



Welcome to [E-XFL.COM](https://www.e-xfl.com)

What is "[Embedded - Microcontrollers](#)"?

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Obsolete
Core Processor	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	I ² C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	46
Program Memory Size	24KB (24K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 12x10b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	68-LCC (J-Lead)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f2402vs020sc00tr



Timer Operating Modes	58
Reading the Timer Count Values	66
Timer Output Signal Operation	66
Timer Control Register Definitions	66
Timer 0-3 High and Low Byte Registers	66
Timer Reload High and Low Byte Registers	67
Timer 0-3 PWM High and Low Byte Registers	69
Timer 0-3 Control Registers	70
Watch-Dog Timer	72
Overview	72
Operation	72
Watch-Dog Timer Refresh	73
Watch-Dog Timer Time-Out Response	73
Watch-Dog Timer Reload Unlock Sequence	74
Watch-Dog Timer Control Register Definitions	75
Watch-Dog Timer Control Register	75
Watch-Dog Timer Reload Upper, High and Low Byte Registers ..	76
UART	78
Overview	78
Architecture	78
Operation	79
Data Format	79
Transmitting Data using the Polled Method	80
Transmitting Data using the Interrupt-Driven Method	81
Receiving Data using the Polled Method	82
Receiving Data using the Interrupt-Driven Method	82
Receiving Data using the Direct Memory Access Controller (DMA)	83
Multiprocessor (9-bit) Mode	84
UART Interrupts	85
UART Baud Rate Generator	85
UART Control Register Definitions	86
UARTx Transmit Data Register	86
UARTx Receive Data Register	87
UARTx Status 0 and Status 1 Registers	87
UARTx Control 0 and Control 1 Registers	89
UARTx Baud Rate High and Low Byte Registers	91
Infrared Encoder/Decoder	95
Overview	95
Architecture	95
Operation	96



Transmitting IrDA Data	96
Receiving IrDA Data	97
Jitter	98
Infrared Encoder/Decoder Control Register Definitions	98
Serial Peripheral Interface	99
Overview	99
Architecture	99
Operation	100
SPI Signals	101
SPI Clock Phase and Polarity Control	102
Multi-Master Operation	104
Error Detection	105
SPI Interrupts	105
SPI Baud Rate Generator	105
SPI Control Register Definitions	106
SPI Data Register	106
SPI Control Register	107
SPI Status Register	108
SPI Mode Register	109
SPI Baud Rate High and Low Byte Registers	110
I2C Controller	111
Overview	111
Operation	111
SDA and SCL Signals	111
I ² C Interrupts	112
Start and Stop Conditions	112
Writing a Transaction with a 7-Bit Address	112
Writing a Transaction with a 10-Bit Address	114
Reading a Transaction with a 7-Bit Address	115
Reading a Transaction with a 10-Bit Address	116
I2C Control Register Definitions	118
I2C Data Register	118
I2C Status Register	118
I2C Control Register	119
I2C Baud Rate High and Low Byte Registers	121
Direct Memory Access Controller	122
Overview	122
Operation	122
DMA0 and DMA1 Operation	122
Configuring DMA0 and DMA1 for Data Transfer	123



Table 32.	IRQ2 Enable and Priority Encoding	53
Table 33.	IRQ1 Enable High Bit Register (IRQ1ENH)	53
Table 34.	IRQ2 Enable Low Bit Register (IRQ2ENL)	54
Table 35.	IRQ2 Enable High Bit Register (IRQ2ENH)	54
Table 36.	Interrupt Edge Select Register (IRQES)	55
Table 37.	Interrupt Port Select Register (IRQPS)	55
Table 38.	Interrupt Control Register (IRQCTL)	56
Table 39.	Timer 0-3 High Byte Register (TxH)	67
Table 40.	Timer 0-3 Low Byte Register (TxL)	67
Table 41.	Timer 0-3 Reload High Byte Register (TxRH)	68
Table 42.	Timer 0-3 Reload Low Byte Register (TxRL)	68
Table 43.	Timer 0-3 PWM High Byte Register (TxPWMH)	69
Table 44.	Timer 0-3 PWM Low Byte Register (TxPWML)	69
Table 45.	Timer 0-3 Control Register (TxCTL)	70
Table 46.	Watch-Dog Timer Approximate Time-Out Delays	73
Table 47.	Watch-Dog Timer Control Register (WDTCTL)	75
Table 48.	Watch-Dog Timer Reload Upper Byte Register (WDTU)	76
Table 49.	Watch-Dog Timer Reload High Byte Register (WDTH)	76
Table 50.	Watch-Dog Timer Reload Low Byte Register (WDTL)	77
Table 51.	UARTx Transmit Data Register (UxTXD)	86
Table 52.	UARTx Receive Data Register (UxRXD)	87
Table 53.	UARTx Status 0 Register (UxSTAT0)	87
Table 54.	UARTx Control 0 Register (UxCTL0)	89
Table 55.	UARTx Status 1 Register (UxSTAT1)	89
Table 56.	UARTx Control 1 Register (UxCTL1)	90
Table 57.	UARTx Baud Rate High Byte Register (UxBRH)	91
Table 58.	UARTx Baud Rate Low Byte Register (UxBRL)	92
Table 59.	UART Baud Rates	93
Table 60.	SPI Clock Phase (PHASE) and Clock Polarity (CLKPOL) Operation	102
Table 61.	SPI Data Register (SPIDATA)	106
Table 62.	SPI Control Register (SPICTL)	107
Table 63.	SPI Status Register (SPISTAT)	108
Table 64.	SPI Mode Register (SPIMODE)	109
Table 65.	SPI Baud Rate High Byte Register (SPIBRH)	110
Table 66.	SPI Baud Rate Low Byte Register (SPIBRL)	110



Table 67.	I2C Data Register (I2CDATA)	118
Table 68.	I2C Status Register (I2CSTAT)	118
Table 69.	I2C Control Register (I2CCTL)	119
Table 70.	I2C Baud Rate High Byte Register (I2CBRH)	121
Table 71.	I2C Baud Rate Low Byte Register (I2CBRL)	121
Table 72.	DMAx Control Register (DMAxCTL)	124
Table 73.	DMAx I/O Address Register (DMAxIO)	126
Table 74.	DMAx Address High Nibble Register (DMAxH)	126
Table 75.	DMAx End Address Low Byte Register (DMAxEND) .	128
Table 76.	DMAx Start/Current Address Low Byte Register (DMAxSTART)	128
Table 77.	DMA_ADC Register File Address Example	129
Table 78.	DMA_ADC Address Register (DMAA_ADDR)	129
Table 79.	DMA_ADC Control Register (DMAACTL)	130
Table 80.	DMA_ADC Status Register (DMAA_STAT)	131
Table 81.	ADC Control Register (ADCCTL)	135
Table 82.	ADC Data High Byte Register (ADCD_H)	137
Table 83.	ADC Data Low Bits Register (ADCD_L)	137
Table 84.	Z8F640x family Flash Memory Configurations	138
Table 85.	Flash Code Protection Using the Option Bits	142
Table 86.	Flash Control Register (FCTL)	144
Table 87.	Flash Status Register (FSTAT)	145
Table 88.	Flash Page Select Register (FPS)	146
Table 89.	Flash Frequency High Byte Register (FFREQH)	147
Table 90.	Flash Frequency Low Byte Register (FFREQL)	147
Table 91.	Option Bits At Program Memory Address 0000H	149
Table 92.	Options Bits at Program Memory Address 0001H	150
Table 93.	OCD Baud-Rate Limits	154
Table 94.	On-Chip Debugger Commands	156
Table 95.	OCD Control Register (OCDCTL)	161
Table 96.	OCD Status Register (OCDSTAT)	162
Table 97.	OCD Watchpoint Control/Address (WPTCTL)	163
Table 98.	OCD Watchpoint Address (WPTADDR)	164
Table 99.	OCD Watchpoint Data (WPTDATA)	164
Table 100.	Recommended Crystal Oscillator Specifications (20MHz Operation)	166



Signal and Pin Descriptions

Overview

The Z8F640x family products are available in a variety of packages styles and pin configurations. This chapter describes the signals and available pin configurations for each of the package styles. For information regarding the physical package specifications, please refer to the chapter Packaging on page 206.

Available Packages

Table 2 identifies the package styles that are available for each device within the Z8F640x family product line.

Table 2. Z8F640x family Package Options

Part Number	40-pin PDIP	44-pin LQFP	44-pin PLCC	64-pin LQFP	68-pin PLCC	80-pin QFP
Z8F1601	X	X	X			
Z8F1602				X	X	
Z8F2401	X	X	X			
Z8F2402				X	X	
Z8F3201	X	X	X			
Z8F3202				X	X	
Z8F4801	X	X	X			
Z8F4802				X	X	
Z8F4803						X
Z8F6401	X	X	X			
Z8F6402				X	X	
Z8F6403						X



Table 2. Signal Descriptions (Continued)

Signal Mnemonic	I/O	Description
Reset		
$\overline{\text{RESET}}$	I	RESET. Generates a Reset when asserted (driven Low).
Power Supply		
VDD	I	Power Supply.
AVDD	I	Analog Power Supply.
VSS	I	Ground.
AVSS	I	Analog Ground.

Pin Characteristics

Table 3 provides detailed information on the characteristics for each pin available on the Z8F640x family products. Data in Table 3 is sorted alphabetically by the pin symbol mnemonic.

Table 3. Pin Characteristics of the Z8F640x family

Symbol Mnemonic	Direction	Reset Direction	Active Low or Active High	Tri-State Output	Internal Pull-up or Pull-down	Schmitt Trigger Input	Open Drain Output
AVSS	N/A	N/A	N/A	N/A	No	No	N/A
AVDD	N/A	N/A	N/A	N/A	No	No	N/A
DBG	I/O	I	N/A	Yes	No	Yes	Yes
VSS	N/A	N/A	N/A	N/A	No	No	N/A
PA[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
PB[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
PC[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
PD[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
PE[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable

x represents integer 0, 1,... to indicate multiple pins with symbol mnemonics that differ only by the integer

Architecture

Figure 65 illustrates a block diagram of the interrupt controller.

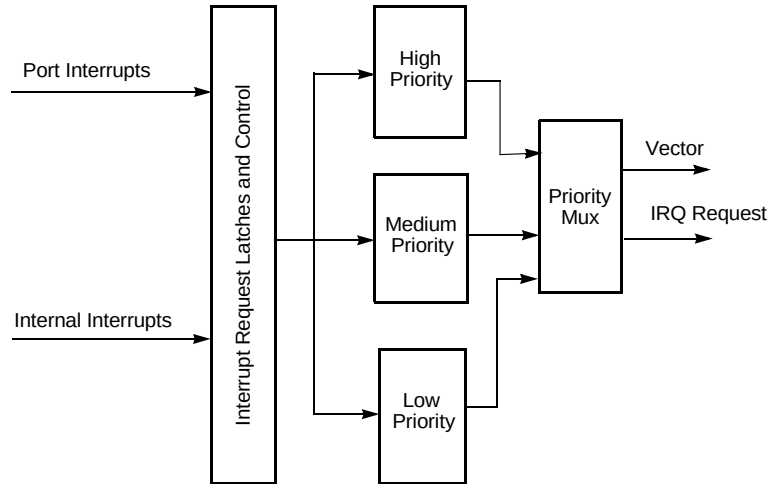


Figure 65. Interrupt Controller Block Diagram

Operation

Master Interrupt Enable

The master interrupt enable bit (IRQE) in the Interrupt Control register globally enables and disables interrupts.

Interrupts are globally enabled by any of the following actions:

- Execution of an EI (Enable Interrupt) instruction
- Execution of an IRET (Return from Interrupt) instruction
- Writing a 1 to the IRQE bit in the Interrupt Control register

Interrupts are globally disabled by any of the following actions:

- Execution of a DI (Disable Interrupt) instruction
- eZ8 CPU acknowledgement of an interrupt service request from the interrupt controller
- Writing a 0 to the IRQE bit in the Interrupt Control register
- Reset



mode. Refer to the **Reset and Stop Mode Recovery** chapter for more information on STOP Mode Recovery.

If interrupts are enabled, following completion of the Stop Mode Recovery the eZ8 CPU responds to the interrupt request by fetching the Watch-Dog Timer interrupt vector and executing code from the vector address.

WDT Reset in Normal Operation

If configured to generate a Reset when a time-out occurs, the Watch-Dog Timer forces the Z8F640x family device into the Short Reset state. The WDT status bit in the Watch-Dog Timer Control register is set to 1. Refer to the **Reset and Stop Mode Recovery** chapter for more information on Short Reset.

WDT Reset in Stop Mode

If configured to generate a Reset when a time-out occurs and the Z8F640x family device is in STOP mode, the Watch-Dog Timer initiates a Stop Mode Recovery. Both the WDT status bit and the STOP bit in the Watch-Dog Timer Control register are set to 1 following WDT time-out in STOP mode. Refer to the **Reset and Stop Mode Recovery** chapter for more information.

Watch-Dog Timer Reload Unlock Sequence

Writing the unlock sequence to the Watch-Dog Timer Control register (WDTCTL) unlocks the three Watch-Dog Timer Reload Byte registers (WDTU, WDTH, and WDTL) to allow changes to the time-out period. These write operations to the WDTCTL register address produce no effect on the bits in the WDTCTL register. The locking mechanism prevents spurious writes to the Reload registers. The follow sequence is required to unlock the Watch-Dog Timer Reload Byte registers (WDTU, WDTH, and WDTL) for write access.

1. Write 55H to the Watch-Dog Timer Control register (WDTCTL)
2. Write AAH to the Watch-Dog Timer Control register (WDTCTL)
3. Write the Watch-Dog Timer Reload Upper Byte register (WDTU)
4. Write the Watch-Dog Timer Reload High Byte register (WDTH)
5. Write the Watch-Dog Timer Reload Low Byte register (WDTL)

All three Watch-Dog Timer Reload registers must be written in the order just listed. There must be no other register writes between each of these operations. If a register write occurs, the lock state machine resets and no further writes can occur, unless the sequence is restarted. The value in the Watch-Dog Timer Reload registers is loaded into the counter when the Watch-Dog Timer is first enabled and every time a WDT instruction is executed.

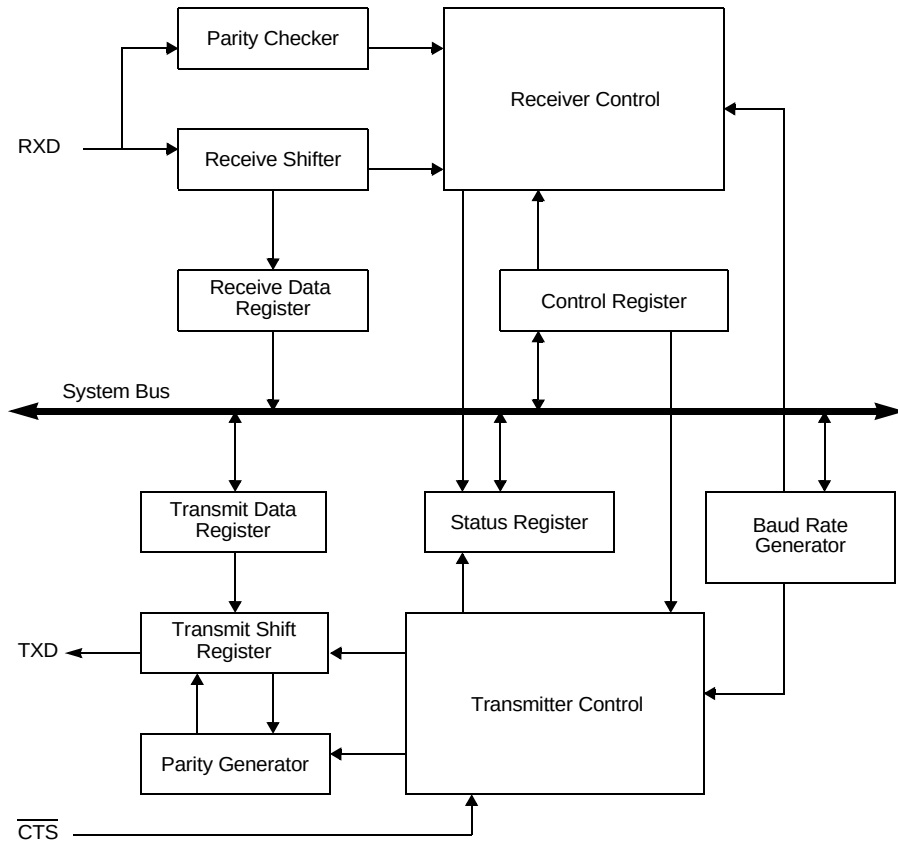


Figure 67. UART Block Diagram

Operation

Data Format

The UART always transmits and receives data in an 8-bit data format, least-significant bit first. An even or odd parity bit can be optionally added to the data stream. Each character begins with an active Low Start bit and ends with either 1 or 2 active High Stop bits.



Infrared Encoder/Decoder

Overview

The Z8F640x family products contain two fully-functional, high-performance UART to Infrared Encoder/Decoders (Endecs). Each Infrared Endec is integrated with an on-chip UART to allow easy communication between the Z8F640x family device and IrDA Physical Layer Specification Version 1.3-compliant infrared transceivers. Infrared communication provides secure, reliable, low-cost, point-to-point communication between PCs, PDAs, cell phones, printers and other infrared enabled devices.

Architecture

Figure 71 illustrates the architecture of the Infrared Endec.

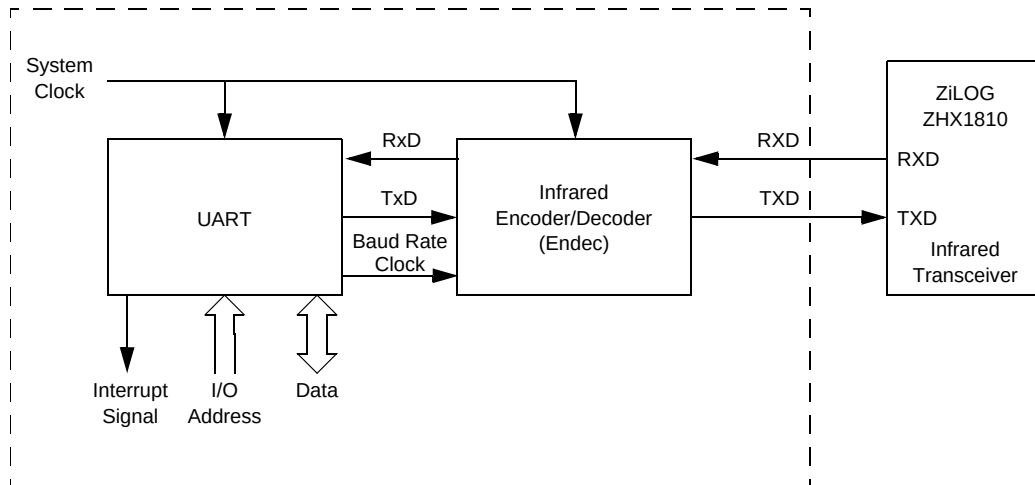


Figure 71. Infrared Data Communication System Block Diagram

Transfer Format PHASE Equals Zero

Figure 77 illustrates the timing diagram for an SPI transfer in which PHASE is cleared to 0. The two SCK waveforms show polarity with CLKPOL reset to 0 and with CLKPOL set to one. The diagram may be interpreted as either a Master or Slave timing diagram since the SCK Master-In/Slave-Out (MISO) and Master-Out/Slave-In (MOSI) pins are directly connected between the Master and the Slave.

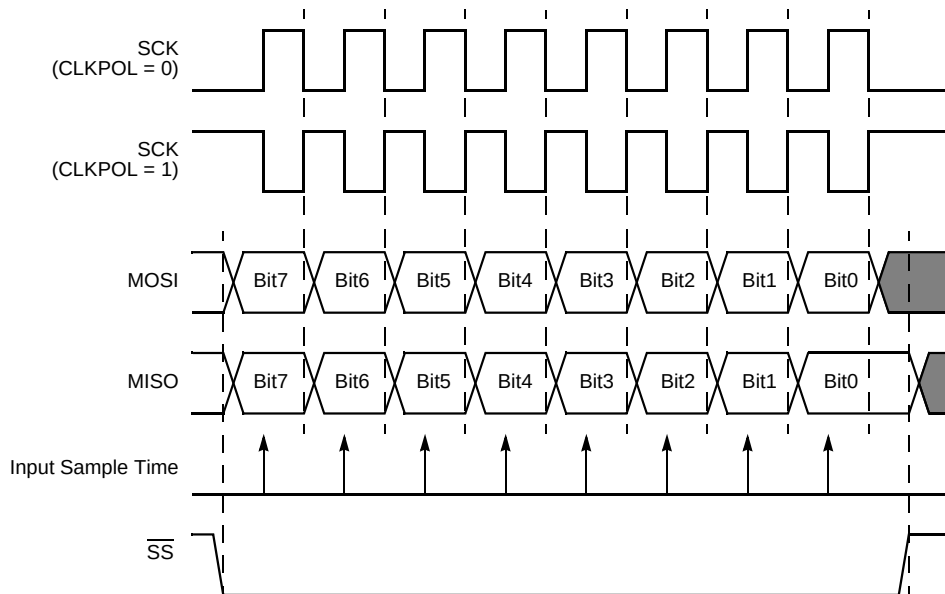


Figure 77. SPI Timing When PHASE is 0

Transfer Format PHASE Equals One

Figure 78 illustrates the timing diagram for an SPI transfer in which PHASE is one. Two waveforms are depicted for SCK, one for CLKPOL reset to 0 and another for CLKPOL set to 1.

1. Disable the SPI by clearing the `SPIEN` bit in the SPI Control register to 0.
2. Load the desired 16-bit count value into the SPI Baud Rate High and Low Byte registers.
3. Enable the Baud Rate Generator timer function and associated interrupt by setting the `BIRQ` bit in the SPI Control register to 1.

SPI Control Register Definitions

SPI Data Register

The SPI Data register stores both the outgoing (transmit) data and the incoming (received) data. Reads from the SPI Data register always return the current contents of the 8-bit shift register.

With the SPI configured as a Master, writing a data byte to this register initiates the data transmission. With the SPI configured as a Slave, writing a data byte to this register loads the shift register in preparation for the next data transfer with the external Master. In either the Master or Slave modes, if a transmission is already in progress, writes to this register are ignored and the Overrun error flag, `OVR`, is set in the SPI Status register.

When the character length is less than 8 bits (as set by the `NUMBITS` field in the SPI Mode register), the transmit character must be left justified in the SPI Data register. A received character of less than 8 bits will be right justified. For example, if the SPI is configured for 4-bit characters, the transmit characters must be written to `SPIDATA[7:4]` and the received characters are read from `SPIDATA[3:0]`.

Table 60. SPI Data Register (SPIDATA)

BITS	7	6	5	4	3	2	1	0
FIELD	DATA							
RESET	X	X	X	X	X	X	X	X
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	F60H							

DATA—Data
Transmit and/or receive data.

When the DMA is configured for two-byte word transfers, the DMAx I/O Address register must contain an even numbered address.

Table 72. DMAx I/O Address Register (DMAxIO)

BITS	7	6	5	4	3	2	1	0
FIELD	DMA_IO							
RESET	X	X	X	X	X	X	X	X
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	FB1H, FB9H							

DMA_IO—DMA on-chip peripheral control register address

This byte sets the low byte of the on-chip peripheral control register address on Register File Page FH (addresses F00H to FFFH).

DMAx Address High Nibble Register

The DMAx Address High register specifies the upper four bits of address for the Start/Current and End Addresses of DMAx.

Table 73. DMAx Address High Nibble Register (DMAxH)

BITS	7	6	5	4	3	2	1	0
FIELD	DMA_END_H				DMA_START_H			
RESET	X	X	X	X	X	X	X	X
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	FB2H, FHAH							

DMA_END_H—DMAx End Address High Nibble

These bits, used with the DMAx End Address Low register, form a 12-bit End Address. The full 12-bit address is given by {DMA_END_H[3:0], DMA_END[7:0]}.

DMA_START_H—DMAx Start/Current Address High Nibble

These bits, used with the DMAx Start/Current Address Low register, form a 12-bit Start/Current Address. The full 12-bit address is given by {DMA_START_H[3:0], DMA_START[7:0]}.

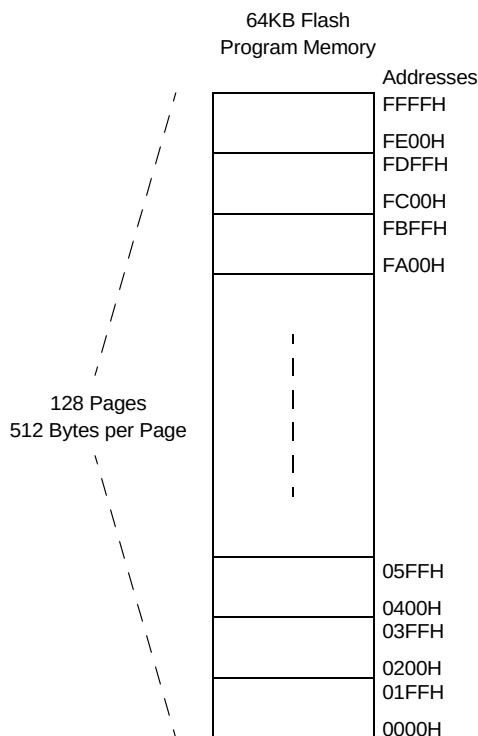


Figure 84. Flash Memory Arrangement

Operation

The Flash Controller programs and erases the Flash memory. The Flash Controller provides the proper Flash controls and timing for byte programming, Page Erase, and Mass Erase of the Flash memory. The Flash Controller contains a protection mechanism, via the Flash Control register (FCTL) to prevent accidental programming or erasure. The Flow Chart in Figure 85 illustrates basic Flash Controller operation. The following subsections provide details on the various operations (Lock, Unlock, Byte Programming, Page Erase, and Mass Erase) listed in Figure 85.

Flash Control Register Definitions

Flash Control Register

The Flash Controller must be unlocked via the Flash Control register before programming or erasing the Flash memory. Writing the sequence 73H 8CH, sequentially, to the Flash Control register unlocks the Flash Controller. When the Flash Controller is unlocked, writing to the Flash Control register can initiate either Page Erase or Mass Erase of the Flash memory. Writing an invalid value or an invalid sequence returns the Flash Controller to its locked state. The Write-only Flash Control Register shares its Register File address with the Read-only Flash Status Register.

Table 85. Flash Control Register (FCTL)

BITS	7	6	5	4	3	2	1	0
FIELD	FCMD							
RESET	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W
ADDR	FF8H							

FCMD—Flash Command

73H = First unlock command.

8CH = Second unlock command.

95H = Page erase command (must be third command in sequence to initiate Page Erase).

63H = Mass erase command (must be third command in sequence to initiate Mass Erase).

RPEN—Read Protect Option Bit Enabled

0 = The Read Protect Option Bit is disabled (1).

0 = The Read Protect Option Bit is enabled (0), disabling many OCD commands.

Reserved

These bits are always 0.

OCD Watchpoint Control Register

The OCD Watchpoint Control register is used to configure the debug Watchpoint.

Table 96. OCD Watchpoint Control/Address (WPTCTL)

BITS	7	6	5	4	3	2	1	0
FIELD	WPW	WPR	WPDM	Reserved	WPTADDR[11:8]			
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

WPW—Watchpoint Break on Write

This bit cannot be set if the Read Protect Option Bit is enabled.

0 = Watchpoint Break on Register File write is disabled.

1 = Watchpoint Break on Register File write is enabled.

WPR—Watchpoint Break on Read

This bit cannot be set if the Read Protect Option Bit is enabled.

0 = Watchpoint Break on Register File read is disabled.

1 = Watchpoint Break on Register File write is enabled.

WPDM—Watchpoint Data Match

If this bit is set, then the Watchpoint only generates a Debug Break if the data being read or written matches the specified Watchpoint data. Either the WPR and/or WPW bits must also be set for this bit to affect operation. This bit cannot be set if the Read Protect Option Bit is enabled.

0 = Watchpoint Break on read and/or write does not require a data match.

1 = Watchpoint Break on read and/or write requires a data match.

Reserved

This bit is reserved and must be 0.

RADDR[11:8]—Register address

These bits specify the upper 4 bits of the Register File address to match when generating a Watchpoint Debug Break. The full 12-bit Register File address is given by {WPTCTL3:0}, WPTADDR[7:0]}.



Packaging

Figure 103 illustrates the 40-pin PDIP (plastic dual-inline package) available for the Z8F1601, Z8F2401, Z8F3201, Z8F4801, and Z8F6401 devices.

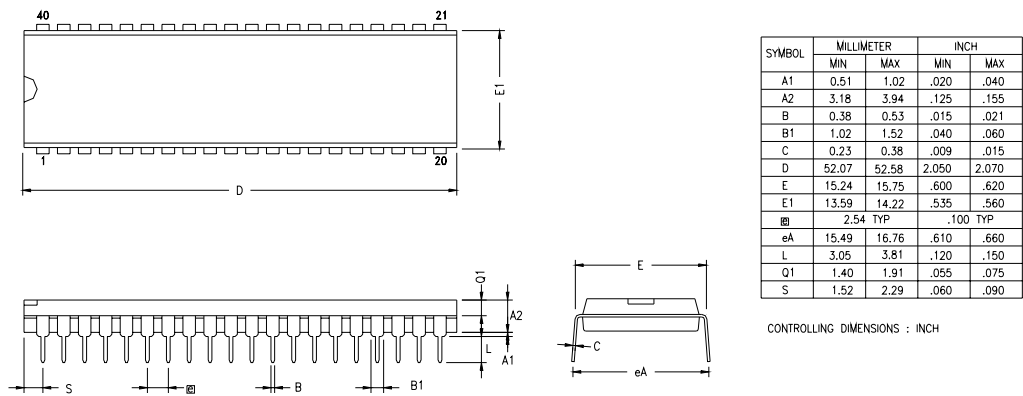


Figure 103. 40-Lead Plastic Dual-Inline Package (PDIP)

Figure 105 illustrates the 64-pin LQFP (low-profile quad flat package) available for the Z8F1602, Z8F2402, Z8F3202, Z8F4802, and Z8F6402 devices.

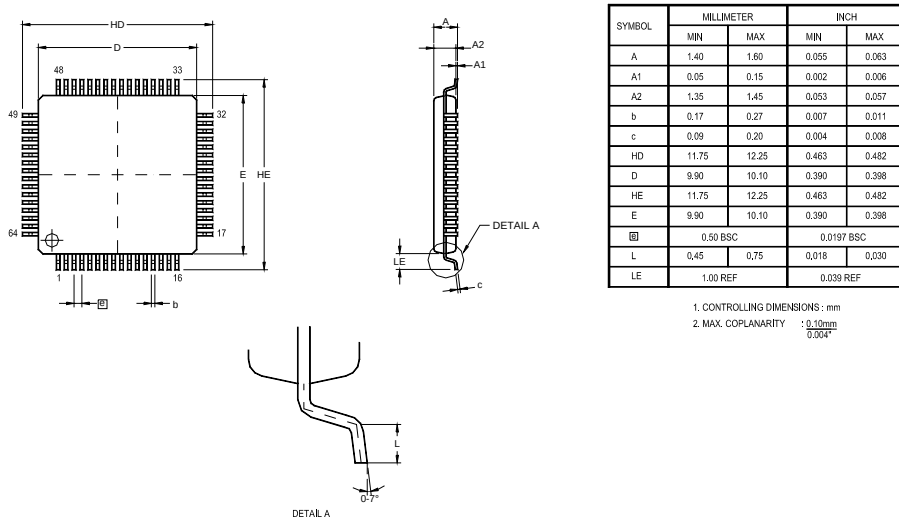


Figure 106. 64-Lead Low-Profile Quad Flat Package (LQFP)



- read watchpoint (21H) 161
- step instruction (10H) 160
- stuff instruction (11H) 160
- write data memory (0CH) 159
- write OCD control register (04H) 158
- write program counter (06H) 158
- write program memory (0AH) 159
- write register (08H) 158
- write watchpoint (20H) 161
- on-chip debugger 5
- on-chip debugger (OCD) 151
- on-chip debugger signals 14
- on-chip oscillator 165
- one-shot mode 70
- opcode map
 - abbreviations 203
 - cell description 202
 - first 204
 - second after 1FH 205
- OR 190
- ordering information 211
- ORX 190
- oscillator signals 14

P

- p 184
- packaging
 - LQFP
 - 44 lead 207
 - 64 lead 208
 - PDIP 206
 - PLCC
 - 44 lead 207
 - 68 lead 209
 - QFP 210
- part number description 214
- part selection guide 2
- PC 185
- PDIP 206
- peripheral AC and DC electrical characteristics 173
- PHASE=0 timing (SPI) 103
- PHASE=1 timing (SPI) 104
- pin characteristics 15

- PLCC
 - 44 lead 207
 - 68-lead 209
- polarity 184
- POP 189
- pop using extended addressing 189
- POPX 189
- port availability, device 33
- port input timing (GPIO) 176
- port output timing, GPIO 177
- power supply signals 15
- power-down, automatic (ADC) 133
- power-on and voltage brown-out 173
- power-on reset (POR) 27
- problem description or suggestion 217
- product information 216
- program control instructions 190
- program counter 185
- program memory 18
- PUSH 189
- push using extended addressing 189
- PUSHX 189
- PWM mode 70
- PxADDR register 37
- PxCTL register 38

Q

- QFP 210

R

- R 184
- r 184
- RA, register address 184
- RCF 188, 189
- receive
 - 10-bit data format (I2C) 116
 - 7-bit data transfer format (I2C) 115
 - IrDA data 97
- receive interrupt 112
- receiving UART data-DMA controller 83
- receiving UART data-interrupt-driven method 82
- receiving UART data-pollled method 82



- SDA and SCL (IrDA) signals 111
- second opcode map after 1FH 205
- serial clock 101
- serial peripheral interface (SPI) 99
- set carry flag 188, 189
- set register pointer 189
- shift right arithmetic 191
- shift right logical 191
- signal descriptions 13
- single assertion (pulse) interrupt sources 47
- single-shot conversion (ADC) 133
- SIO 5
- slave data transfer formats (I2C) 114
- slave select 102
- software trap 190
- source operand 185
- SP 185
- SPI
 - architecture 99
 - baud rate generator 105
 - baud rate high and low byte register 110
 - clock phase 102
 - configured as slave 100
 - control register 107
 - control register definitions 106
 - data register 106
 - error detection 105
 - interrupts 105
 - mode fault error 105
 - mode register 109
 - multi-master operation 104
 - operation 100
 - overflow error 105
 - signals 101
 - single master, multiple slave system 100
 - single master, single slave system 99
 - status register 108
 - timing, PHASE = 0 103
 - timing, PHASE=1 104
- SPI controller signals 13
- SPI mode (SPIMODE) 109
- SPIBRH register 110
- SPIBRL register 110
- SPICTL register 107

- SPIDATA register 106
- SPIMODE register 109
- SPISTAT register 108
- SRA 191
- src 185
- SRL 191
- SRP 189
- SS, SPI signal 101
- stack pointer 185
- status register, I2C 118
- STOP 189
- stop mode 31, 189
- stop mode recovery
 - sources 29
 - using a GPIO port pin transition 30
 - using watch-dog timer time-out 29
- SUB 188
- subtract 188
- subtract - extended addressing 188
- subtract with carry 188
- subtract with carry - extended addressing 188
- SUBX 188
- SWAP 191
- swap nibbles 191
- symbols, additional 185
- system and short resets 26

T

- TCM 188
- TCMX 188
- technical support 213
- test complement under mask 188
- test under mask 188
- timer signals 14
- timers 5, 57
 - architecture 57
 - block diagram 58
 - capture mode 62, 71
 - capture/compare mode 65, 71
 - compare mode 63, 71
 - continuous mode 59, 70
 - counter mode 60
 - counter modes 70