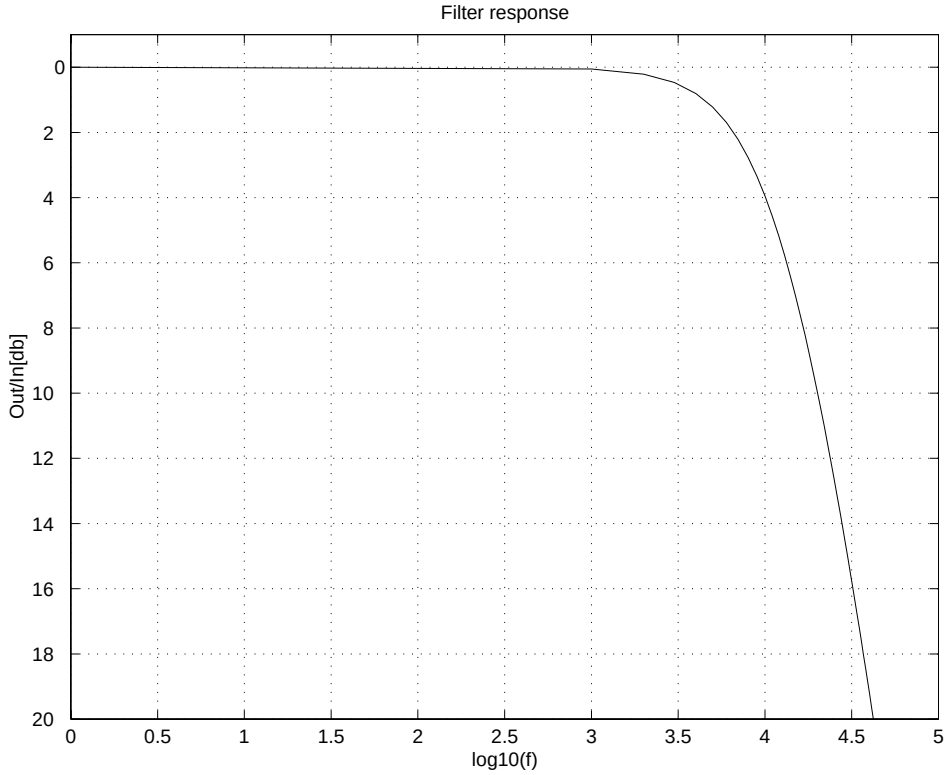


Figure 15. Low-Pass Filter (with 8-MHz Crystal)

The sampling frequency of the modulator f_{ADC} can be selected between f_{SCLK} and $f_{SCLK}/2$ (bit1 from ADCCTRL). Reducing the clock frequency lowers the power dissipated in the ADC block.

The ADC can be enabled or disabled. When enabled, the ΣΔ converter tracks the input voltage. When switching between the channels (step response), the required time to reach the final value is given by the time constant of the low-pass filter:

$$T_{delay} = \frac{2}{f_{3db}} = \frac{2}{0.0021 f_{ADC}} = \frac{952}{f_{ADC}}$$

When available, the reference for the ADC is set externally with the V_{ref+} and V_{ref-} pins. The output code represents the following ratio:

$$D_{out} = \frac{V_{in} - V_{Ref-}}{V_{Ref+} - V_{Ref-}} \times 256$$

Though the ADC functions for smaller input voltage range (V_{Ref+}–V_{Ref-}), the noise and offsets remain constant over the specified electrical range. The errors of the converter increase due to small input signals.

For fast access to the output of the ADC, the current data is available in the ADC result register (r8, bank00).

To reduce the interference between the digital part and the analog part, separate AV_{SS} and AV_{DD} pins are available on the packages where the ADC can be used.

Note: In the smaller packages, which do not support the ADC, the user must keep the converter *not active* in order to not have power dissipated in the ADC block. By default, ADC is off.

Active Glitch Filter

The Z86D99/Z86L99 family incorporates an active power/glitch filter that can be used to improve the quality of the power supply when the device is operating in noisy environments. The chips use three separate power buses:

- pad ring power bus (all the output drivers plus the crystal/RC oscillator) called $V_{DD\_padring}$
- core power bus (all digital circuitry) called V_{DD_CORE}
- analog power bus (all analog circuitry) called AV_{DD}

Depending on the pin availability, one or more of the power buses are connected together.

The active power filter can be used in the packages that have the V_{DD} separate. Figure 16 shows the internal schematic.

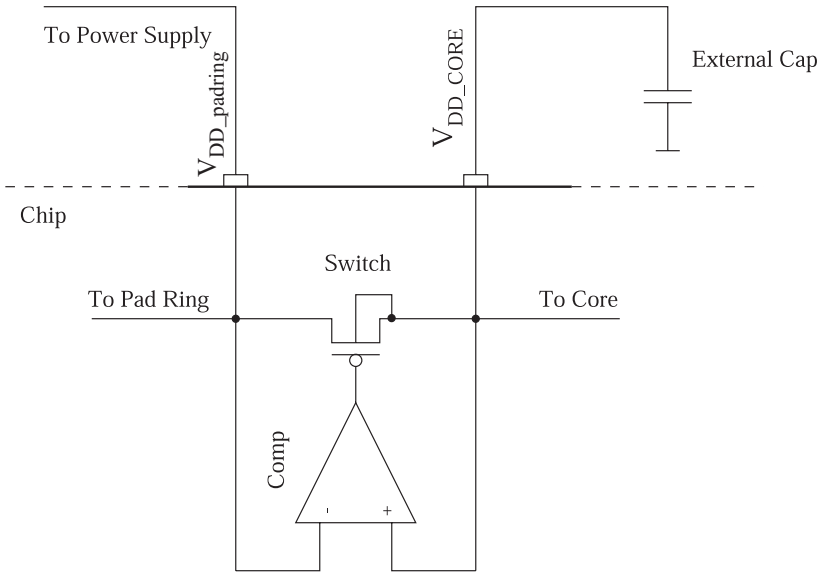


Figure 16. Active Glitch/Power Filter

When the internal power/glitch filter is not used, both $V_{DD\_padring}$ and V_{DD_CORE} must be connected together externally to the power supply.

When the internal circuitry is used, the $V_{DD\_padring}$ has to be connected to the power supply and the V_{DD_CORE} has to be connected to an external energy storage capacitor (1–10 µF range). The core is connected only to this capacitor during power supply glitches.

Table 8 describes the active glitch/filter specifications.

Table 8. Active Glitch/Filter Specifications (Preliminary)

Parameter	Max	Min	Condition
Diff. stage gain		75 dB	
Diff. stage bandwidth		15 MHz	
Rise time	255 ns		50 mV pulse
Fall time	214 ns		50 mV pulse
R_{dson}	10 Ω		

On the wafer level, all three power buses are available. Depending on the number of pins of the package, one or more power buses are connected together.

The active glitch/power filter effectively increases the noise immunity for battery-operated designs where the controller is driving high current loads (for example, IR LED).

Controlled Current Output

P43 is an open-drain output-only pin on the Z86D990/D991, but it can be configured as output or Tristate High Impedance on the Z86L990/L991. To function properly, Bit 3 of P4M must be set to zero to configure the pin as an open-drain output. For the Z86L990/L991 after reset, P43 defaults to Tristate High Impedance while the Z86D990/D991 P43 is always configured as output. The data at Port 4 must be initialized as it is undefined at power-on reset.

The current output is a controlled current source that is controlled by the output of the value of P43 (see Table 9). P43 *cannot* be configured as input, and if P43 is read, P43 always returns the state of the output value (1 for no sink and 0 for sink).

P43 uses internal current reference and will draw current if it outputs a low logic even without external connection. This applies to both Run mode and Stop mode.

Table 9. Current Sink Pad P43 Specifications (Preliminary)

Parameter	Min	Max	Conditions
Rise time		0.4 µ	LED load
Fall time		0.02 µ	LED load
V_{outmin}		0.54 V	@27C
Comparator response		0.2 µ	
Regulated current	80 mA	120 mA	
Internal resistance		80 Ω	

The pad driver can function in two modes:

- controlled current output, when the voltage on the pad is over a minimum value

$$V_{pad} > V_{outmin}$$

- resistive pull down when the driver cannot regulate the current; in this mode, the gate of the NMOS pull down is raised to the power rail.

The I-V characteristics of the pad are presented in Figure 17.

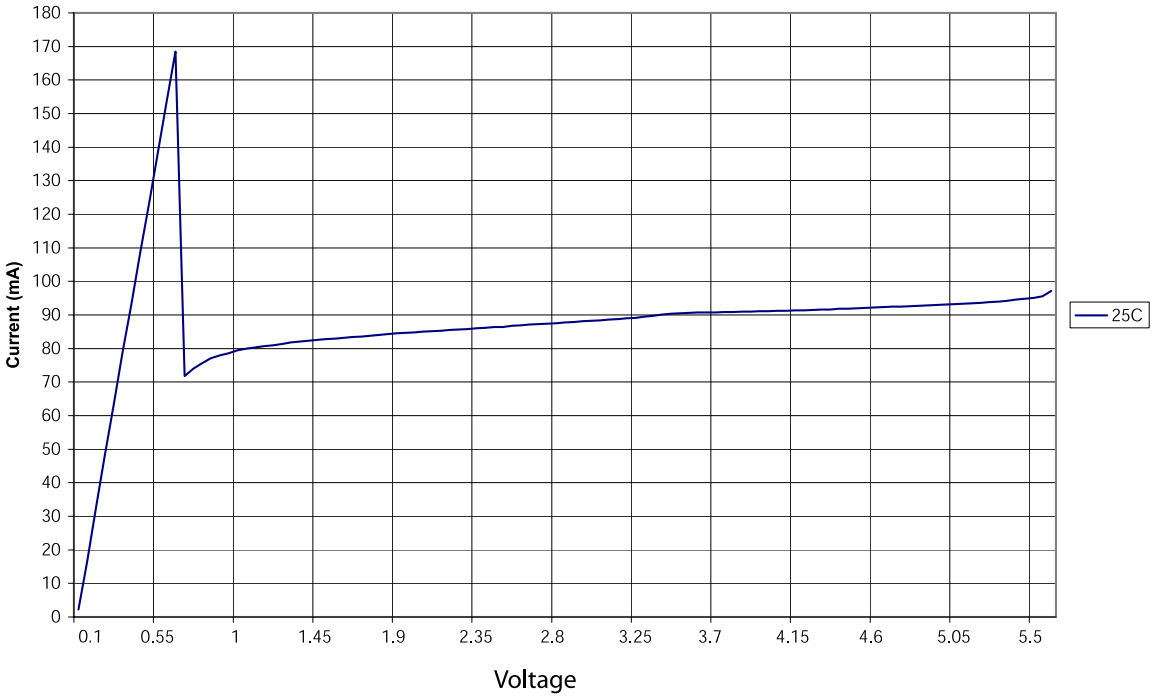


Figure 17. I-V Characteristics for the Current Sink Pad P43

The CPU reads the mode of the pad driver by reading bit number 2 from the LB register. This bit is the output of a Set-Reset flip-flop that sets whenever the voltage on the pad is lower than V_{outmin} and is reset by a CPU write to the respective register.

T1 Timer

The Z86D99/Z86L99 family provides one general-purpose 8-bit counter/timer, T_1, driven by its own 6-bit prescaler, PRE_1. The T_1 counter/timer is independent of the processor instruction sequence, which relieves software from time-critical operations such as interval timing and event counting.

The T_1 counter/timer operates in either single-pass or continuous mode. At the end-of-count, counting either stops or the initial value is reloaded and counting continues. Under software control, new values are loaded immediately or when the end-of-count is reached. Software also controls the counting mode, how the counter/timer is started or stopped, and the counter/timer's use of I/O lines. Both the counter and prescaler registers can be altered while the counter/timer is running.

Counter/timer 1 is driven by a timer clock generated by dividing the internal clock by four. The divide-by-four stage, the 6-bit prescaler, and the 8-bit counter/timer form a synchronous 16-bit divide chain. Counter/timer T_1 can also be driven by an external input (T_{IN}) using Port P52. Port $P5_6$ can serve as a timer output (T_{OUT}) through which T_1 or the internal clock can be output. The timer output toggles at the end-of-count. Figure 18 is a block diagram of the counter/timer.

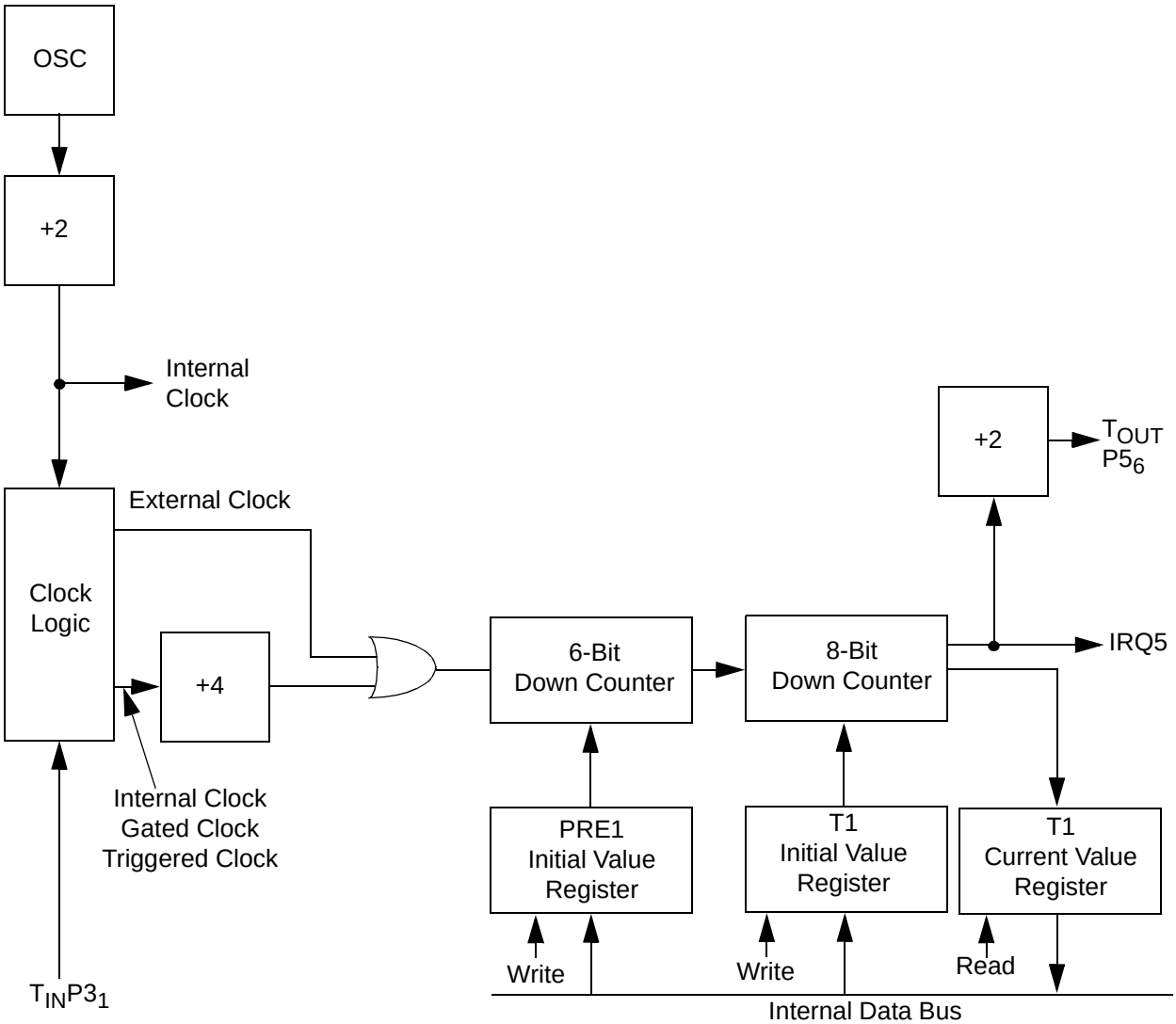


Figure 18. T_1 Counter/Timer Block Diagram

The counter/timer, prescaler, and associated mode registers are mapped into the register file as shown in Figure 19. The software uses the counter/timer as a general-purpose register, which eliminates the need for special instructions.

DEC		Hex identifiers	
243	T1 prescaler	F3	PRE1
242	Timer/counter 1	F2	T1
241	Timer mode	F1	TMR

Figure 19. Register File

Prescaler and Counter/Timer

The prescaler PRE_1 (F3h) consists of an 8-bit register and a 6-bit down-counter as shown in Figure 18 on page 35. The prescaler register is a read-write register. Figure 20 shows the prescaler register.

R243 PRE1
Prescaler 1 Register
(F3h; Read/Write)

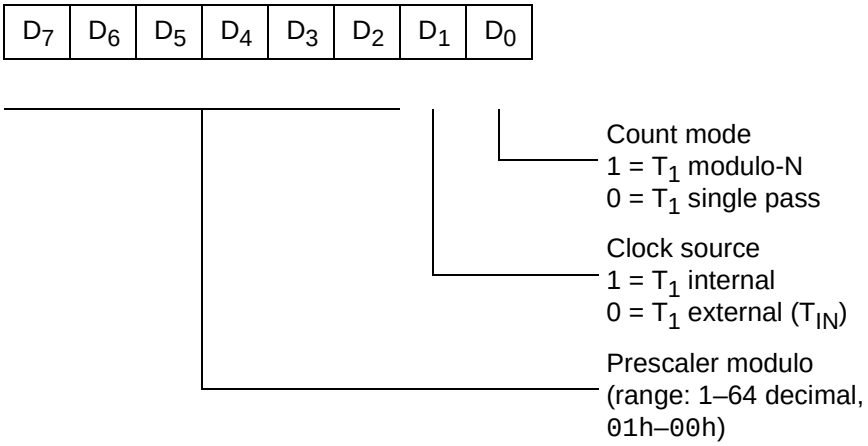


Figure 20. Prescaler 1 Register

The six most significant bits (D_2–D_7) of PRE_1 hold the prescaler count modulo, a value from 11 to 64 decimal. The prescaler register also contains control bits that specify T_1 counting modes. These bits also indicate whether the clock source for T_1 is internal or external.

The counter/timer T_1 (F2h) consists of an 8-bit down-counter, a write-only register that holds the initial count value, and a read-only register that holds the current count value (see Figure 18 on page 35). The initial value can range from 1 to 256 decimal (01h, 02h, ..., 00h). Figure 21 illustrates the counter/timer register.

R242 T1
Counter/Timer 1 Register
(F2h; Read/Write)

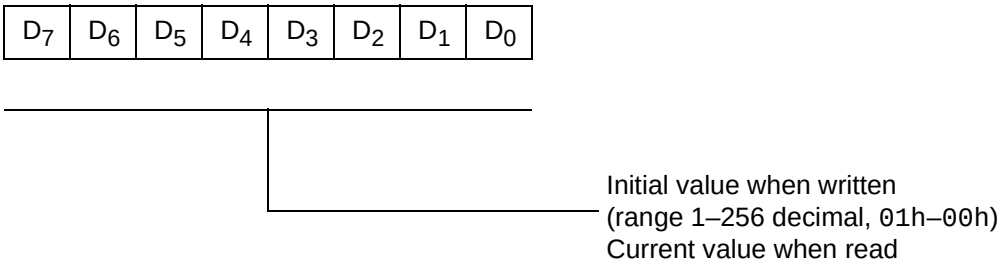


Figure 21. Counter/Timer 1 Register

Counter/Timer Operation

Under software control, T_1 is started and stopped using the Timer Mode register (F1h) bits D_2–D_3: a Load bit and an Enable Count bit. See Figure 22.

R241 TMR
Timer Mode Register
(F1h; Read/Write)

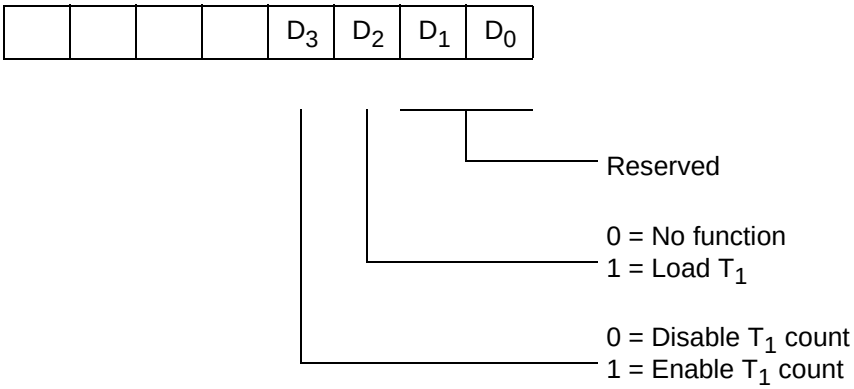


Figure 22. Timer Mode Register

Load and Enable Count Bits

Setting the Load bit D_2 to 1 transfers the initial values in the prescaler and the counter/timer registers into their respective down-counters. The next internal clock resets bit D_2 to 0, readying the Load bit for the next load operation. The initial values can be loaded into the down-counters at any time. If the counter/timer is running, the counter/timer continues to run and starts the count over with the initial value. Therefore, the Load bit actually functions as a software re-trigger.

The T_1 counter/timer remains at rest as long as the Enable Count bit D_3 is 0. To enable counting, the Enable Count bit D_3 must be set to 1. Counting actually starts when the Enable Count bit is written by an instruction. The first decrement occurs four internal clock periods after the Enable Count bit has been set.

The Load and Enable Count bits can be set at the same time. For example, using the instruction `OR TMR #%0C` sets both D_2 and D_3 of TMR to 1. The initial values of PRE_1 and T_1 are loaded into their respective counters, and the count is started after the M2T2 machine state after the operand is fetched as shown in Figure 23.

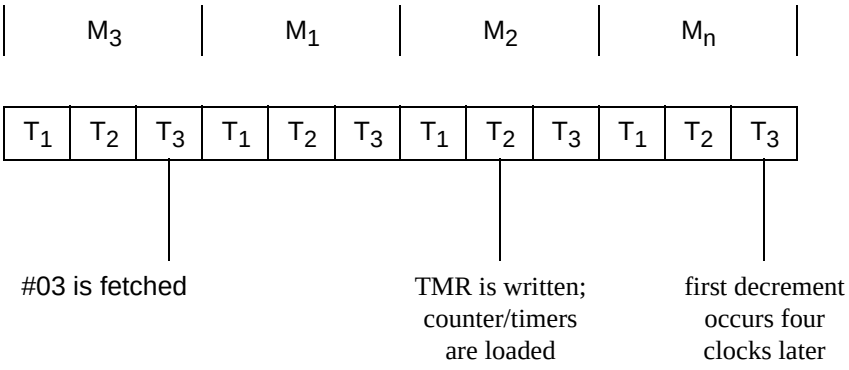


Figure 23. Starting the Count

Prescaler Operations

During counting, the programmed clock source drives the prescaler 6-bit counter. The counter is counted down from the value specified by bits D_2–D_7 of the corresponding prescaler register, PRE_0 or PRE_1 (Figure 24). When the prescaler counter reaches its end-of-count, the initial value is reloaded and counting continues. The prescaler never actually reaches zero. For example, if the prescaler is set to divide by three, the count sequence is as follows:

3-2-1-3-2-1-3-2...

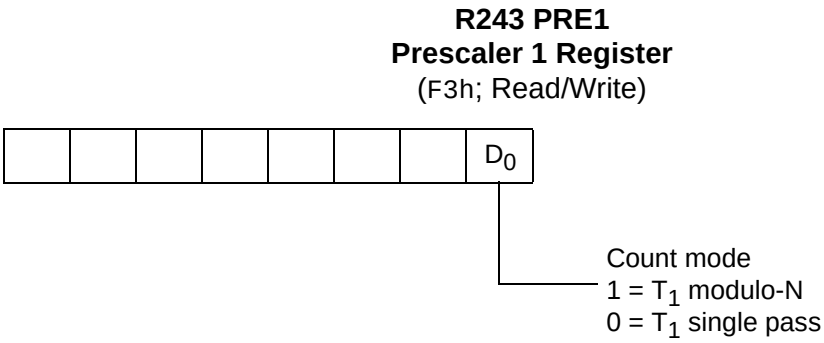


Figure 24. Counting Modes

When the PRE_1 register is loaded with 000000 in the six most significant bits, the prescaler divides by 64. If that number is 000001, the prescaler does not divide and passes its clock on to T_1.

Each time the prescaler reaches its end-of-count, a carry is generated, which allows the counter/timer to decrement by one on the next timer clock input. When T_1 and PRE_1 both reach their end-of-count, an interrupt request is generated—IRQ_5 for T_1. Depending on the counting mode selected, the counter/timer either comes to rest with its value at 00h (single-pass mode), or the initial value is automatically reloaded and counting continues (continuous mode). In single-pass mode, the prescaler still continues to decrement when the timer T_1 has reached its end-of-count. The prescaler always starts from its programmed value upon restarting the counter.

The counting modes are controlled by bit D_0 of PRE_1, with D_0 cleared to 0 for single-pass counting mode or set to 1 for continuous mode.

The counter/timer can be stopped at any time by setting the Enable Count bit to 0 and restarted by setting the Enable Count bit back to 1. The T_1 counter/timer continues its count value at the time it was stopped. The current value in the T_1 counter/timer can be read at any time without affecting the counting operation.

New initial values can be written to the prescaler or the counter/timer registers at any time. These values are transferred to their respective down-counters on the next load operation. If the counter/timer mode is continuous, the next load occurs on the timer clock following an end-of-count. New initial values must be written before the load operation because the prescaler always effectively operates in continuous count mode.

If the value loaded in the T_1 register is 01h, the timer is actually not timing or counting at all; the timer is passing the prescaler end-of-count through. Because the prescaler is continuously running, regardless of the single-pass/continuous mode operation, the 8-bit timer continuously times out at the rate of the prescaler end-of-count if the T_1 timer value is programmed to 01h.

The time interval (i) until end-of-count, is given by

i = t x p x v

where t is 8 divided by XTAL frequency, p is the prescaler value (1 – 64), and v is the counter/timer value (1 – 256). The prescaler and counter/timer are true divide-by-n counters.

T_{OUT} Modes

The Timer Mode register TMR (F1h) (Figure 25) is used in conjunction with the Port 5 Mode register P5M to configure $P5_6$ for T_{OUT} operation. In order for T_{OUT} to function, $P5_6$ must be defined as an output line by setting P5M bit D_6 to 0. Output is controlled by one of the counter/timers (T_0 or T_1) or the internal clock.

R241 TMR
Timer Mode Register
(F1h; Read/Write)

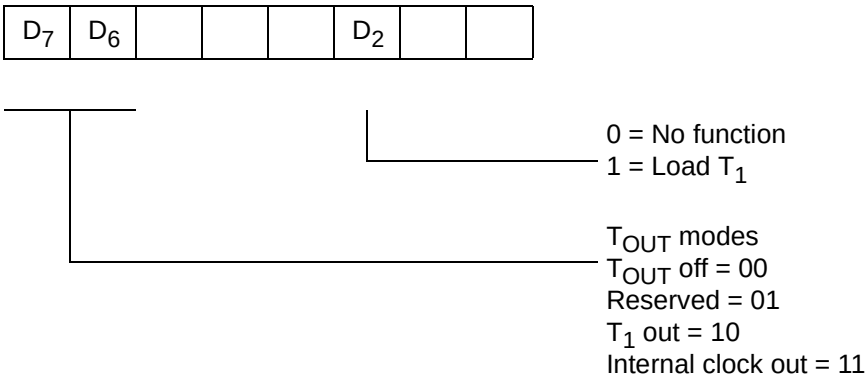


Figure 25. Timer Mode Register T_{OUT} Operation

The $P5_6$ output is selected by TMR bits D_7 and D_6. T_1 is selected by setting D_7 and D_6 to 1 and 0, respectively. The counter/timer T_{OUT} mode is turned off by setting TMR bits D_7 and D_6 both to 0, freeing $P3_6$ to be a data output line.

T_{OUT} is initialized to a logic 1 whenever the TMR Load bit D_2 is set to 1.

At end-of-count, the interrupt request line IRQ_5 clocks a toggle flip-flop. The output of this flip-flop drives the T_{OUT} line $P5_6$. In all cases, when the counter/timer reaches its end-of-count, T_{OUT} toggles to its opposite state (see Figure 26). If, for example, the counter/timer is in continuous counting mode, T_{OUT} has a 50% duty cycle output. You can control the duty cycle by varying the initial values after each end-of-count.

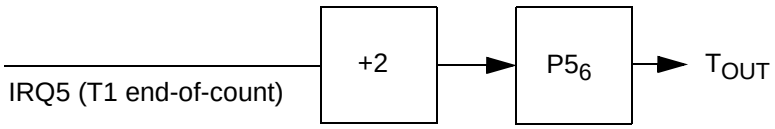


Figure 26. Counter/Timer Output Using T_{OUT}

The internal clock can be selected as output instead of T_1 by setting TMR bits D_7 and D_6 both to 1. The internal clock (XTAL frequency/2) is then directly output on $P5_6$ (Figure 27).

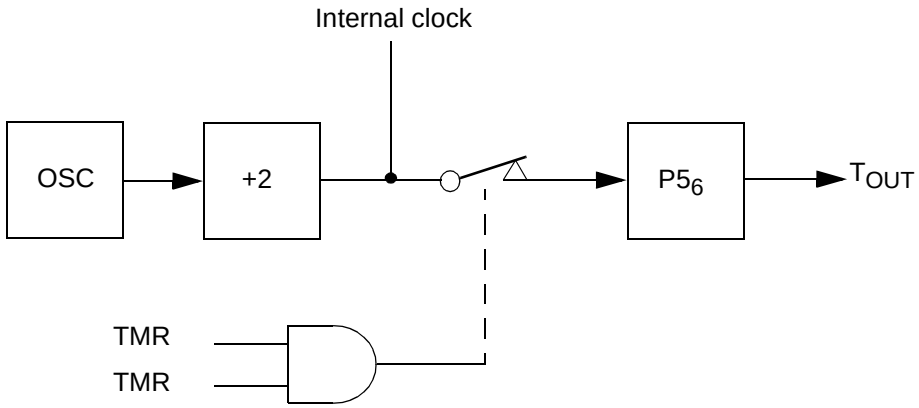


Figure 27. Internal Clock Output Using T_{OUT}

While programmed as T_{OUT}, $P5_6$ cannot be modified by a write to port register P5. However, the Z8 software can examine $P5_6$'s current output by reading the port register.

T_{IN} Modes

The Timer Mode register TMR (F1h) (Figure 28) is used in conjunction with the Prescaler register PRE_1 (F3h) (Figure 29) to configure $P5_2$ as T_{IN}. T_{IN} is used in conjunction with T_1 in one of four modes:

- External clock input
- Gated internal clock
- Triggered internal clock
- Retriggerable internal clock

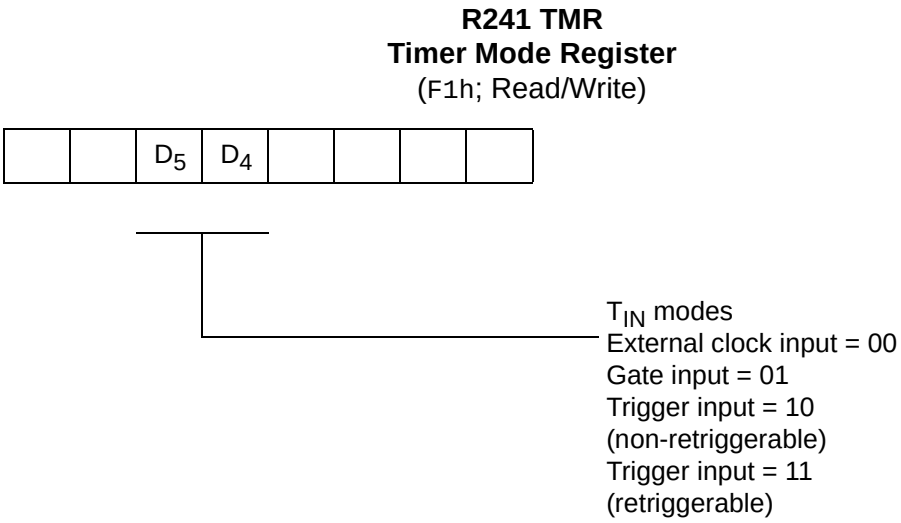


Figure 28. Timer Mode Register T_{IN} Operation

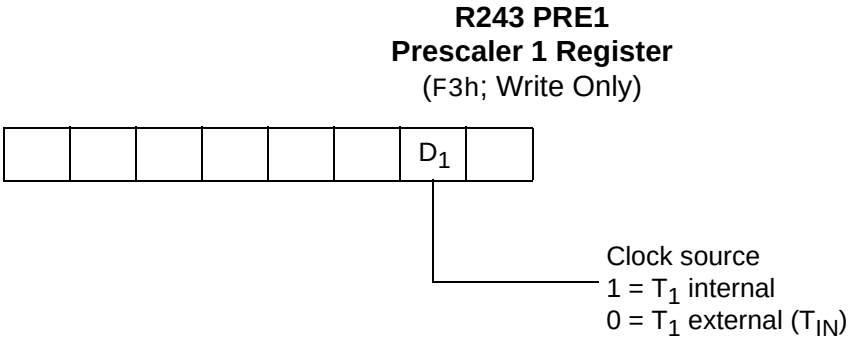


Figure 29. Prescaler 1 T_{IN} Operation

The T_1 counter/timer clock source must be configured for external by setting PRE_1 bit D_2 to 0. The Timer Mode register bits D_5 and D_4 can then be used to select the T_{IN} operation.

For T_1 to start counting as a result of a T_{IN} input, the Enable Count bit D_3 in TMR must be set to 1. When using T_{IN} as an external clock or a gate input, the initial values must be loaded into the down-counters by setting the Load bit D_2 in TMR to 1 before counting begins. Initial values are automatically loaded in Trigger and Retrigger modes, so software loading is unnecessary.

Configure $P5_2$ as an input line by setting P5M bit D_2 to 1.

Each High-to-Low transition on T_{IN} generates interrupt request IRQ_0, regardless of the selected T_{IN} mode or the enabled/disabled state of T_1. IRQ_0 must therefore be masked or enabled according to the needs of the application.

External Clock Input Mode

The T_{IN} External Clock Input mode (TMR bits D_5 and D_4 both set to 0) supports the counting of external events, where an event is considered to be a High-to-Low transition on T_{IN} (see Figure 30) occurrence (single-pass mode) or on every nth occurrence (continuous mode) of that event.

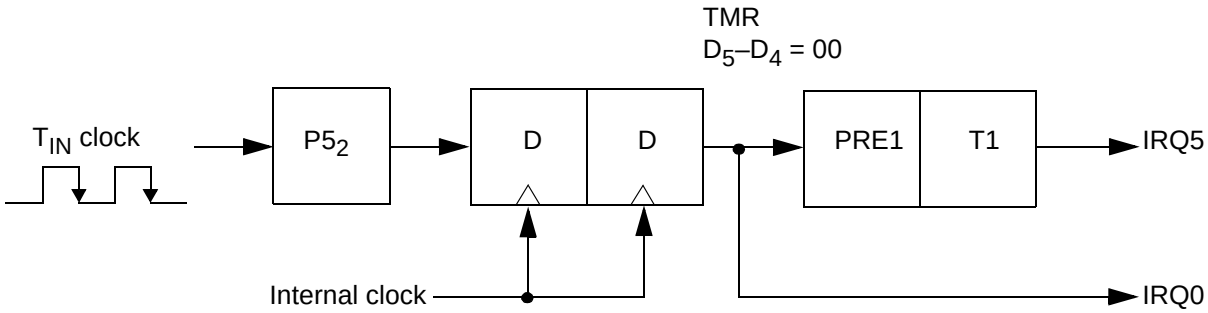


Figure 30. External Clock Input Mode

Gated Internal Clock Mode

The T_{IN} Gated Internal Clock mode (TMR bits D_5 and D_4 set to 0 and 1, respectively) measures the duration of an external event. In this mode, the T_1 prescaler is driven by the internal timer clock, gated by a High level on T_{IN} (see Figure 31). T_1 counts while T_{IN} is High and stops counting when T_{IN} is Low. Interrupt request IRQ_0 is generated on the High-to-Low transition of T_{IN}, signaling the end of the gate input. Interrupt request IRQ_5 is generated if T_1 reaches its end-of-count.

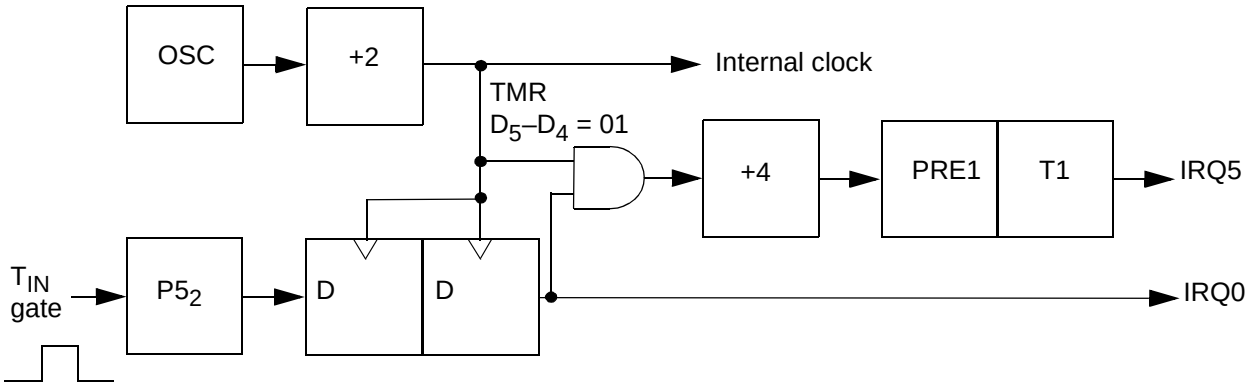


Figure 31. Gated Clock Input Mode

Triggered Input Mode

The T_{IN} Triggered Input mode (TMR bits D_5 and D_4 set to 1 and 0, respectively) causes T_1 to start counting as the result of an external event (see Figure 32). T_1 is then loaded and clocked by the internal timer clock following the first High-to-Low transition on the T_{IN} input. Subsequent T_{IN} transitions do not affect T_1. In the single-pass mode, the Enable bit is reset whenever T_1 reaches its end-of-count. Further T_{IN} transitions have no effect on T_1 until software sets the Enable Count bit again. In continuous mode, when T_1 is triggered, counting continues until software resets the Enable Count bit. Interrupt request IRQ_5 is generated when T_1 reaches its end-of-count.

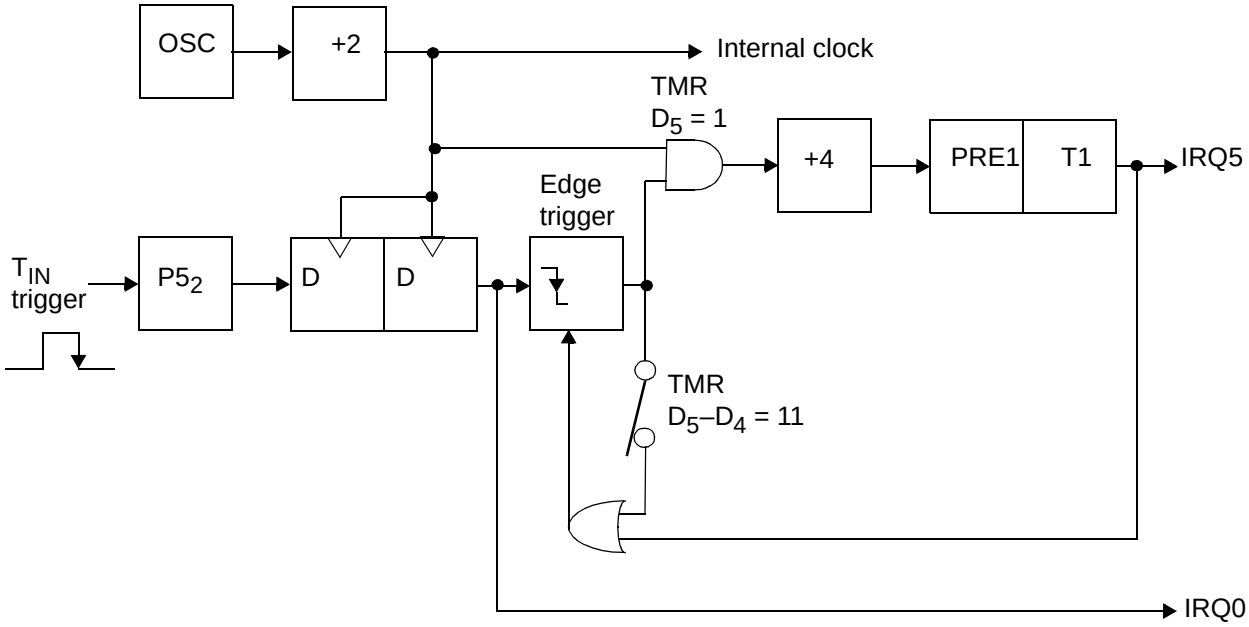


Figure 32. Triggered Clock Mode

Retriggerable Input Mode

The T_{IN} Retriggerable Input mode (TMR bits D_5 and D_4 both set to 1) causes T_1 to load and start counting on every occurrence of a High-to-Low transition on T_{IN} (see Figure 32). Interrupt request IRQ_5 is generated if the programmed time interval (determined by T_1 prescaler and counter/timer register initial values) has elapsed since the last High-to-Low transition on T_{IN}. In single-pass mode, the end-of-count resets the Enable Count bit. Subsequent T_{IN} transitions do not cause T_1 to load and start counting until software sets the Enable Count bit again. In continuous mode, counting continues when T_1 is triggered until software resets the Enable Count bit. When enabled, each High-to-Low T_{IN} transition causes T_1 to

reload and restart counting. Interrupt request IRQ_5 is generated on every end-of-count.

T8 and T16 Timer Operation

The T8 timer is a programmable 8-bit counter/timer with two 8-bit capture registers and two 8-bit load registers. The T16 timer is a programmable 16-bit counter/timer with one 16-bit capture register pair and one 16-bit load register pair. See Figure 33. The T8 and T16 counters/timers have two modes of operation:

- The transmit mode is used to generate complex waveforms. There are two submodes:
  - The normal mode can be used in single-pass or modulo-N (repeating) mode.
  - The ping-pong mode is used when the T8 timer counts down, enables the T16 timer that counts down, enabling T8, and so on, until the mode is disabled.
- The demodulation mode is used to capture and demodulate complex waveforms.

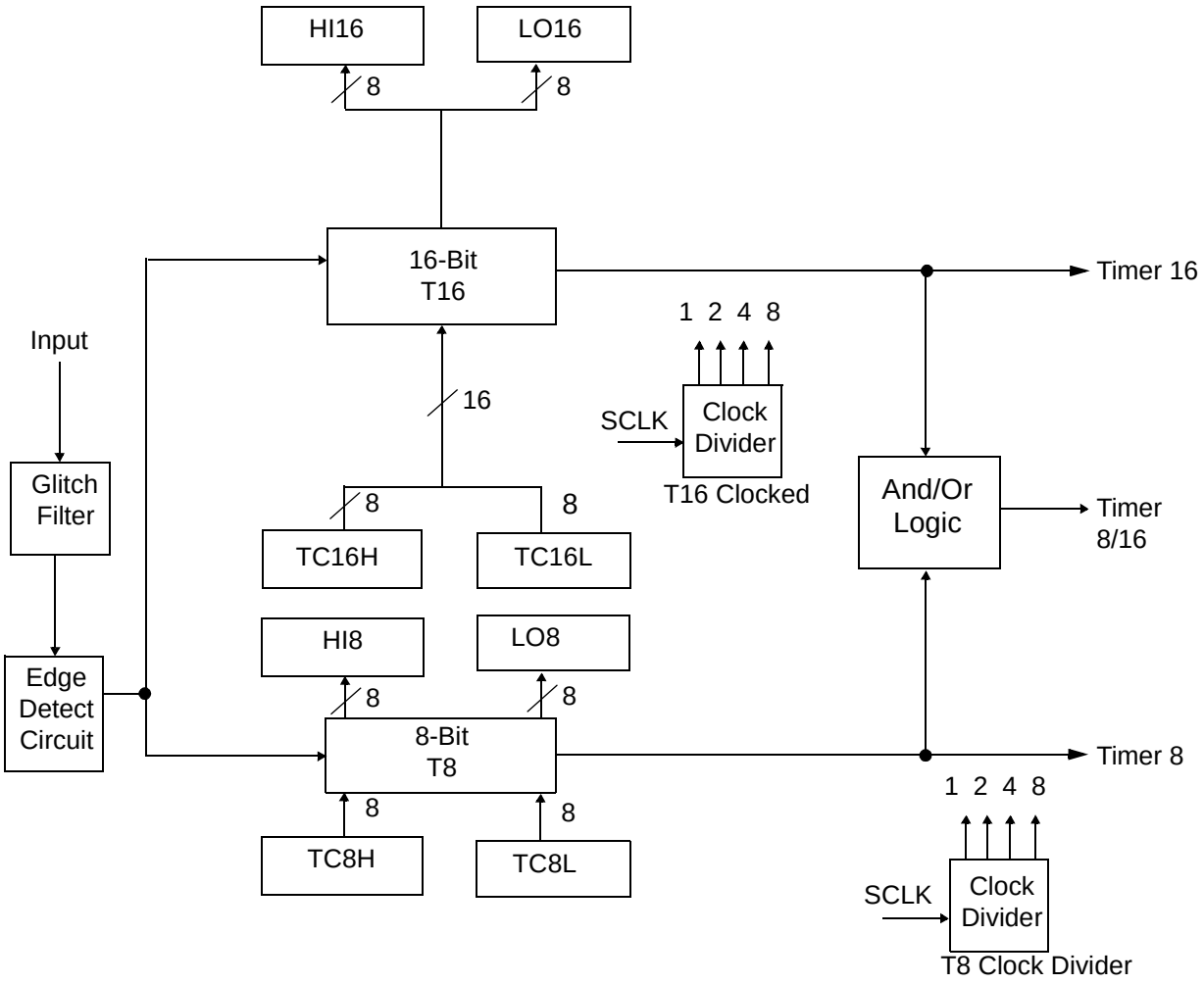


Figure 33. Counter/Timer Architecture

T8 Transmit Mode

Before T8 is enabled, the output of T8 depends on CTR1, D1. If CTR1, D1 is 0, T8_OUT is 1. If CTR1, D1 is 1, T8_OUT is 0.

When T8 is enabled, the output T8_OUT switches to the initial value (CTR1 D1). If the initial value (CTR1 D1) is 0, TC8L is loaded; otherwise, TC8H is loaded into the counter. In single-pass mode (CTR0 D6), T8 counts down to 0 and stops, T8_OUT toggles, the time-out status bit (CTR0 D5) is set, and a time-out interrupt can be generated if it is enabled (CTR0 D1). In modulo-N mode, upon reaching terminal count, T8_OUT is toggled, but no interrupt is generated. Then T8 loads a new count (if T8_OUT level is 0), TC8L is loaded; if T8_OUT is 1, TC8H is loaded.

T8 counts down to 0, toggles T8_OUT, sets the time-out status bit (CTR0 D5), and generates an interrupt if enabled (CTR0 D1). This completes one cycle. T8 then loads from TC8H or TC8L, according to the T8_OUT level, and repeats the cycle.

The user can modify the values in TC8H or TC8L at any time.The new values take effect when they are loaded. Do not write these registers at the time the values are to be loaded into the counter/timer. An initial count of 1 is not allowed. An initial count of 0 causes TC8 to count from 0 to FFh to FEh. Transition from 0 to FFh is not a time-out condition (see Figure 34).

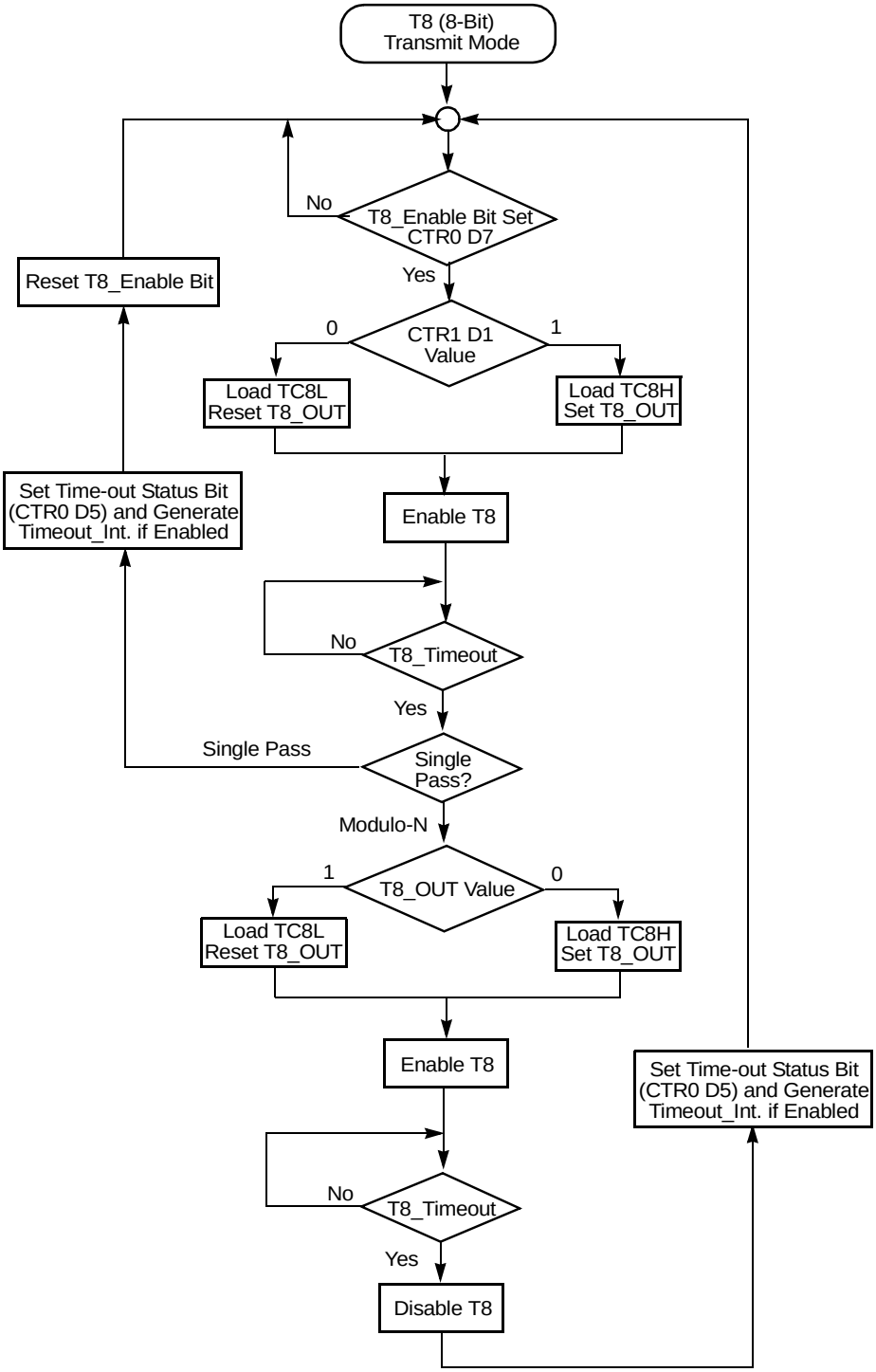


Figure 34. Transmit Mode Flowchart

Note: Do not use the same instructions for stopping the counter/timers and setting the status bits. Two successive commands are necessary—the first command for stopping counter/timers and the second command for resetting the status bits—because one counter/timer clock interval must complete for the initiated event to actually occur.

T8 Demodulation Mode

Program TC8L and TC8H to FFh. After T8 is enabled, when the first edge (rising, falling, or both, depending on CTR1 D5, D4) is detected, it starts to count down. When a subsequent edge (rising, falling, or both, depending on CTR1 D5, D4) is detected during counting, the current value of T8 is one's complemented and put into one of the capture registers. If T8 is a positive edge, data is placed in LO8. If T8 is a negative edge, data is placed in H18. One of the edge-detect status bits (CTR1 D1, D0) is set, and an interrupt can be generated if enabled (CTR0 D2). Meanwhile, T8 is loaded with TC8H and starts counting again. If T8 reaches 0, the time-out status bit (CTR0 D5) is set, and an interrupt can be generated if enabled (CTR0 D1), and T8 continues counting from FFh (see Figure 35).

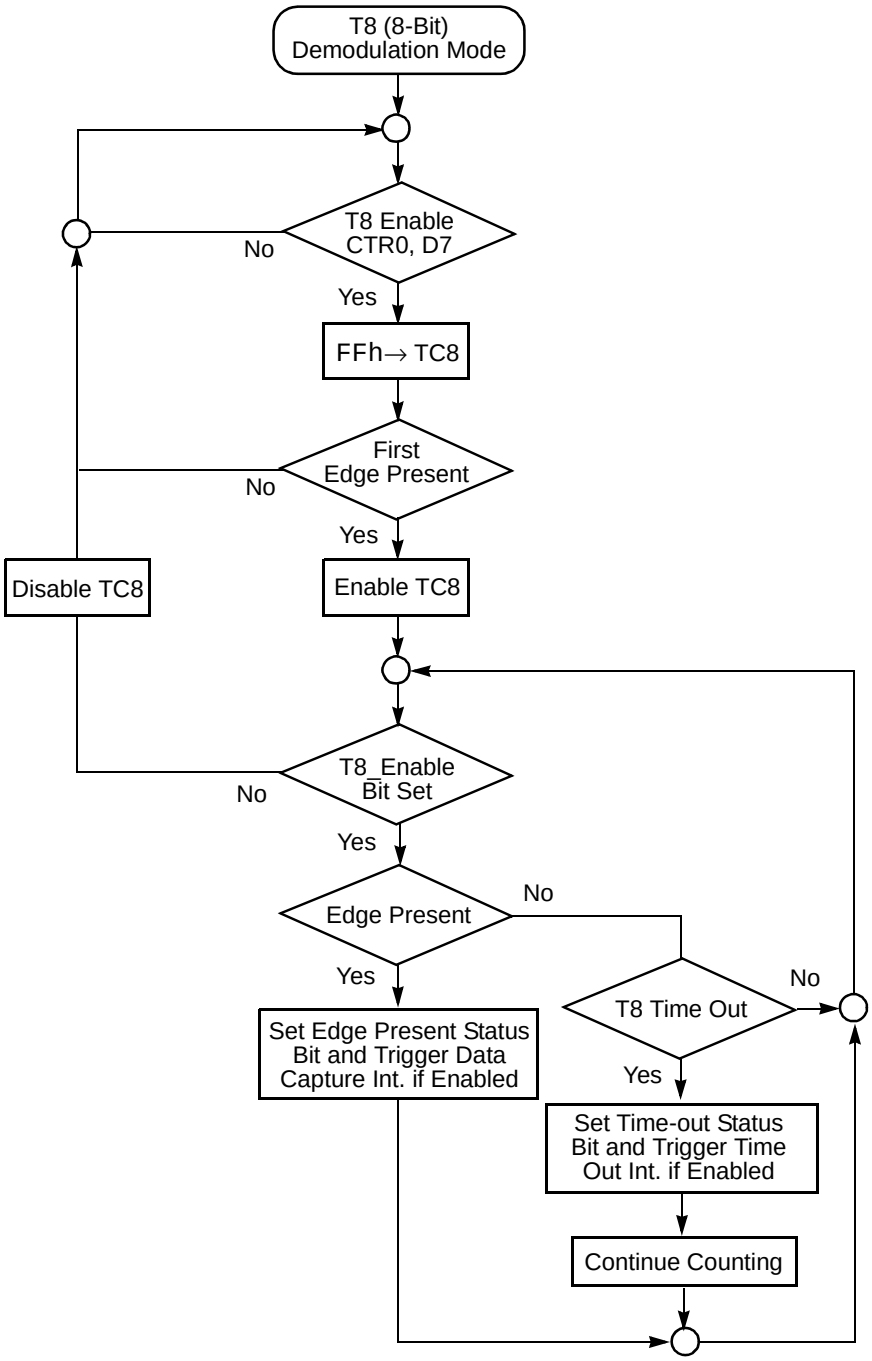


Figure 35. Demodulation Mode Flowchart

T16 Transit Mode

In normal or ping-pong mode, the output of T16, when not enabled, is dependent on CTR1, D0. If CTR1, D0 is a 0, T16_OUT is a 1; if CTR1, D0 is a 1, T16_OUT is 0. The user can force the output of T16 to either a 0 or 1, whether it is enabled or not, by programming CTR1 D3, D2 to a 10 or 11.

When T16 is enabled, TC16H * 256 + TC16L is loaded, and T16_OUT is switched to its initial value (CTR1 d0). When T16 counts down to 0, T16_OUT is toggled (in normal or ping-pong mode), an interrupt is generated if enabled (CTR2 D1), and a status bit (CTR2 D5) is set. If it is in modulo-N mode, it is loaded with TC16H * 256 + TC16L, and the counting continues.

The user can modify the values in TC16H and TC16L at any time. The new values take effect when they are loaded. Do not load these registers at the time the values are to be loaded into the counter/timer. An initial count of 1 is not allowed. An initial count of 0 causes T16 to count from 0 to FFFFh to FFFEh. Transition from 0 to FFFFh is not a time-out condition.

T16 Demodulation Mode

Program TC16L and TC16H to FFh. After T16 is enabled, when the first edge (rising, falling, or both, depending on CTR1 D5, D4) is detected, T16 captures HI16 and LO16, reloads, and begins counting.

Ping-Pong Mode

This operation mode is only valid in transmit mode. T8 and T16 must be programmed in single-pass mode (CTR0 D6, CTR2 D6), and ping-pong mode must be programmed in CTR1 D3, D2. The user can begin the operation by enabling either T8 or T16 (CTR0 D7 or CTR2 D7). For example, if T8 is enabled, T8_OUT is set to this initial value (CTR1 D1). According to T8_OUT's level, TC8H or TC8L is loaded into T8. After the terminal count is reached, T8 is disabled, and T16 is enabled. T16_OUT switches to its initial value (CTR1 D0), data from TC16H and TC16L is loaded, and T16 starts to count. After T16 reaches the terminal count, it stops. T8 is enabled again, and the whole cycle repeats. Interrupts can be allowed when T8 or T16 reaches terminal control (CTR0 D1, CTR2 D1). To stop the ping-pong operation, write 00 to bits D3 and D2 or CTR1.

Note: Enabling ping-pong operation while the counters/timers are running can cause intermittent counter/timer function. Disable the counters/timers, then reset the status flags before starting the ping-pong mode.

Control and Status Registers

The Z86D99/Z86L99 family has 4 I/O port registers, 33 status and control registers, and 233 general-purpose RAM registers. The I/O port and control registers are included in the general-purpose register memory to allow any Z8 instruction to process I/O or control information directly, thus eliminating the requirement for special I/O or control instructions. The Z8 instruction set permits direct access to any of these 37 registers. In addition, each of the 233 general-purpose registers can also function as an accumulator, an address pointer, or an index register. Registers identified as “Reserved” do not exist or have not been implemented in this design.

Register Summary

Table 10 through Table 13 summarize the name and location of all registers. The register-by-register descriptions follow this section.

Table 10. I/O Port Registers (Group 0, Bank 0, Registers 0–F)

Grp/Bnk Reg		Register Function	Identifier
(00h)	rF	General-Purpose RAM Register	GPR
(00h)	rE	General-Purpose RAM Register	GPR
(00h)	rD	General-Purpose RAM Register	GPR
(00h)	rC	General-Purpose RAM Register	GPR
(00h)	rB	General-Purpose RAM Register	GPR
(00h)	rA	General-Purpose RAM Register	GPR
(00h)	r9	General-Purpose RAM Register	GPR
(00h)	r8	General-Purpose RAM Register	GPR
(00h)	r7	Analog/Digital Converted Data	ADCDATA
(00h)	r6	Port 6 Control Register	P6
(00h)	r5	Port 5 Control Register	P5
(00h)	r4	Port 4 Control Register	P4
(00h)	r3	Reserved	
(00h)	r2	Port 2 Control Register	P2
(00h)	r1	Reserved	
(00h)	r0	Reserved	

Table 11. Control and Status Registers (Group F, Bank 0, Registers 0–F)

Grp/Bnk Reg		Register Function	Identifier
(F0h)	rF	Stack Pointer	SP
(F0h)	rE	General-purpose RAM Register	GPR
(F0h)	rD	Register Pointer	RP
(F0h)	rC	Program Control Flag Register	Flags
(F0h)	rB	Interrupt Mask Register	IMR
(F0h)	rA	Interrupt Request Register	IRQ
(F0h)	r9	Interrupt Priority Register	IPR
(F0h)	r8	Reserved	
(F0h)	r7	Port 3 Mode Register	P3M
(F0h)	r6	Port 2 Mode Register	P2M
(F0h)	r5	Reserved	
(F0h)	r4	Reserved	
(F0h)	r3	T1 Prescale Register	PRE1
(F0h)	r2	T1 Data Register	T1
(F0h)	r1	T1 Mode Register	TMR
(F0h)	r0	Reserved	

Table 12. Timer Control Registers (Group 0, Bank D, Registers 0–F)

Grp/Bnk Reg		Register Function	Identifier
(0Dh)	rF	Reserved	
(0Dh)	rE	Reserved	
(0Dh)	rD	Reserved	
(0Dh)	rC	Low-Battery Detect Flag	LB
(0Dh)	rB	T16 MS-Byte Capture Register	HI8
(0Dh)	rA	T16 LS-Byte Capture Register	LO8
(0Dh)	r9	T8 High Capture Register	HI16
(0Dh)	r8	T8 Low Capture Register	LO16
(0Dh)	r7	T16 MS-Byte Hold Register	TC16H
(0Dh)	r6	T16 LS-Byte Hold Register	TC16L
(0Dh)	r5	T8 High Hold Register	TC8H
(0Dh)	r4	T8 Low Hold Register	TC8L
(0Dh)	r3	T8/T16 Control Register B	CTR3
(0Dh)	r2	T16 Control Register	CTR2
(0Dh)	r1	T8/T16 Control Register A	CTR1
(0Dh)	r0	T8 Control Register	CTR0

Table 13. SMR and Port Mode Registers (Group 0, Bank F, Registers 0–F)

Grp/Bnk	Reg	Register Function	Identifier
(0Fh)	rF	Reserved	
(0Fh)	rE	Reserved	
(0Fh)	rD	Reserved	
(0Fh)	rC	Reserved	
(0Fh)	rB	Stop Mode Recovery Register	SMR
(0Fh)	rA	Reserved	
(0Fh)	r9	Reserved	
(0Fh)	r8	ADC Control Register	ADCCTRL
(0Fh)	r7	Reserved	
(0Fh)	r6	Port 6 Mode	P6M
(0Fh)	r5	Port 5 Stop Mode Recovery	P5SMR
(0Fh)	r4	Port 5 Mode Register	P5M
(0Fh)	r3	Reserved	
(0Fh)	r2	Port 4 Mode Register	P4M
(0Fh)	r1	Port 2 Stop Mode Recovery	P2SMR
(0Fh)	r0	Port Configuration Register	P456CON

Register Error Conditions

Registers in the Z8 Standard Register File must be used correctly because certain conditions produce inconsistent results and must be avoided.

- Registers F5h–F9h are write-only registers. If an attempt is made to read these registers, FFh is returned. Reading any write-only register returns FFh.
- When the Register Pointer (register FDH) is read, the least significant four bits (lower nibble) indicate the current Expanded Register File Bank. (For example, 0000 indicates the Standard Register File, while 1010 indicates Expanded Register File Bank A.)
- Writing to bits that are selected as timer outputs changes the I/O register but has no effect on the pin signal.
- The Z8 instruction DJNZ uses any general-purpose working register as a counter.
- Logical instructions such as OR and AND require that the current contents of the operand be read. They do not function properly on write-only registers.

Registers (Grouped by Function)

The following is a summary of the 37 special-purpose registers of the Z86D99/ Z86L99 family grouped by function. The following are the functional groups:

- Flags and Pointers
- Analog-to-Digital Converter Control
- Interrupt Control
- I/O Port Control
- Timer Control—General-Purpose Timer (T1)
- Timer Control—T8 and T16 Timers
- Stop-Mode Recovery Control

For any of the registers described in this section (see Table 14), bits identified as “Reserved” either do not exist (meaning they have not been implemented in this design) or have a special purpose in a ZiLOG engineering or test environment.

Caution: Do not attempt to use these bits as the results are unpredictable and meaningless.

Table 14. Register Description Locations

Address Grp/Bnk Register	Register Function	Symbol	Location
00h r2 (R2)	Port 2 Data	P2	page 68
00h r4 (R4)	Port 4 Data	P4	page 69
00h r5 (R5)	Port 5 Data	P5	page 70
00h r6 (R6)	Port 6 Data	P6	page 71
00h r7 (R7)	ADC Data	ADCDATA	page 62
0Dh r0	T8 Timer Control	CTR0	page 77
0Dh r1	T8/T16 Control (A)	CTR1	page 74
0Dh r2	T16 Timer Control	CTR2	page 80
0Dh r3	T8/T16 Control (B)	CTR3	page 76
0Dh r4	T8 Low Load	TC8L†	page 79
0Dh r5	T8 High Load	TC8H†	page 79
0Dh r6	T16 Low Load	TC16L†	page 82
0Dh r7	T16 High Load	TC16H†	page 82
0Dh r8	T16 Low Capture	LO16†	page 81
0Dh r9	T16 High Capture	HI16†	page 81
0Dh r10	T8 Low Capture	LO8†	page 78

Table 14. Register Description Locations (Continued)

Address Grp/Bnk Register		Register Function	Symbol	Location
0Dh	r11	T8 High Capture	HI8†	page 78
0Dh	r12	Low Battery Detect	LB	page 60
0Fh	r0	Port Configuration (A)	P456CON	page 67
0Fh	r1	Port 2 SMR Source	P2SMR	page 84
0Fh	r2	Port 4 Mode	P4M	page 69
0Fh	r4	Port 5 Mode	P5M	page 70
0Fh	r5	Port 5 SMR Source	P5SMR	page 84
0Fh	r6	Port 6 Mode	P6M	page 71
0Fh	r8	ADC Control	ADCCTRL	page 61
0Fh	r11	Stop Mode Recovery	SMR	page 83
F0h	r1 (R241)	T1 Timer Mode	TMR	page 72
F0h	r2 (R242)	T1 Timer Data	T1	page 72
F0h	r3 (R243)	T1 Timer Prescale	PRE1	page 73
F0h	r6 (R246)	Port 2 Mode	P2M	page 68
F0h	r7 (R247)	Port Configuration (B)	P3M	page 67
F0h	r9 (R249)	Interrupt Priority	IPR	page 64
F0h	r10 (R250)	Interrupt Request	IRQ	page 65
F0h	r11 (R251)	Interrupt Mask	IMR	page 63
F0h	r12 (R252)	Program Control Flags	Flags	page 57
F0h	r13 (R253)	Register Pointer	RP	page 58
F0h	r15 (R255)	Stack Pointer	SP	page 59

Note: †This register is not reset following Stop Mode Recovery (SMR).

Flags and Pointer Registers

In addition to the three standard Z8 flag and pointer registers (Program Control Register Pointer, and Stack Pointer), the Z86D99/Z86L99 family includes a Low-Battery Detect Flag register.

Program Control Flag Register (Flags)

The Program Control Flag register (see Table 15) reflects the current status of the Z8 as shown in Table 15. The FLAGS register contains six bits of status information that are set or cleared by CPU operations. Four of the bits (C, V, Z, and S) can be tested for use with conditional jump instructions. Two flags (H and D) cannot be tested and are used for BCD arithmetic. The two remaining flags in the register (F1 and F2) are available to the user, but they must be set or cleared by instructions and are not usable with conditional jumps.

Table 15. FLAGS Register [Group/Bank F0h, Register C (R252)]

Bit	7	6	5	4	3	2	1	0
Bit/Field	**C**	**Z**	**S**	**V**	**D**	**H**	**F2**	**F1**
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	X	X	X	X	X	X	X	X

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
7_____	Carry Flag (C)	R/W	1	Indicates the “carry out” of bit 7 position of a register being used as an accumulator; on Rotate and Shift instructions this bit contains the most recent value shifted out of the specified register
_6_____	Zero Flag (Z)	R/W	1	Indicates that the contents of an accumulator register is zero following an arithmetic or logical operation
__5_____	Sign Flag (S)	R/W	X	Stores the value of the most significant bit of a result following an arithmetic, logical, Rotate, or Shift operation; in arithmetic operations on signed numbers, a positive number is identified by a 0, and a negative number is identified by a 1
___4____	Overflow Flag (V)	R/W	1	For signed arithmetic, Rotate, and Shift operations, the flag is set to 1 when the result is greater than the maximum possible number (>127) or less than the minimum possible number (<-128) that can be represented in two’s complement form; following logical operations, this flag is set to 0

Table 15. FLAGS Register [Group/Bank F0h, Register C (R252)] (Continued)

____3___	Decimal Adjust Flag (D)	R/W	1 0	Used for BCD arithmetic—after a subtraction, the flag is set to 1; following an addition, it is cleared to 0
_____2__	Half Carry Flag (H)	R/W	1 0	Set to 1, whenever an addition generates a “carry out” of bit position 3 (overflow) of an accumulator; or subtraction generates a “borrow into” bit 3
______1_	User Flag (F2)	R/W	1 0	User definable
_______0	User Flag (F1)	R/W	1 0	User definable

Register Pointer (RP)

Z8 instructions can access registers directly or indirectly using either a 4-bit or 8-bit address field. The upper nibble of the Register Pointer, as described in Table 16, contains the base address of the active Working Register GROUP. The lower nibble contains the base address of the Expanded Register File BANK. When using 4-bit addressing, the 4-bit address of the working register (r0 to rF) is combined with the upper nibble of the Register Pointer (identifying the WR GROUP), thus forming the 8-bit actual address.

Table 16. RP Register [Group/Bank F0h, Register D (R253)]

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit/Field	**Working Register Group**				**Expanded Register File Bank**			
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	**Bit/Field**	**R/W**	**Value**	**Description**
7654_____	Working Register Group Pointer	R/W	X	Identifies 1 of 16 possible WR Groups, each containing 16 Working Registers
_____3210	Expanded Register File Bank Pointer	R/W	X	Identifies 1 of 16 possible ERF Banks; only Banks 0, D, and F are valid for the Z86D99/Z86L99 family

Stack Pointer (SP)

The Z86D99/Z86L99 family of products is configured for an internal stack. The size of the stack is limited only by the available memory space or general-purpose RAM registers dedicated to this task. An 8-bit stack pointer, as described in Table 17, is used for all stack operations.

Table 17. SP Register [Group/Bank F0h, Register F (R255)]

Bit	7	6	5	4	3	2	1	0
Bit/Field	Stack Pointer							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	X	X	X	X	X	X	X	X

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210	Stack Pointer	R/W	X	Points to the data stored on the top of the stack; an overflow or underflow can occur if the stack address is incremented or decremented during normal stack operations

Low-Battery Detect Flag (LB)

When the Z86D99/Z86L99 is used in a battery-operated application, one of the on-chip comparators can be used to check that the V_{CC} is at the required level for correct operation of the device. When voltage begins to approach the V_{BO} point, an on-chip low-battery detection circuit is tripped, which in turn sets a user-readable flag. The LB register, as described in Table 18, is used to set and reset the LB flag.

Table 18. LB Register (Group/Bank 0Dh, Register C)

Bit	7	6	5	4	3	2	1	0
Bit/Field	Reserved					Pad LVD	LVD_ Flag	LVD_ Enable
R/W	W	W	W	W	W	R/W	R/W	R/W
Reset	1	1	1	1	1	X	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543___	Reserved	R	1	Always reads 11111
		W	X	No Effect
_____2__	Pad LVD	R	1	Pad is not regulated when P43=0 (V_{pad}<V_{min}; see page 33)
		R	0	Pad is regulating the current when P43=0 (V_{pad}>V_{min}; see page 33)
		W	X	Reset Pad LB flag
______1_	LVD_Flag	R	1	LB Flag Set if V_{DD}<V_{LV}
		R	0	LB Flag Reset
		W	X	No Effect
_______0	LVD_Enable	R/W	1	Enable LB *
			0	Disable LB

Note: * When LVD is enabled, IRQ5 is set only for low-voltage detection. Timer 1 will not generate an interrupt request.

Note: The LB flag will be valid after enabling the detection for 20 µS (design estimation, not tested in production). LB does not work at STOP mode. It must be disabled during STOP mode in order to reduce current.

Analog-to-Digital Converter Control Registers

The Z86D99/Z86L99 family features an 8-bit analog-to-digital converter with external voltage references. The output of the ADC is stored in the ADC Data Register, as shown in Table 20. The ADC is configured using the ADC Control Register, as shown in Table 19.

Table 19. ADCCTRL Register (Group/Bank 0Fh, Register 8)

Bit	7	6	5	4	3	2	1	0
Bit/Field	P47_ A/D	P46_ A/D	P45_ A/D	P44_ A/D	Channel Selection		A/D Pwr On	ADC Clock Select
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
7_____	P47_A/D	R/W	1 0	P47 configured as A/D Input P47 configured as digital input
_6_____	P46_A/D	R/W	1 0	P46 configured as A/D Input P46 configured as digital input
__5____	P45_A/D	R/W	1 0	P45 configured as A/D Input P45 configured as digital input
___4___	P44_A/D	R/W	1 0	P44 configured as A/D Input P44 configured as digital input
____32__	Channel Selection	R/W	11 10 01 00	Channel 3 (P47) Channel 2 (P46) Channel 1 (P45) Channel 0 (P44)
_____1_	A/D_PowerON	R/W	1 0	ON OFF
_____0	ADC Clock Select	R/W	1 0	SCLK/2 SCLK

ADC Control Register (ADCCTRL)

The ADCCTRL register controls the operation of the analog-to-digital converter. Bits 2 and 3 of the ADCCTRL register determine which of the four analog input channels feeds into the ADC at any given time. Bits 4 through 7 enable or disable the digital input buffer. When configured as an ADC input channel, the port has to be configured in Input Mode and with the digital input buffer disabled.

ADC Data Register (ADCDATA)

The ADCDATA register is a read-only register that contains the digital output of the analog-to-digital converter. See Table 20.

Table 20. ADCDATA Register (Group/Bank 00h, Register 7)

Bit	7	6	5	4	3	2	1	0
Bit/Field	ADC Data							
R/W	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210	ADC Data	R	Data	Output of the ADC
		W	X	No Effect

Interrupt Control Registers

The Z8 allows up to six different interrupts from a variety of sources. These interrupts can be masked and their priorities set by using the Interrupt Mask Register and Interrupt Priority Register. The Interrupt Request Register stores the interrupt requests for both vectored and polled interrupts.

Interrupt Mask Register

The IMR, as described in Table 21, individually or globally enables the six interrupt requests. Bit 7 of the IMR is the master enable and must be set before any of the individual interrupt requests can be recognized. Bit 7 must be set and reset by the enable interrupts and disable interrupts instructions only. The IMR is automatically reset during an interrupt service routine and set following the execution of an Interrupt Return (IRET) instruction.

Table 21. IMR (Group/Bank 0Fh, Register B)

Bit	7	6	5	4	3	2	1	0
Bit/Field	**Master**	**Re-served**	**IRQ5**	**IRQ4**	**IRQ3**	**IRQ2**	**IRQ1**	**IRQ0**
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
7_____	Master	R/W	1 0	Enable Master Interrupt Disable Master Interrupt
_6_____	Reserved	R W	1 X	Always reads 1 No Effect
__5_____	IRQ_5	R/W	1 0	Enable IRQ_5 Disable IRQ_5
___4____	IRQ_4	R/W	1 0	Enable IRQ_4 Disable IRQ_4
____3___	IRQ_3	R/W	1 0	Enable IRQ_3 Disable IRQ_3
_____2__	IRQ_2	R/W	1 0	Enable IRQ_2 Disable IRQ_2
______1_	IRQ_1	R/W	1 0	Enable IRQ_1 Disable IRQ_1
_______0	IRQ_0	R/W	1 0	Enable IRQ_0 Disable IRQ_0

Note: Bit 7 must be reset by the DI instruction before the contents of the Interrupt Mask Register or the Interrupt Priority Register are changed except in the following situations:

- Immediately after a hardware reset
- Immediately after executing an interrupt service routine and before IMR bit 7 has been set by any instruction

Interrupt Priority Register (IPR)

The IPR, as described in Table 22, is a write-only register that sets priorities for the vectored interrupts in order to resolve simultaneous interrupt requests. There are 48 sequence possibilities for interrupts. The six interrupts, IRQ_0 to IRQ_5, are divided into three groups of two interrupt requests each, as follows:

- Group A consists of IRQ_3 and IRQ_5
- Group B consists of IRQ_0 and IRQ_2
- Group C consists of IRQ_1 and IRQ_4

Table 22. IPR (Group/Bank 0Fh, Register 9)

Bit	7	6	5	4	3	2	1	0
Bit/Field	Reserved		Grp A IRQ3_5	Int_Group		Grp B IRQ0_2	Grp C IRQ1_4	Int_ Group
R/W	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76_____	Reserved	W	X	No Effect
__5_____	Grp A Priority: IRQ_3 and IRQ_5	W	1 0	IRQ_3>IRQ_5 (Group A) IRQ_5>IRQ_3
___43__0	Interrupt Group Priority	W	111 110 101 100 011 010 001 000	Reserved B>A>C C>B>A B>C>A A>C>B A>B>C C>A>B Reserved
_____2__	Grp B Priority: IRQ_0 and IRQ_2	W	1 0	IRQ_0>IRQ_2 (Group B) IRQ_2>IRQ_0
______1_	Grp C Priority: IRQ_1 and IRQ_4	W	1 0	IRQ_4>IRQ_1 (Group C) IRQ_1>IRQ_4

Priorities can be set both within and between groups using the IPR. Bits 1, 2, and 5 of the IPR define the priority of individual members within the groups. Bits 0, 3, and 4 are encoded to define six priority orders between the three groups. Bits 6 and 7 are reserved.

Interrupt Request Register

The IRQ, as described in Table 23, is a read/write register that stores the interrupt requests for both vectored and polled interrupts. When an interrupt request is made by any of the six interrupts, the corresponding bit in the IRQ is set to 1.

Table 23. IRQ (Group/Bank 0Fh, Register A)

Bit	7	6	5	4	3	2	1	0
Bit/Field	Interrupt Edge		Set IRQ5	Set IRQ4	Set IRQ3	Set IRQ2	Set IRQ1	Set IRQ0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76_____	Interrupt Edge Trigger	R/W	11 10 01 00	P51 Rise/FallingP52 Rise/Falling P51 Rising P52 Falling P51 FallingP52 Rising P51 FallingP52 Falling
__5_____	Set IRQ_5	R R W W	1 0 1 0	IRQ_5 Inactive IRQ_5 Active Set IRQ_5 Reset IRQ_5
___4____	Set IRQ_4	R R W W	1 0 1 0	IRQ_4 Inactive IRQ_4 Active Set IRQ_4 Reset IRQ_4
____3___	Set IRQ_3	R R W W	1 0 1 0	IRQ_3 Inactive IRQ_3 Active Set IRQ_3 Reset IRQ_3
_____2__	Set IRQ_2	R R W W	1 0 1 0	IRQ_2 Inactive IRQ_2 Active Set IRQ_2 Reset IRQ_2
______1_	Set IRQ_1	R R W W	1 0 1 0	IRQ_1 Inactive IRQ_1 Active Set IRQ_1 Reset IRQ_1
_______0	Set IRQ_0	R R W W	1 0 1 0	IRQ_0 Inactive IRQ_0 Active Set IRQ_0 Reset IRQ_0

Whenever a power-on reset is executed, the IRQ is reset to 00h and disabled. Before the IRQ accepts requests, it must be enabled by executing an enable interrupts instruction.

Note: IRQ is always cleared to 00h and is in read-only mode until the first EI instruction that enables the IRQ to be read/write. Setting the Global Interrupt Enable bit in the Interrupt Mask Register (IMR bit 7) does not enable the IRQ. Execution of an EI instruction is required.

For polled processing, IRQ must be initialized by an EI instruction. To properly initialize the IRQ, the following code is provided:

```
CLR   IMR   ; make sure vectored interrupts are disabled
EI          ; enable IRQ, otherwise it is read only
            ; not necessary, if interrupts were previously
            ; enabled
DI          ; disable interrupt handling
```

IMR is cleared before the IRQ enabling sequence to ensure no unexpected interrupts occur when EI is executed. This code sequence must be executed before programming the application required values for IPR and IMR.

I/O Port Control Registers

Each of the four ports (Ports 2, 4, 5, and 6) has an input register, an output register, and an associated buffer and control logic. Because there are separate input and output registers associated with each port, writing bits defined as inputs stores the data in the output register. This data cannot be read as long as the bits are defined as inputs. However, if the bits are reconfigured as output, the data stored in the output register is reflected on the output pins and can then be read. This mechanism allows the user to initialize the outputs before driving their loads.

Port Configuration Registers (P456CON and P3M)

The port configuration register (described in Table 24) switches the comparator inputs from digital to analog and allows Ports 4, 5, and/or 6 to be switched from push/pull active outputs to open drain outputs. In ZiLOG Test Mode, bit 3 of this register is used to enable the Address Strobe/Data Strobe. Bit 3 is not available in User Mode.

Table 24. P456CON Register (Group/Bank 0Fh, Register 0)

Bit	7	6	5	4	3	2	1	0
Bit/Field	Not Used		P51_ Mode	P52_ Mode	Reserved	P6_ Output	P5_ Output	P4_ Output
R/W	R/W	R/W	R/W	R/W	R/W	W	W	W
Reset	0	0	0	0	0	1	1	1†

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76_____	Not Used	R/W		These bits exist but do not have any function assigned to them; they are reserved for future extensions and must not be used.
__5_____	Comparator 1 Mode	R/W	1 0	Analog (P50, P51 as Inputs) Digital inputs
___4____	Comparator 2 Mode	R/W	1 0	Analog comparator inputs (P52, P53 configured as Inputs) Digital inputs
____3___	Reserved			
_____2__	Port 6 Output Configuration	W	1 0	Push-Pull Active Open Drain Outputs Always reads back 1*
______1_	Port 5 Output Configuration	W	1 0	Push-Pull Active Open Drain Outputs Always reads back 1*
_______0	Port 4 Output Configuration	W	1 0	Push-Pull Active Open Drain Outputs Always reads back 1*†

Note: *Do not use the read-modify-write instructions (for example, OR and AND) with this register. Bits 0, 1, and 2 always read back 1.

Note: †For Z86L990/L991, P43 can never be configured as push-pull. After any reset, P43 is configured as tristate high impedance.

Port 2 outputs are configured using the P3M Register, shown in Table 25. Bit 0 of the P3M Register switches Port 2 from push/pull active to open drain outputs. No other bits in this register are implemented.

Table 25. P3M Register [Group/Bank F0h, Register 7 (R247)]

Bit	7	6	5	4	3	2	1	0
Bit/Field	Reserved							P2_ Output
R/W	W	W	W	W	W	W	W	W
Reset	1	1	1	1	1	1	1	1

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
7654321_	Reserved	R	1	Always reads 1111111
		W	X	No Effect
_____0	Port 2 Output Configuration	W	1	Push-Pull Active
			0	Open Drain Outputs

Port 2 Control and Mode Registers (P2 and P2M)

Port 2 is a general-purpose 8-bit, bidirectional I/O port, as shown in Table 26. Each of the eight Port 2 I/O lines can be independently programmed as either input or output using the Port 2 Mode Register (see Table 27.)

Table 26. P2 Register [Group/Bank 00h, Register 2 (R2)]

Bit	7	6	5	4	3	2	1	0
Bit/Field	Port 2 Data							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	X	X	X	X	X	X	X	X

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210	Port 2 Data	R/W	Data	Port 2 Input/Output Register

Table 27. P2M Register [Group/Bank F0h, Register 6 (R246)]

Bit	7	6	5	4	3	2	1	0
Bit/Field	P27M	P26M	P25M	P24M	P23M	P22M	P21M	P20M
R/W	W	W	W	W	W	W	W	W
Reset	1	1	1	1	1	1	1	1

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210 (by bit)	Port 2 Mode Select	R	1	Always reads 11111111
		W	1	Input
		W	0	Output

A bit set to 1 in the P2M Register configures the corresponding bit in Port 2 as an input, while a bit set to 0 configures an output line.

Port 4 Control and Mode Registers (P4 and P4M)

Port 4 is a general-purpose 8-bit, bidirectional I/O port, as shown in Table 28. Each of the eight Port 4 I/O lines can be independently programmed as either input or output using the Port 4 Mode Register (see Table 29.)

Table 28. P4 Register [Group/Bank 00h, Register 4 (R4)]

Bit	7	6	5	4	3	2	1	0
Bit/Field	**Port 4 Data**							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	X	X	X	X	X	X	X	X

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210	Port 4 Data	R/W	Data	Port 4 Input/Output Register

Table 29. P4M Register (Group/Bank 0Fh, Register 2)

Bit	7	6	5	4	3	2	1	0
Bit/Field	**P47M**	**P46M**	**P45M**	**P44M**	**P43M**	**P42M**	**P41M**	**P40M**
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
7654_210 (by bit)	Port 4 Mode Select	R/W	1 0	Input Output
____3___	P43 Mode Select	R/W	0 1	Output Tristate High Impedance (available on Z86L990/L991 only)

A bit set to 1 in the P4M Register configures the corresponding bit in Port 4 as an input, while a bit set to 0 configures an output line.

Note: P43, the controlled current output pad, cannot be configured as an input. (P43 read = P43 out)

Port 5 Control and Mode Registers (P5 and P5M)

Port 5 is a general-purpose 8-bit, bidirectional I/O port, as shown in Table 30. Each of the eight Port 5 I/O lines can be independently programmed as either input or output using the Port 5 Mode Register (see Table 31.)

Table 30. P5 Register [Group/Bank 00h, Register 5 (R5)]

Bit	7	6	5	4	3	2	1	0
Bit/Field	**Port 5 Data**							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	X	X	X	X	X	X	X	X
R = Read, W = Write, X = Indeterminate								

Bit Position	Bit/Field	R/W	Value	Description
76543210	Port 5 Data	R/W	Data	Port 5 Input/Output Register

Table 31. P5M Register (Group/Bank 0Fh, Register 4)

Bit	7	6	5	4	3	2	1	0
Bit/Field	**P57M**	**P56M**	**P55M**	**P54M**	**P53M**	**P52M**	**P51M**	**P50M**
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1
R = Read, W = Write, X = Indeterminate								

Bit Position	Bit/Field	R/W	Value	Description
7654__10 (by bit)	Port 5 Mode Select	R/W	1 0	Input Output
____32__	P53, P52 Mode Select	R/W	1	Input Regardless of what is written to this pin, P53 and P52 are always in input mode.

A bit set to a 1 in the P5M Register configures the corresponding bit in Port 5 as an input, while a bit set to 0 configures an output line.

Note: Regardless of how P5M bits 2 and 3 are set, P52 and P53 are always in input mode.

Port 6 Control and Mode Registers (P6 and P6M)

Port 6 is a general-purpose 8-bit, bidirectional I/O port, as shown in Table 32. Each of the eight Port 6 I/O lines can be independently programmed as either input or output using the Port 6 Mode Register (see Table 33.)

Table 32. P6 Register [Group/Bank 00h, Register 6 (R6)]

Bit	7	6	5	4	3	2	1	0
Bit/Field	**Port 6 Data**							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	X	X	X	X	X	X	X	X
R = Read, W = Write, X = Indeterminate								
Bit Position	**Bit/Field**		**R/W**	**Value**	**Description**			
76543210	Port 6 Data		R/W	Data	Port 6 Input/Output Register			

Table 33. P6M Register (Group/Bank 0Fh, Register 6)

Bit	7	6	5	4	3	2	1	0
Bit/Field	**P67M**	**P66M**	**P65M**	**P64M**	**P63M**	**P62M**	**P61M**	**P60M**
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1
R = Read, W = Write, X = Indeterminate								
Bit Position	**Bit/Field**		**R/W**	**Value**	**Description**			
76543210 (by bit)	Port 6 Mode Select		R/W	1 0	Input Output			

A bit set to 1 in the P6M Register configures the corresponding bit in Port 6 as an input, while a bit set to 0 configures an output line.

Timer Control Registers—General-Purpose Timer (T1)

The Z86D99/Z86L99 family provides one standard 8-bit Z8 counter/timer, T1, driven by its own 6-bit prescaler, PRE1. T1 is independent of the processor instruction sequence, relieving software from time-critical operations such as interval timing or event counting. There are three registers that control the operation of T1: T1 Data Register (T1), T1 Mode Register (TMR), and T1 Prescale Register (PRE1). Because the timer, prescaler, and mode register are mapped into the standard Z8 register file, the software can treat the counter/timer as a general-purpose register, thus eliminating the requirement for special instructions.

T1 Data Register (T1)

The counter/timer register (T1) consists of an 8-bit down counter, a write-only register that holds the initial count value, and a read-only register that holds the current count value. The initial value of T1 can range from 1 to 255 (0 represents 256) (see Table 34.)

Table 34. T1 Register [Group/Bank F0h, Register 2 (R242)]

Bit	7	6	5	4	3	2	1	0
Bit/Field	**T1_Value**							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210	T_1 Value	R	Data	Current Value
		W	Data	Initial Value (Range 1 to 256 Decimal)

T1 Mode Register (TMR)

Under software control, T1 counter/timer is started and stopped using the T1 Mode Register as shown in Table 35.

Table 35. TMR Register [Group/Bank F0h, Register 1 (R241)]

Bit	7	6	5	4	3	2	1	0
Bit/Field	**TOUT_Mode**		**TIN_Mode**		**T1_ Count**	**T1_ Load**	**Reserved**	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	1	1

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76_____	T_{OUT} Mode	R/W	11	Internal Clock OUT on P56
			10	T_1OUT on P56
			01	Reserved
			00	Not used (P56 configured as I/O)
__54____	T_{IN} Mode	R/W	11	Trigger Input (Retriggerable)
			10	Trigger Input (Not-retriggerable)
			01	Gate Input
			00	External Clock Input (T_{IN} on P52)
____3___	T_1 Count	R/W	1	Enable T_1 Count
			0	Disable T_1 Count

Table 35. TMR Register [Group/Bank F0h, Register 1 (R241)] (Continued)

_____2__	T_1 Load	R/W	1 0	Load T_1 No effect
______10	Reserved	R W	1 X	Always reads 11 No effect

T1 Prescale Register (PRE1)

The T1 prescaler consists of an 8-bit register and a 6-bit down-counter. The six most significant bits (D2–D7) of PRE1 hold the prescaler's count modulo, a value from 1 to 64 decimal, as shown in Table 36. The prescale register also contains control bits that specify the counting mode and clock source for T1.

Table 36. PRE1 Register [Group/Bank F0h, Register 3 (R243)]

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit/Field	**Prescaler_Modulo**						**Clock_ Source**	**Count_ Mode**
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	**Bit/Field**	**R/W**	**Value**	**Description**
765432__	Prescaler Modulo	R/W	Data	Range: 1 to 64 Decimal
_____1_	Clock Source	R/W	1 0	T_1 Internal T_1 External (T_{IN} on P52)
_____0	Count Mode	R/W	1 0	T_1 Modulo-n T_1 Single Pass

Timer Control Registers—T8 and T16 Timers

One of the unique features of the Z86D99/Z86L99 family is a special timer architecture to automate the generation and reception of complex pulses or signals. This timer architecture consists of one programmable 8-bit counter timer with two capture registers and two load registers and a programmable 16-bit counter/timer with one 16-bit capture register pair and one 16-bit load register pair and their associated control registers. These counter/timers can work independently or can be combined together using a number of user-selectable modes governed by the T8/T16 control registers.

T8/T16 Control Register A (CTR1)

The T8/T16 Control Register A controls the functions in common with both the T_8 and T_{16} counter/timers. The T_8 and T_{16} counter/timers have two primary modes of operation: Transmit Mode and Demodulation Mode. Transmit Mode is used for generating complex waveforms. The Transmit Mode has two submodes: Normal Mode and Ping-Pong Mode. The settings for CTR1 in Transmit Mode are given in Table 37.

Table 37. CTR1 Register (In Transmit Mode) (Group/Bank 0Dh, Register 1)

Bit	7	6	5	4	3	2	1	0
Bit/Field	Mode	P43 Out	T8/T16_Logic		Transmit_ Submode		Initial_ T8_Out	Initial_ T16_ Out
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
7_____	Mode	R/W	1 0	Demodulation Transmit
_6_____	P43_Out	R/W	1 0	P43 configured as T8/T16 Output P43 configured as I/O
__54____	T_8/T_{16} Logic	R/W	11 10 01 00	NAND NOR OR AND
____32__	Transmit_ Submode	R/W	11 10 01 00	T16_Out = 1 T16_Out = 0 Ping-Pong Mode Normal Operation
_____1_	Initial_T8_Out	R/W	1 0	T8_Out set to 1 initially T8_Out set to 0 initially
_____0	Initial_T16_Out	R/W	1 0	T16_Out set to 1 initially T16_Out set to 0 initially

In Demodulation Mode, the T8 and T16 counter/timers are used to capture and demodulate complex waveforms. The settings for CTR1 in Demodulation Mode are given in Table 38.

Table 38. CTR1 Register (in Demodulation Mode) (Group/Bank 0Dh, Register 1)

Bit	7	6	5	4	3	2	1	0
Bit/Field	Mode	Demod_Input	Edge_Detect		Glitch_Filter		Rising Edge	Falling Edge
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
7_____	Mode	R/W	1	Demodulation
			0	Transmit
_6_____	Demodulator_ Input	R/W	1	P20 as Demodulator Input
			0	P51 as Demodulator Input
__54____	Edge_Detect	R/W	11	Reserved
			10	Both Edges
			01	Rising Edge
			00	Falling Edge
____32__	Glitch_Filter	R/W	11	16 SCLK Cycles
			10	8 SCLK Cycles
			01	4 SCLK Cycles
			00	No Filter
_____1_	Rising_Edge	R	1	Rising Edge Detected
		R	0	No Rising Edge
		W	1	Reset Flag to 0
		W	0	No Effect
_____0	Falling_Edge	R	1	Falling Edge Detected
		R	0	No Falling Edge
		W	1	Reset Flag to 0
		W	0	No Effect

T8/T16 Control Register B (CTR3)

The T8/T16 Control Register B, known as CTR3, is a new register to the Z86D99/ Z86L99 family. This register allows the T_8 and T_{16} counters to be synchronized. The settings of CTR3 are described in Table 39.

Table 39. CTR3 Register (Group/Bank 0Dh, Register 3)

Bit	7	6	5	4	3	2	1	0
Bit/Field	T16_ Enable	T8_ Enable	Sync Mode	Reserved				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	X	X	X	X	X

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
7_____	T_{16} Enable	R	1	Counter Enabled
		R	0	Counter Disabled
		W	1	Enable Counter
		W	0	Stop Counter
_6_____	T_8 Enable	R	1	Counter Enabled
		R	0	Counter Disabled
		W	1	Enable Counter
		W	0	Stop Counter
__5____	Sync Mode	R/W	1	Enable Sync Mode
			0	Diable Sync Mode
___43210	Reserved	R	1	Always reads 11111
		W	X	No Effect

T8 Control Register (CTR0)

As shown in Table 40, the T8 Control Register, known as CTR0, controls the operation of the 8-bit T_8 timer.

Table 40. CTR0 Register (Group/Bank 0Dh, Register 0)

Bit	7	6	5	4	3	2	1	0
Bit/Field	T8_ Enable	Single/ Modulo-n	Time_ Out	T8_Clock		Capture INT_ Mask	Counter INT_ Mask	P40_ Out
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
7_____	T_8 Enable	R	1	Counter Enabled
		R	0	Counter Disabled
		W	1	Enable Counter
		W	0	Stop Counter
_6_____	Single/ Modulo-n	R/W	1	Single Pass
			0	Modulo-n
__5____	Time_Out	R	1	Counter Timeout Occurred
		R	0	No Counter Timeout
		W	1	Reset Flag to 0
		W	0	No Effect
___43___	T_8 Clock	R/W	11	SCLK/8
			10	SCLK/4
			01	SCLK/2
			00	SCLK
_____2__	Capture Interrupt Mask	R/W	1	Enable Data Capture Interrupt
			0	Disable Data Capture Interrupt
______1_	Counter Interrupt Mask	R/W	1	Enable Time_Out Interrupt
			0	Disable Time_Out Interrupt
_______0	P40_Out	R/W	1	P40 configured as T_8 Output
			0	P40 configured as I/O

T8 High Capture Register (HI8)

The T8 High Capture Register, as described in Table 41, holds the captured data from the output of the T_8 counter/timer. This register is typically used to hold the number of counts when the input signal is high (or 1).

Table 41. HI8 Register (Group/Bank 0Dh, Register B)

Bit	7	6	5	4	3	2	1	0
Bit/Field	T8_Capture_HI							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210	T_8 Capture High Value	R W	Data	Captured Data No Effect

T8 Low Capture Register (LO8)

The T8 Low Capture Register, as described in Table 42, holds the captured data from the output of the T_8 counter/timer. This register is typically used to hold the number of counts when the input signal is low (or 0).

Table 42. LO8 Register (Group/Bank 0Dh, Register A)

Bit	7	6	5	4	3	2	1	0
Bit/Field	T8_Capture_LO							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210	T_8 Capture Low Value	R W	Data	Captured Data No Effect

T8 High Load Register (TC8H)

The T8 High Load Register, as described in Table 43, is loaded with the counter value necessary to keep the T8_Out signal in the high state for the required time.

Table 43. TC8H Register (Group/Bank 0Dh, Register 5)

Bit	7	6	5	4	3	2	1	0
Bit/Field	T8_Level_HI							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210	T_8 Level High Value	R/W	Data	Duration that T8_Out remains High

T8 Low Load Register (TC8L)

The T8 Low Load Register, as described in Table 44, is loaded with the counter value necessary to keep the T8_Out signal in the low state for the required time.

Table 44. TC8L Register (Group/Bank 0Dh, Register 4)

Bit	7	6	5	4	3	2	1	0
Bit/Field	T8_Level_LO							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210	T_8 Level Low Value	R/W	Data	Duration that T8_Out remains Low

T16 Control Register (CTR2)

The T16 Control Register, known as CTR2, controls the operation of the 16-bit T_{16} timer (see Table 45).

Table 45. CTR2 Register (Group/Bank 0Dh, Register 2)

Bit	7	6	5	4	3	2	1	0
Bit/Field	T16_ Enable	Single/ Mod-ulo-n	Time_ Out	T16_Clock		Capture INT_ Mask	Counter INT_ Mask	P41_ Out
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
7_____	T_{16} Enable	R	1	Counter Enabled
		R	0	Counter Disabled
		W	1	Enable Counter
		W	0	Stop Counter
_6_____	Single/ Modulo-n	R/W		In Transmit Mode:
			1	Single Pass
			0	Modulo-n
				In Demodulation Mode:
			1	T_{16} Does Not Recognize Edge
			0	T_{16} Recognizes Edge
__5_____	Time_Out	R	1	Counter Timeout Occurred
		R	0	No Counter Timeout
		W	1	Reset Flag to 0
		W	0	No Effect
___43___	T_{16} Clock	R/W	11	SCLK/8
			10	SCLK/4
			01	SCLK/2
			00	SCLK
_____2__	Capture Interrupt Mask	R/W	1	Enable Data Capture Interrupt
			0	Disable Data Capture Interrupt
______1_	Counter Interrupt Mask	R/W	1	Enable Time_Out Interrupt
			0	Disable Time_Out Interrupt
_______0	P41_Out	R/W	1	P41 configured as T_{16} Output
			0	P41 configured as I/O

T16 MS-Byte Capture Register (HI16)

The T16 MS-Byte Capture Register, as described in Table 46, holds the captured data from the output of the T_{16} counter/timer. This register holds the most significant byte of the data.

Table 46. HI16 Register (Group/Bank 0Dh, Register 9)

Bit	7	6	5	4	3	2	1	0
Bit/Field	T16_Capture_HI							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210	T_{16} Capture HI	R W	Data	MS-Byte of Captured Data No Effect

T16 LS-Byte Capture Register (LO16)

The T16 LS-Byte Capture Register, as described in Table 47, holds the captured data from the output of the T_{16} counter/timer. This register holds the least significant byte of the data.

Table 47. LO16 Register (Group/Bank 0Dh, Register 8)

Bit	7	6	5	4	3	2	1	0
Bit/Field	T16_Capture_LO							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210	T_{16} Capture LO	R W	Data	LS-Byte of Captured Data No Effect

T16 MS-Byte Load Register (TC16H)

The T16 MS-Byte Load Register, as described in Table 48, is loaded with the most significant byte of the T_{16} counter value.

Table 48. TC16H Register (Group/Bank 0Dh, Register 7)

Bit	7	6	5	4	3	2	1	0
Bit/Field	T16_Data_HI							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210	T_{16} Data HI	R/W	Data	MS-Byte of the T_{16} Counter

T16 LS-Byte Load Register (TC16L)

The T16 LS-Byte Load Register, as described in Table 49, is loaded with the least significant byte of the T_{16} counter value.

Table 49. TC16L Register (Group/Bank 0Dh, Register 6)

Bit	7	6	5	4	3	2	1	0
Bit/Field	T16_Data_LO							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210	T_{16} Data LO	R/W	Data	LS-Byte of the T_{16} Counter

Stop-Mode Recovery Control Registers

The Z86D99/Z86L99 family of products allows 16 individual I/O pins (Ports 2 and 5) to be used as a stop-mode recovery sources. The STOP mode is exited when one of these SMR sources is toggled.

Stop-Mode Recovery Register

The SMR register serves two functions. Bit D7 of the SMR register, as shown in Table 50, is the Stop Mode Flag that is set upon entering stop mode. A 0 in this bit indicates that the device has been reset by a POR or WDT Reset. A POR or WDT Reset is sometimes referred to as a “cold” start. A 1 in bit D7 indicates that the device was awakened by a SMR source. Waking a device with a SMR source is sometimes referred to as a “warm” start.

The Stop Mode Recovery source can be selected by any combination of P2 and P5 by P2SMR and P5SMR, respectively. If the pin is selected as the SMR source, its logic level is latched into a register. A wait up signal is generated if its logic level changes. This applies to all selected pins for the SMR source.

The comparators of P5 cannot be used as an SMR source. The comparator is turned off in STOP mode.

Table 50. SMR Register (Group/Bank 0Fh, Register B)

Bit	7	6	5	4	3	2	1	0
Bit/Field	Stop Flag	Re-served	Stop Delay	Reserved			SCLK Select	
R/W	R	R/W	W	R/W	R/W	R/W	W	W
Reset	0	0	1	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
7_____	Stop Mode Flag	R	1	Stop Recovery (warm start)
		R	0	POR/WDT Reset (cold start)
		W	X	No Effect
_6_____	Reserved	R	1	Always reads 1
		W	X	No Effect
__5_____	Stop Delay	R	1	Always reads 1
		W	1	Enable 5ms /Reset delay
		W	0	Disable /Reset delay after SMR
___432__	Reserved	R	1	Always reads 111
		W	X	No Effect
_____10	System Clock Select	R	11	Always reads 11
		W	11	SCLK, TCLK = XTAL/16
		W	10	SCLK, TCLK = XTAL
		W	01	SCLK, TCLK = XTAL/32
		W	00	SCLK, TCLK = XTAL/2

The second function of the SMR register is the selection of the external clock divide value. The purpose of this control is to selectively reduce device power

consumption during normal processor execution (SCLK control) and/or HALT mode (where TCLK sources counter/timers and interrupt logic).

Port 2 Stop Mode Recovery (P2SMR)

The P2SMR register, as described in Table 51, defines which I/O lines in Port 2 are to be used as stop mode recovery sources.

Table 51. P2SMR Register (Group/Bank 0Fh, Register 1)

Bit	7	6	5	4	3	2	1	0
Bit/Field	**P27RS**	**P26RS**	**P25RS**	**P24RS**	**P23RS**	**P22RS**	**P21RS**	**P20RS**
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210 (by bit)	Port 2 Stop Mode Recovery	R/W	1 0	Recovery Source Not

Port 5 Stop-Mode Recovery (P5SMR)

The P5SMR register, as described in Table 52, defines which I/O lines in Port 5 are to be used as stop-mode recovery sources.

Table 52. P5SMR Register (Group/Bank 0Fh, Register 5)

Bit	7	6	5	4	3	2	1	0
Bit/Field	**P57RS**	**P56RS**	**P55RS**	**P54RS**	**P53RS**	**P52RS**	**P51RS**	**P50RS**
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

R = Read, W = Write, X = Indeterminate

Bit Position	Bit/Field	R/W	Value	Description
76543210 (by bit)	Port 5 Stop Mode Recovery	R/W	1 0	Recovery Source Not

Electrical Characteristics

This section covers the absolute maximum ratings, standard test conditions, DC characteristics, and AC characteristics.

Absolute Maximum Ratings

Table 53 lists the absolute maximum ratings.

Table 53. Absolute Maximum Ratings

Symbol	Description	Min	Max	Units
V_{MAX}	Supply Voltage (*)	–0.3	+7.0	V
T_{STG}	Storage Temp.	–65°	+150°	C
T_A	Oper. Ambient Temp.		†	C
V_{RAM}	Minimum RAM Voltage	1.0 V**		

Note:
*Voltage on all pins with respect to GND.
†See “Ordering Information” on page 95.
** Estimated value, not tested.

Stresses greater than those listed in the preceding table can cause permanent damage to the device. This rating is a stress rating only. Functional operation of the device at any condition above those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for an extended period can affect device reliability.

Standard Test Conditions

The characteristics listed below apply for standard test conditions as noted. All voltages are referenced to GND. Positive current flows into the referenced pin (see Figure 36).

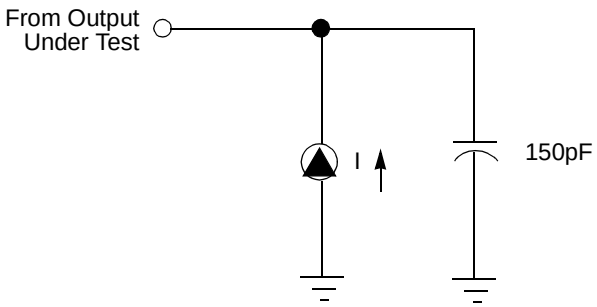


Figure 36. Test Load Diagram

DC Characteristics

Table 54 lists the DC characteristics for the Z86D99X (OTP only). Table 55 lists the DC characteristics for the Z86L99X (mask only).

Table 54. DC Characteristics for the Z86D99X (OTP Only)

Symbol	Parameter	V_{DD}	Min	Max	Units	Comments
V_{DD}	Power Supply Voltage		3	5.5		
V_{CH}	Clock Input High Voltage	3.0 V 5.5 V	0.8Vdd 0.8Vdd	Vdd+0.3 Vdd+0.3	V V	Driven by Ext. clock generator
V_{CL}	Clock Input Low Voltage	3.0 V 5.5 V	Vss–0.3 Vss–0.3	0.2Vdd 0.2Vdd		Driven by Ext. clock generator
V_{IH}	Input High Voltage	3.0 V 5.5 V	0.7Vdd 0.7Vdd	Vdd+0.3 Vdd+0.3	V V	
V_{IL}	Input Low Voltage	3.0 V 5.5 V	Vss–0.3 Vss–0.3	0.2Vdd 0.2Vdd	V V	
V_{OH1}	Output High Voltage Regular I/O	3.0 V 5.5 V	V_{DD}–0.8 V_{DD}–0.8		V V	–1.2 mA
V_{OH2}	High Drive Pins (P54, P55, P56, P57)	3.0 V 5.5 V	V_{DD}–0.8 V_{DD}–0.8		V V	–5.0 mA
V_{OL1}	Regular I/O Output low voltage	3.0 V 5.5 V		0.4 0.8	V V	2 mA 4.0 mA
V_{OL2}	High Drive Pins (P54, P55, P56, P57)	3.0 V 5.5 V		0.4 0.8	V V	4 mA 7.0 mA
I_{CCO}	Controlled Current Output (P43)	3.0 V 5.5 V	70 70	120 120	mA mA	Vout = 1.2 V to VDD (see Figure 17)
I_{IL}	Input Leakage	3.0 V 5.5 V	–1 –1	1 µA 1 µA	µA µA	Vin=0 V, Vdd Vin=0 V, Vdd
I_{CC}	Supply Current	3.0 V 5.5 V 3.0 V 5.5 V		10 15 250 850	mA mA µA µA	at 8 MHz at 8 MHz at 32 KHz at 32 KHz ADC is off.
I_{CC1}	Standby Current (Halt Mode)	3.0 V 5.5 V 3.0 V 5.5 V		3 5 2 4	mA mA mA mA	Vin=0 V, Vdd at 8 MHz Clock divided by 16 XTAL running ADC is off.
I_{CC2}	Standby Current (STOP Mode)	3.0 V 5.5 V		20 30	µA µA	Vin=0 V, Vdd; ADC is off. P43=1 or high impedance WDT, Comparators, Low Voltage Detection, and ADC (if applicable) are disabled. The IC might draw more current if any of the above peripherals is enabled.
I_{ADC}	Current with A/D Running	3.0 V 5.5 V		500 900	µA µA	
V_{LV}	Vdd Low-Voltage Protection			2.90	V	Low voltage protection is also known as brownout. Typical is 2.6 V.
V_{LB}	Low-Battery Detection			VLV+ 0.5	V V	

Table 55. DC Characteristics for the Z86L99X (Mask Only)

Symbol	Parameter	V_{DD}	Min	Max	Units	Comments
V_{DD}	Power Supply Voltage		2.3	5.5		
V_{CH}	Clock Input High Voltage	2.3 V 5.5 V	0.8Vdd 0.8Vdd	Vdd+0.3 Vdd+0.3	V V	Driven by Ext. clock generator
V_{CL}	Clock Input Low Voltage	2.3 V 5.5 V	Vss–0.3 Vss–0.3	0.2Vdd 0.2Vdd		Driven by Ext. clock generator
V_{IH}	Input High Voltage	2.3 V 5.5 V	0.7Vdd 0.7Vdd	Vdd+0.3 Vdd+0.3	V V	
V_{IL}	Input Low Voltage	2.3 V 5.5 V	Vss–0.3 Vss–0.3	0.2Vdd 0.2Vdd	V V	
V_{OH1}	Output High Voltage Regular I/O	2.3 V 5.5 V	2.0 5.0		V V	–0.5 mA
		2.3 V 5.5 V	1.9 5.0		V V	–1.2 mA
V_{OH2}	High Drive Pins (P54, P55, P56, P57)	2.3 V 5.5 V	1.9 5.1		V V	–3 mA
		2.3 V 5.5 V	1.7 4.7		V V	–5 mA
V_{OL1}	Regular I/O Output low voltage	2.3 V 5.5 V		0.4 V 0.4 V	V V	2 mA
		2.3 V 5.5 V		0.8 V 0.8 V	V V	4 mA
V_{OL2}	High Drive Pins (P54, P55, P56, P57)	2.3 V 5.5 V		0.4 V 0.4 V	V V	4 mA
		2.3 V 5.5 V		0.8 V 0.8 V	V V	7 mA
I_{CCO}	Controlled Current Output (P43)	2.3 V 5.5 V	70 70	120 120	mA mA	Vout = 1.2 V to VDD at room temperature (see Figure 17)
I_{IL}	Input Leakage	2.3 V 5.5 V	–1 –1	1 µA 1 µA	µA µA	Vin=0 V, Vdd Vin=0 V, Vdd
I_{CC}	Supply Current	2.3 V 5.5 V 2.3 V 5.5 V		3 8 250 850	mA mA µA µA	at 8 MHz at 8 MHz at 32 KHz at 32 KHz ADC is off.
I_{CC1}	Standby Current (Halt Mode)	2.3 V 5.5 V		2 5	mA mA	Vin=0 V, Vdd at 8 MHz
I_{CC2}	Standby Current (STOP Mode)	2.3 V 5.5 V		8 25.0	µA µA	Vin=0 V, Vdd;ADC is off. WDT, Comparators, Low Voltage Detection, and ADC (if applicable) are disabled. The IC might draw more current if any of the above peripherals is enabled.
I_{CC2}	Standby Current (STOP Mode)	5.5 V		15	µA	at 30 °C
I_{LV}	Standby Current (Low Voltage)			20	µA	Measured at V_{DD}=V_{LV}–0.2 V.
I_{ADC}	Current with A/D Running	2.3 V 5.5 V		500 900	µA µA	
V_{LV}	Vdd Low-Voltage Protection			2.2	V	Low voltage protection is also known as brownout. Typical is around 1.7 V at room temperature.
V_{LB}	Low-Battery Detection			3.0	V	Typical is around 2.4 V at room temperature.

Analog-to-Digital Converter Characteristics

Table 56 lists the analog-to-digital converter characteristics.

Table 56. Analog-to-Digital Converter Characteristics

Parameter	Minimum	Typical	Maximum	Units
Resolution		8		bits
Integral Nonlinearity		0.5	1	LSB
Differential Nonlinearity		0.5	1	LSB
Zero Error at 25 °C			7.8	mV
Supply Voltage Range (OTP)	3.0		5.5	V
Supply Voltage Range (ROM)	2.3		5.5	V
Power Dissipation (No Load)			1.2	mW
Clock Frequency (f ADC)			4	MHz
Input Voltage Range	V_{Ref-}		V_{Ref+}	V
Step Response			2/(0.0021 X f ADC)	s
ADC Input Capacitance	25		40	pF
Vref Input Capacitance	25		40	pF
V_{Ref+} Range	V_{Ref-}+2.0		AV_{DD}	V
V_{Ref-} Range	AGND		V_{Ref+}–2.0	V
$(V_{Ref+})–(V_{Ref-})$	2.0		AV_{DD}	V
Temperature Range	0		70	°C
3-db Frequency		(0.0021 X f ADC)		Hz
Signal to Noise	47			db
ADC Output Code		Dout		
Vref Input Source Impedance			1.0	kOhms
ADC Input Source Impedance			1.0	kOhms

Notes: Dout= $[(Vin–V_{Ref-})/(V_{Ref+}–V_{Ref-})]$ X 256
f ADC = set in ADCCTRL configuration register
Step Response is the time to track the input if a step from V_{Ref-} to V_{Ref+} is applied.

The ADC input is a switching capacitor that charges up to the applied input voltage whenever it is configured as an ADC input. If you switch it from digital mode to

the ADC input mode, the switching capacitor starts to charge up from 0 V. For the maximum swing (Dout = 0 to FF), it takes 2/(0.0021x f ADC). For an 8-MHz MCU crystal (with clock divide-by-two mode), the internal system clock is 4 MHz. In ADCCTRL, if you select the ADC frequency = system clock divided by 1 option, f ADC = 4 MHz. The step response = 238 uS.

AC Characteristics

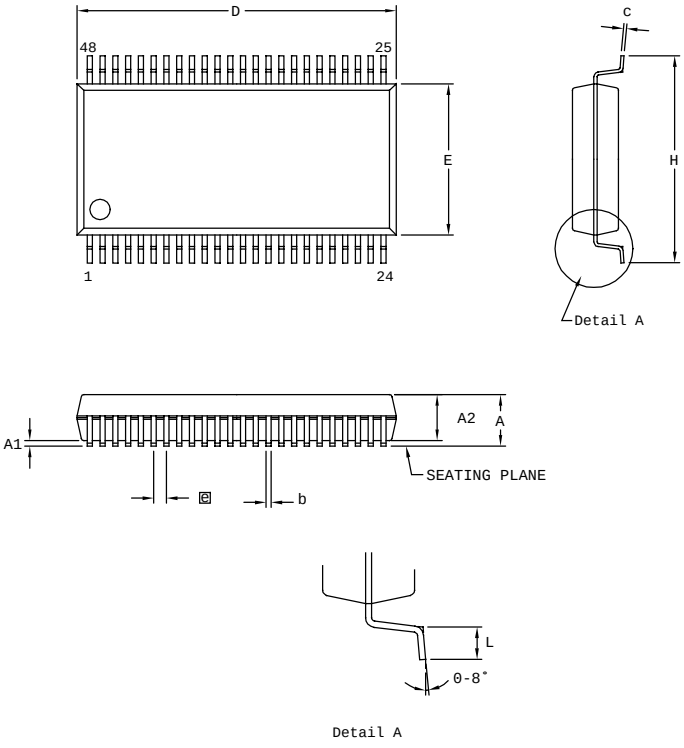
Table 57 lists the AC characteristics.

Table 57. AC Characteristics

No.	Symbol	Parameter	VDD	Min	Max	Units
1	TpC	Input Clock Period	2.3 V 5.5 V	120 120	DC DC	ns
2	TrC, TfC	Clock Input Rise and Fall Times	2.3 V 5.5 V		25 ns 25 ns	
3	TwC	Input Clock Width	2.3 V 5.5 V	5.0 5.0		ns ns
4	TwTinL	Timer Input Low Width	2.3 V 5.5 V	2TPC 2TPC		 ns
5	TwTinH	Timer Input High Width	2.3 V 5.5 V	2 2		TpC TpC
6	TpT1in	Timer 1 Input Period	2.3 V 5.5 V	8 8		TpC TpC
7	TrTin, TfTin	Timer Input Rise and Fall Time	2.3 V 5.5 V		100 100	ns ns
8	TwIL	Interrupt Request Low Time	2.3 V 5.5 V	100 70		ns ns
9	TwIH	Interrupt Request Input High Time	2.3 V 5.5 V	5 5		TpC TpC
10	Twsm	Stop-Mode Recovery Width Spec	2.3 V 5.5 V	12 12		ns ns
12	Twdt	Watch-Dog Timer Time Out	2.3 V 5.5 V	25 10		ms ms

Packaging

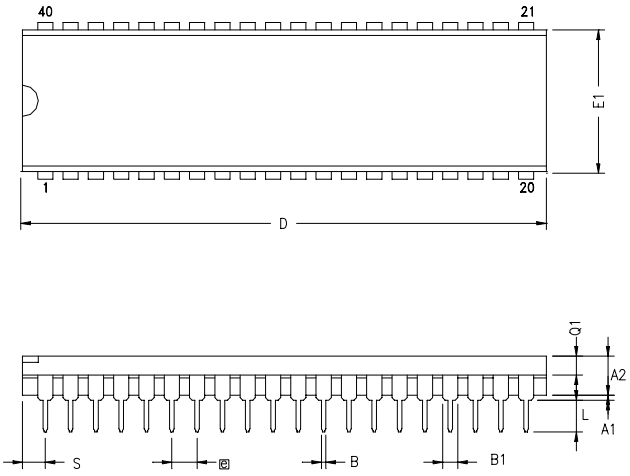
Figure 37 through Figure 40 show the available packages.



SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	2.41	2.79	0.095	0.110
A1	0.23	0.38	0.009	0.015
A2	2.18	2.39	0.086	0.094
b	0.20	0.34	0.008	0.0135
c	0.13	0.25	0.005	0.010
D	15.75	16.00	0.620	0.630
E	7.39	7.59	0.291	0.299
e	0.635 BSC		0.025 BSC	
H	10.16	10.41	0.400	0.410
L	0.51	1.016	0.020	0.040

CONTROLLING DIMENSIONS : MM
LEADS ARE COPLANAR WITHIN .004 INCH

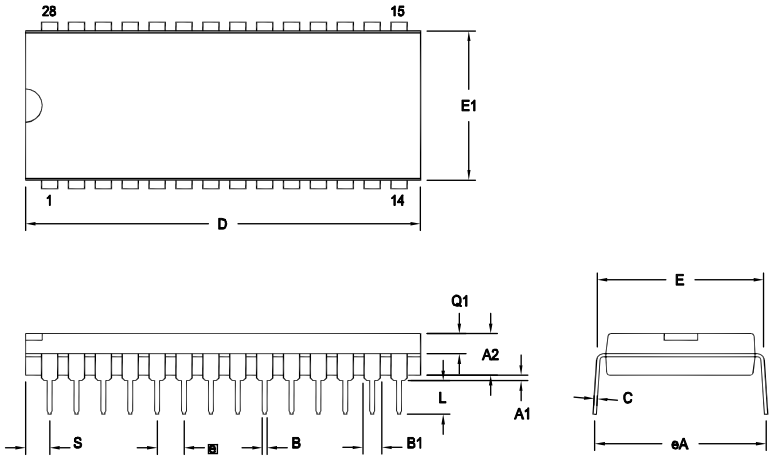
Figure 37. 48-Pin SSOP



SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A1	0.51	1.02	.020	.040
A2	3.18	3.94	.125	.155
B	0.38	0.53	.015	.021
B1	1.02	1.52	.040	.060
C	0.23	0.38	.009	.015
D	52.07	52.58	2.050	2.070
E	15.24	15.75	.600	.620
E1	13.59	14.22	.535	.560
e	2.54 TYP		.100 TYP	
eA	15.49	16.76	.610	.660
L	3.05	3.81	.120	.150
Q1	1.40	1.91	.055	.075
S	1.52	2.29	.060	.090

CONTROLLING DIMENSIONS : INCH

Figure 38. 40-Pin PDIP



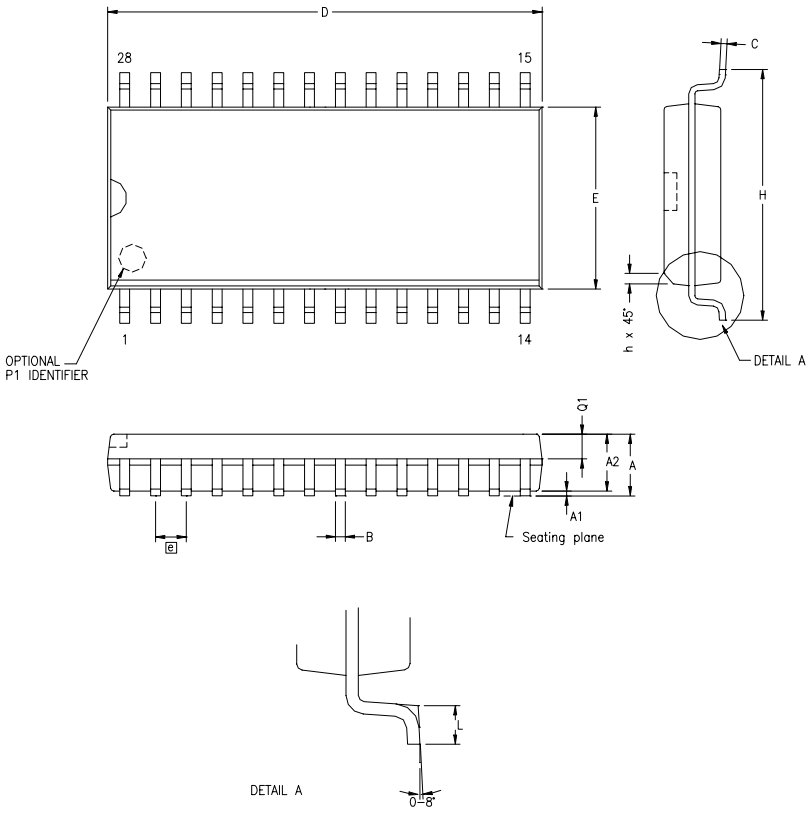
SYMBOL	OPT #	MILLIMETER		INCH	
		MIN	MAX	MIN	MAX
A1		0.38	1.02	.015	.040
A2		3.18	4.19	.125	.165
B		0.38	0.53	.015	.021
B1	01	1.40	1.65	.055	.065
	02	1.14	1.40	.045	.055
C		0.23	0.38	.009	.015
D	01	36.58	37.34	1.440	1.470
	02	35.31	35.94	1.390	1.415
E		15.24	15.75	.600	.620
E1	01	13.59	14.10	.535	.555
	02	12.83	13.08	.505	.515
e		2.54 TYP		.100 BSC	
eA		15.49	16.76	.610	.660
L		3.05	3.81	.120	.150
Q1	01	1.40	1.91	.055	.075
	02	1.40	1.78	.055	.070
S	01	1.52	2.29	.060	.090
	02	1.02	1.52	.040	.060

CONTROLLING DIMENSIONS : INCH

OPTION TABLE	
OPTION #	PACKAGE
01	STANDARD
02	IDF

Note: ZiLOG supplies both options for production. Component layout PCB design should cover bigger option 01.

Figure 39. 28-Pin PDIP



SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	2.40	2.64	.094	.104
A1	0.10	0.30	.004	.012
A2	2.24	2.44	.088	.096
B	0.36	0.46	.014	.018
C	0.23	0.30	.009	.012
D	17.78	18.00	.700	.710
E	7.40	7.60	.291	.299
e	1.27 BSC		.050 BSC	
H	10.00	10.65	.394	.419
h	0.30	0.71	.012	.028
L	0.61	1.00	.024	.039
Q1	0.97	1.09	.038	.043

CONTROLLING DIMENSIONS : MM
LEADS ARE COPLANAR WITHIN .004 INCH.

Figure 40. 28-Pin SOIC

Design Considerations

The Z8 uses a Pierce oscillator with an internal feedback circuit. The advantages of this circuit are low cost, large output signal, low-power level in the crystal, stability with respect to V_{CC} and temperature, and low impedances (not disturbed by stray effects.)

One drawback is the requirement for high gain in the amplifier to compensate for feedback path losses. Traces connecting crystal, capacitors, and the Z8 oscillator pins must be as short and wide as possible. Short and wide traces reduce parasitic inductance and resistance. The components (capacitors, crystal, and resistors) must be placed as close as possible to the oscillator pins of the Z8.

The traces from the oscillator pins of the integrated circuit (IC) and the ground side of the lead capacitors must be guarded from all other traces (clock, V_{CC}, and system ground) to reduce cross-talk and noise injection. Guarding the traces is usually accomplished by keeping other traces and system ground trace planes away from the oscillator circuit and by placing a Z8 device V_{SS} ground ring around the traces/components. The ground side of the oscillator lead capacitors must be connected to a single trace to the Z8 V_{SS} (GND) pin. It must not be shared with any other system ground trace or components except at the Z8 device V_{SS} pin. Not sharing the ground side of the oscillator lead capacitors is to prevent differential system ground noise injection into the oscillator.

Ordering Information

Part	PSI	Description
Z86D99 (OTP)	Z86D990PZ008SC	40-pin PDIP
	Z86D990HZ008SC	48-pin SSOP
	Z86D991PZ008SC	28-pin PDIP
	Z86D991SZ008SC	28-pin SOIC
Z86L99 (Mask ROM)	Z86L990PZ008SC	40-pin PDIP
	Z86L990HZ008SC	48-pin SSOP
	Z86L991PZ008SC	28-pin PDIP
	Z86L991SZ008SC	28-pin SOIC
	Z86L996PZ008SC	28-pin PDIP
	Z86L996SZ008SC	28-pin SOIC
	Z86L997PZ008SC	28-pin PDIP
	Z86L997SZ008SC	28-pin SOIC
Emulator	Z86L9900100ZEM	Emulator/Programmer
Adapter	Z86D9900100ZDH	48 SSOP Adapter
Evaluation Board	Z86L9900100ZCO	Evaluation Board

For fast results, contact your local ZiLOG sale offices for assistance in ordering part(s). Updated information can be found on the ZiLOG website: HTTP://WWW.ZILOG.COM

Precharacterization Product

The product represented by this document is newly introduced and ZiLOG has not completed the full characterization of the product. The document states what ZiLOG knows about this product at this time, but additional features or nonconformance with some aspects of the document might be found, either by ZiLOG or its customers in the course of further application and characterization work. In addition, ZiLOG cautions that delivery might be uncertain at times, due to start-up yield issues.

ZiLOG, Inc.

532 Race Street

San Jose, CA 95126-3432

Telephone: (408) 558-8500

FAX: 408 558-8300

Internet: HTTP://WWW.ZILOG.COM