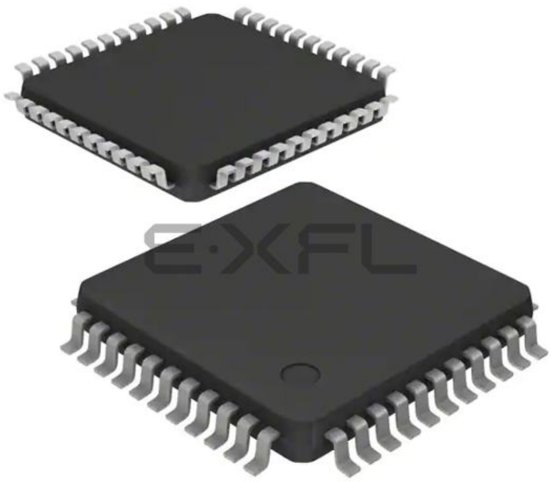

Welcome to E-XFL.COM

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	I²C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	31
Program Memory Size	48KB (48K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LQFP
Supplier Device Package	44-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f4801an020sc00tr

List of Figures

Table 2. Signal Descriptions (Continued)

Signal Mnemonic	I/O	Description
UART Controllers		
TXD0 / TXD1	O	Transmit Data. These signals are the transmit outputs from the UARTs. The TXD signals are multiplexed with general-purpose I/O pins.
RXD0 / RXD1	I	Receive Data. These signals are the receiver inputs for the UARTs and IrDAs. The RXD signals are multiplexed with general-purpose I/O pins.
$\overline{CTS0}$ / $\overline{CTS1}$	I	Clear To Send. These signals are control inputs for the UARTs. The $\overline{CTS}$ signals are multiplexed with general-purpose I/O pins.
Timers (Timer 3 is unavailable in the 40-and 44-pin packages)		
T0OUT / T1OUT/ T2OUT / T3OUT	O	Timer Output 0-3. These signals are output pins from the timers. The Timer Output signals are multiplexed with general-purpose I/O pins. T2OUT is not supported in the 40-pin package. T3OUT is not supported in the 40- and 44-pin packages.
T0IN / T1IN/ T2IN / T3IN	I	Timer Input 0-3. These signals are used as the capture, gating and counter inputs. The Timer Input signals are multiplexed with general-purpose I/O pins. T3IN is not supported in the 40- and 44-pin packages.
Analog		
ANA[11:0]	I	Analog Input. These signals are inputs to the analog-to-digital converter (ADC). The ADC analog inputs are multiplexed with general-purpose I/O pins.
VREF	I	Analog-to-digital converter reference voltage input. The VREF pin should be left unconnected (or capacitively coupled to analog ground) if the internal voltage reference is selected as the ADC reference voltage.
Oscillators		
XIN	I	External Crystal Input. This is the input pin to the crystal oscillator. A crystal can be connected between it and the XOUT pin to form the oscillator.
XOUT	O	External Crystal Output. This pin is the output of the crystal oscillator. A crystal can be connected between it and the XIN pin to form the oscillator. When the system clock is referred to in this manual, it refers to the frequency of the signal at this pin.
RCOUT	O	RC Oscillator Output. This signal is the output of the RC oscillator. It is multiplexed with a general-purpose I/O pin.
On-Chip Debugger		
DBG	I/O	Debug. This pin is the control and data input and output to and from the On-Chip Debugger. For operation of the On-chip debugger, all power pins (V_{DD} and AV_{DD} must be supplied with power, and all ground pins (V_{SS} and AV_{SS} must be grounded. This pin is open-drain and must have an external pull-up resistor to ensure proper operation.

Reset and Stop Mode Recovery

Overview

The Reset Controller within the Z8F640x family devices controls Reset and STOP Mode Recovery operation. In typical operation, the following events cause a Reset to occur:

- Power-On Reset (POR)
- Voltage Brown-Out (VBO)
- Watch-Dog Timer time-out (when configured via the `WDT_RES` Option Bit to initiate a reset)
- External $\overline{\text{RESET}}$ pin assertion
- On-Chip Debugger initiated Reset (OCDCTL[1] set to 1)

When the Z8F640x family device is in Stop mode, a Stop Mode Recovery is initiated by either of the following:

- Watch-Dog Timer time-out
- GPIO Port input pin transition on an enabled Stop Mode Recovery source
- DBG pin driven Low

Reset Types

The Z8F640x family provides several different types of Reset operation. Stop Mode Recovery is considered a form of Reset. The type of Reset is a function of both the current operating mode of the Z8F640x family device and the source of the Reset. Table 7 lists the types of Reset and their operating characteristics. The System Reset is longer than the Short Reset to allow additional time for external oscillator start-up.

Table 7. Reset and Stop Mode Recovery Characteristics and Latency

	Reset Characteristics and Latency		
Reset Type	**Control Registers**	**eZ8 CPU**	**Reset Latency (Delay)**
System Reset	Reset (as applicable)	Reset	514 WDT Oscillator cycles + 16 System Clock cycles
Short Reset	Reset (as applicable)	Reset	66 WDT Oscillator cycles + 16 System Clock cycles
Stop Mode Recovery	Unaffected, except WDT_CTL register	Reset	514 WDT Oscillator cycles + 16 System Clock cycles

Low-Power Modes

Overview

The Z8F640x family products contain power-saving features. The highest level of power reduction is provided by Stop mode. The next level of power reduction is provided by the Halt mode.

Stop Mode

Execution of the eZ8 CPU's STOP instruction places the Z8F640x family device into Stop mode. In Stop mode, the operating characteristics are:

- Primary crystal oscillator is stopped
- System clock is stopped
- eZ8 CPU is stopped
- Program counter (PC) stops incrementing
- Watch-Dog Timer's internal RC oscillator continues to operate
- If enabled, the Watch-Dog Timer continues to operate
- All other on-chip peripherals are idle

To minimize current in Stop mode, all GPIO pins that are configured as digital inputs must be driven to one of the supply rails (V_{CC} or GND). The Z8F640x family device can be brought out of Stop mode using Stop Mode Recovery. For more information on STOP Mode Recovery refer to the **Reset and Stop Mode Recovery** chapter.

Halt Mode

Execution of the eZ8 CPU's HALT instruction places the Z8F640x family device into Halt mode. In Halt mode, the operating characteristics are:

- Primary crystal oscillator is enabled and continues to operate
- System clock is enabled and continues to operate
- eZ8 CPU is idle
- Program counter (PC) stops incrementing

Table 12. GPIO Port Registers and Sub-Registers

Port Register Mnemonic	Port Register Name
P*x*ADDR	Port A-H Address Register (Selects sub-registers)
P*x*CTL	Port A-H Control Register (Provides access to sub-registers)
P*x*IN	Port A-H Input Data Register
P*x*OUT	Port A-H Output Data Register
Port Sub-Register Mnemonic	**Port Register Name**
P*x*DD	Data Direction
P*x*AF	Alternate Function
P*x*OC	Output Control (Open-Drain)
P*x*HDE	High Drive Enable
P*x*SMRE	STOP Mode Recovery Source Enable

Port A-H Address Registers

The Port A-H Address registers select the GPIO Port functionality accessible through the Port A-H Control registers. The Port A-H Address and Control registers combine to provide access to all GPIO Port control (Table 13).

Table 13. Port A-H GPIO Address Registers (P*x*ADDR)

BITS	7	6	5	4	3	2	1	0
FIELD	PADDR[7:0]							
RESET	00H							
R/W	R/W							
ADDR	FD0H, FD4H, FD8H, FDCH, FE0H, FE4H, FE8H, FECH							

Port A-H Data Direction Sub-Registers

The Port A-H Data Direction sub-register is accessed through the Port A-H Control register by writing 01H to the Port A-H Address register (Table 15).

Table 15. Port A-H Data Direction Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
RESET	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	If 01H in Port A-H Address Register, accessible via Port A-H Control Register							

DD[7:0]—Data Direction
These bits control the direction of the associated port pin. Port Alternate Function operation overrides the Data Direction register setting.
0 = Output. Data in the Port A-H Output Data register is driven onto the port pin.
1 = Input. The port pin is sampled and the value written into the Port A-H Input Data Register. The output driver is tri-stated.

Port A-H Alternate Function Sub-Registers

The Port A-H Alternate Function sub-register (Table 16) is accessed through the Port A-H Control register by writing `02H` to the Port A-H Address register. The Port A-H Alternate Function sub-registers select the alternate functions for the selected pins. Refer to the **GPIO Alternate Functions** section to determine the alternate function associated with each port pin.

Caution: Do not enable alternate function for GPIO port pins which do not have an associated alternate function. Failure to follow this guideline may result in unpredictable operation.

Table 16. Port A-H Alternate Function Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	AF7	AF6	AF5	AF4	AF3	AF2	AF1	AF0
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
ADDR	If 02H in Port A-H Address Register, accessible via Port A-H Control Register							

Port A-H High Drive Enable Sub-Registers

The Port A-H High Drive Enable sub-register (Table 18) is accessed through the Port A-H Control register by writing `04H` to the Port A-H Address register. Setting the bits in the Port A-H High Drive Enable sub-registers to 1 configures the specified port pins for high current output drive operation. The Port A-H High Drive Enable sub-register affects the pins directly and, as a result, alternate functions are also affected.

Table 18. Port A-H High Drive Enable Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	PHDE7	PHDE6	PHDE5	PHDE4	PHDE3	PHDE2	PHDE1	PHDE0
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
ADDR	If 04H in Port A-H Address Register, accessible via Port A-H Control Register							

PHDE[7:0]—Port High Drive Enabled
0 = The Port pin is configured for standard output current drive.
1 = The Port pin is configured for high output current drive.

Port A-H Stop Mode Recovery Source Enable Sub-Registers

The Port A-H STOP Mode Recovery Source Enable sub-register (Table 19) is accessed through the Port A-H Control register by writing `05H` to the Port A-H Address register. Setting the bits in the Port A-H STOP Mode Recovery Source Enable sub-registers to 1 configures the specified Port pins as a STOP Mode Recovery source. During STOP Mode, any logic transition on a Port pin enabled as a STOP Mode Recovery source initiates STOP Mode Recovery.

Table 19. Port A-H STOP Mode Recovery Source Enable Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	PSMRE7	PSMRE6	PSMRE5	PSMRE4	PSMRE3	PSMRE2	PSMRE1	PSMRE0
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
ADDR	If 05H in Port A-H Address Register, accessible via Port A-H Control Register							

PSMRE[7:0]—Port STOP Mode Recovery Source Enabled
0 = The Port pin is not configured as a STOP Mode Recovery source. Transitions on this pin during Stop mode do not initiate STOP Mode Recovery.

1 = The Port pin is configured as a STOP Mode Recovery source. Any logic transition on this pin during Stop mode initiates STOP Mode Recovery.

Port A-H Input Data Registers

Reading from the Port A-H Input Data registers (Table 20) returns the sampled values from the corresponding port pins. The Port A-H Input Data registers are Read-only.

Table 20. Port A-H Input Data Registers (PxIN)

BITS	7	6	5	4	3	2	1	0
FIELD	PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0
RESET	X	X	X	X	X	X	X	X
R/W	R	R	R	R	R	R	R	R
ADDR	FD2H, FD6H, FDAH, FDEH, FE2H, FE6H, FEAH, FEEH							

PIN[7:0]—Port Input Data
Sampled data from the corresponding port pin input.
0 = Input data is logical 0 (Low).
1 = Input data is logical 1 (High).

Interrupt Control Register

The Interrupt Control (IRQCTL) register (Table 37) contains the master enable bit for all interrupts.

Table 37. Interrupt Control Register (IRQCTL)

BITS	7	6	5	4	3	2	1	0
FIELD	IRQE	Reserved						
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R
ADDR	FCFH							

IRQE—Interrupt Request Enable
This bit is set to 1 by execution of an EI (Enable Interrupts) or IRET (Interrupt Return) instruction, or by a direct register write of a 1 to this bit. It is reset to 0 by executing a DI instruction, eZ8 CPU acknowledgement of an interrupt request, or Reset.
0 = Interrupts are disabled.
1 = Interrupts are enabled.

Reserved
These bits must be 0.

Timer 0-3 Control Registers

The Timer 0-3 Control (TxCTL) registers enable/disable the timers, set the prescaler value, and determine the timer operating mode.

Table 44. Timer 0-3 Control Register (TxCTL)

BITS	7	6	5	4	3	2	1	0
FIELD	TEN	TPOL	PRES			TMODE		
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
ADDR	F07H, F0FH, F17H, F1FH							

TEN—Timer Enable
0 = Timer is disabled.
1 = Timer enabled to count.

TPOL—Timer Input/Output Polarity
Operation of this bit is a function of the current operating mode of the timer.

One-Shot mode
When the timer is disabled, the Timer Output signal is set to the value of this bit. When the timer is enabled, the Timer Output signal is complemented upon timer Reload.

Continuous mode
When the timer is disabled, the Timer Output signal is set to the value of this bit. When the timer is enabled, the Timer Output signal is complemented upon timer Reload.

Counter mode
When the timer is disabled, the Timer Output signal is set to the value of this bit. When the timer is enabled, the Timer Output signal is complemented upon timer Reload.

PWM mode
0 = Timer Output is forced Low (0) when the timer is disabled. When enabled, the Timer Output is forced High (1) upon PWM count match and forced Low (0) upon Reload.
1 = Timer Output is forced High (1) when the timer is disabled. When enabled, the Timer Output is forced Low (0) upon PWM count match and forced High (1) upon Reload.

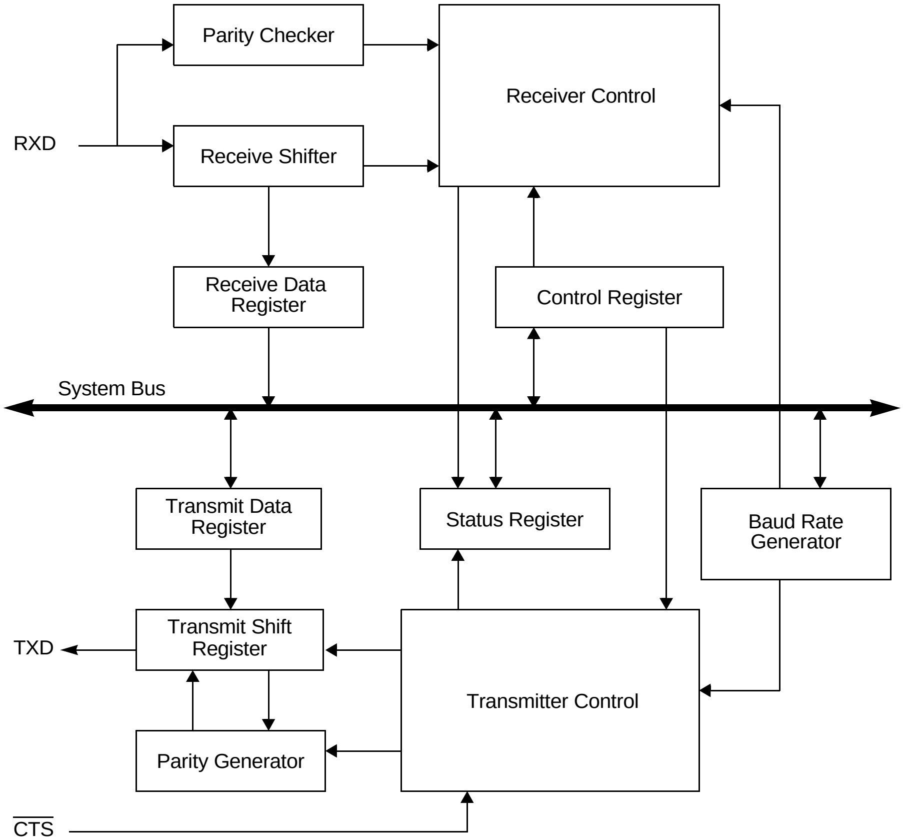

Figure 67. UART Block Diagram

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.

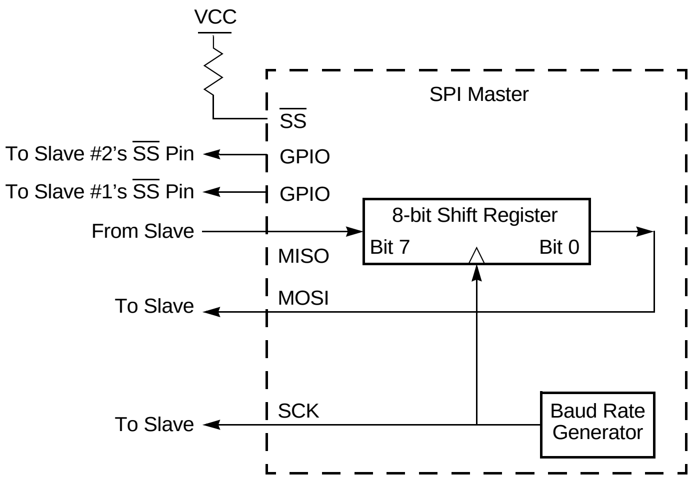

Figure 75. SPI Configured as a Master in a Single Master, Multiple Slave System

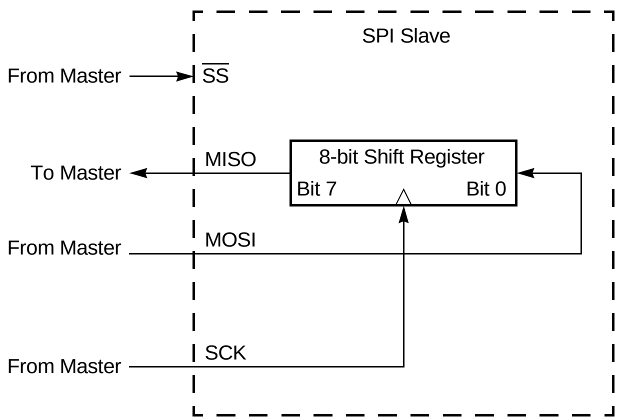

Figure 76. SPI Configured as a Slave

Operation

The SPI is a full-duplex, synchronous, character-oriented channel that supports a four-wire interface (serial clock, transmit, receive and Slave select). The SPI block consists of trans-

Direct Memory Access Controller

Overview

The Z8F640x family device's Direct Memory Access (DMA) Controller provides three independent Direct Memory Access channels. Two of the channels (DMA0 and DMA1) transfer data between the on-chip peripherals and the Register File. The third channel (DMA_ADC) controls the Analog-to-Digital Converter (ADC) operation and transfers the Single-Shot mode ADC output data to the Register File.

Operation

DMA0 and DMA1 Operation

DMA0 and DMA1, referred to collectively as DMA*x*, transfer data either from the on-chip peripheral control registers to the Register File, or from the Register File to the on-chip peripheral control registers. The sequence of operations in a DMA*x* data transfer is:

1. DMA*x* trigger source requests a DMA data transfer.
2. DMA*x* requests control of the system bus (address and data) from the eZ8 CPU.
3. After the eZ8 CPU acknowledges the bus request, DMA*x* transfers either a single byte or a two-byte word (depending upon configuration) and then returns system bus control back to the eZ8 CPU.
4. If Current Address equals End Address:
   - DMA*x* reloads the original Start Address
   - If configured to generate an interrupt, DMA*x* sends an interrupt request to the Interrupt Controller
   - If configured for single-pass operation, DMA*x* resets the `DEN` bit in the DMA*x* Control register to 0 and the DMA is disabled.

   If Current Address does not equal End Address, the Current Address increments by 1 (single-byte transfer) or 2 (two-byte word transfer).

If the current ADC Analog Input is not the highest numbered input to be converted, DMA_ADC initiates data conversion in the next higher numbered ADC Analog Input.

Configuring DMA_ADC for Data Transfer

Follow these steps to configure and enable DMA_ADC:

1. Write the DMA_ADC Address register with the 7 most-significant bits of the Register File address for data transfers.
2. Write to the DMA_ADC Control register to complete the following:
   – Enable the DMA_ADC interrupt request, if desired
   – Select the number of ADC Analog Inputs to convert
   – Enable the DMA_ADC channel

Caution: When using the DMA_ADC to perform conversions on multiple ADC inputs and the `ADC_IN` field in the DMA_ADC Control Register is greater than 000b, the Analog-to-Digital Converter must be configured for Single-Shot mode.

Continuous mode operation of the ADC can **only** be used in conjunction with DMA_ADC if the `ADC_IN` field in the DMA_ADC Control Register is reset to 000b to enable conversion on ADC Analog Input 0 only.

DMA Control Register Definitions

DMA*x* Control Register

The DMA*x* Control register is used to enable and select the mode of operation for DMA*x*.

Table 71. DMA*x* Control Register (DMA*x*CTL)

BITS	7	6	5	4	3	2	1	0
FIELD	DEN	DLE	DDIR	IRQEN	WSEL	RSS		
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
ADDR	FB0H, FB8H							

DEN—DMA*x* Enable
0 = DMA*x* is disabled and data transfer requests are disregarded.

Table 74. DMA*x* Start/Current Address Low Byte Register (DMA*x*START)

BITS	7	6	5	4	3	2	1	0
FIELD	DMA_START							
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	FB3H, FHBH							

DMA_START—DMA*x* Start/Current Address Low
These bits, with the four lower bits of the DMA*x*_H register, form the 12-bit Start/Current address. The full 12-bit address is given by {DMA_START_H[3:0], DMA_START[7:0]}.

DMA*x* End Address Low Byte Register

The DMA*x* End Address Low Byte register, in conjunction with the DMA*x*_H register, forms a 12-bit End Address.

Table 75. DMA*x* End Address Low Byte Register (DMA*x*END)

BITS	7	6	5	4	3	2	1	0
FIELD	DMA_END							
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	FB4H, FBCH							

DMA_END—DMA*x* End Address Low
These bits, with the four upper bits of the DMA*x*_H register, form a 12-bit address. This address is the ending location of the DMA*x* transfer. The full 12-bit address is given by {DMA_END_H[3:0], DMA_END[7:0]}.

DMA_ADC Address Register

The DMA_ADC Address register points to a block of the Register File to store ADC conversion values as illustrated in Table 76. This register contains the seven most-significant bits of the 12-bit Register File addresses. The five least-significant bits are calculated from the ADC Analog Input number (5-bit base address is equal to twice the ADC Analog Input number). The 10-bit ADC conversion data is stored as two bytes with the most significant byte of the ADC data stored at the even numbered Register File address.

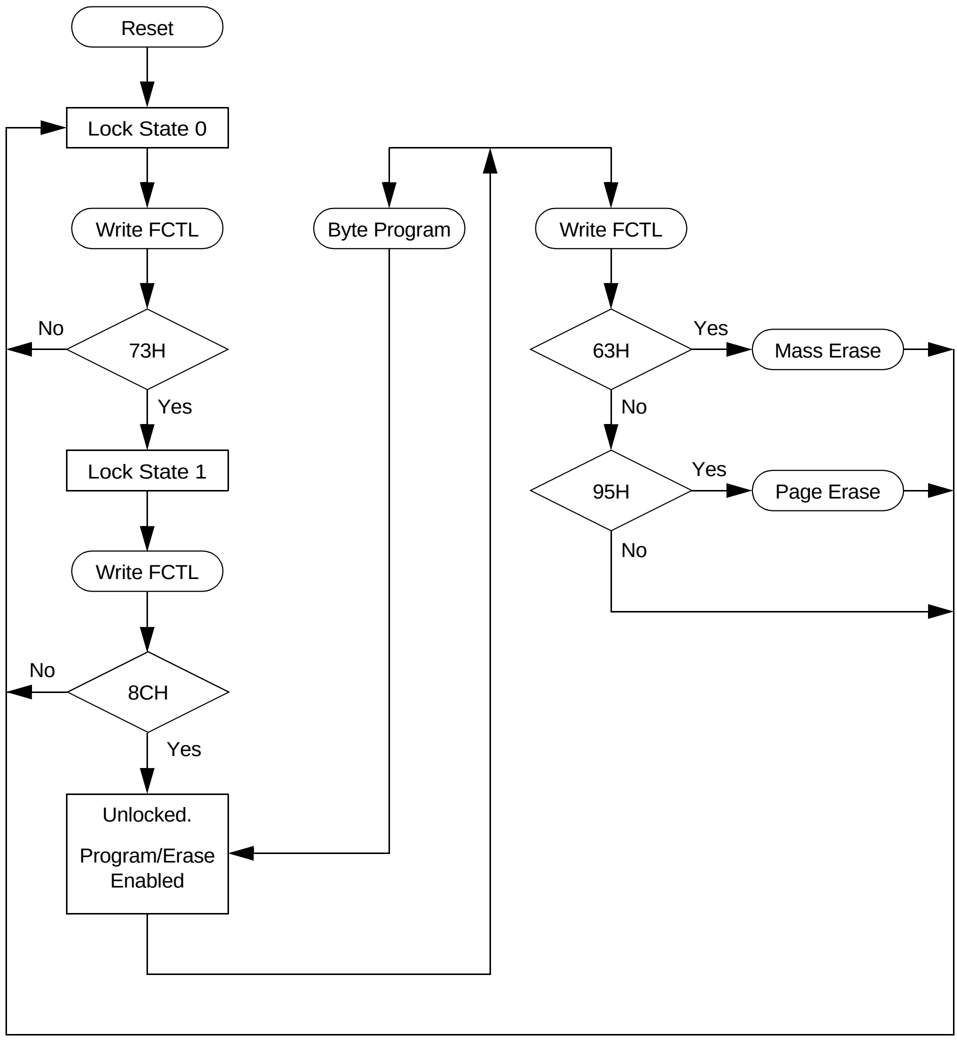

Figure 85. Flash Controller Operation Flow Chart

Flash Status Register

The Flash Status register indicates the current state of the Flash Controller. This register can be read at any time. The Read-only Flash Status Register shares its Register File address with the Write-only Flash Control Register.

Table 86. Flash Status Register (FSTAT)

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

Reserved
These bits are reserved and must be 0.

FSTAT—Flash Controller Status
000000 = Flash Controller locked.
000001 = First unlock command received.
000010 = Flash Controller unlocked (second unlock command received).
001xxx = Program operation in progress.
010xxx = Page erase operation in progress.
100xxx = Mass erase operation in progress.

Operation

OCD Interface

The On-Chip Debugger uses the DBG pin for communication with an external host. This one-pin interface is a bi-directional open-drain interface that transmits and receives data. Data transmission is half-duplex, in that transmit and receive cannot occur simultaneously. The serial data on the DBG pin is sent using the standard asynchronous data format defined in RS-232. This pin can interface the Z8F640x family device to the serial port of a host PC using minimal external hardware.Two different methods for connecting the DBG pin to an RS-232 interface are depicted in Figures 87 and 88.

Caution: For operation of the On-Chip Debugger, ***all*** power pins (VDD and AVDD) must be supplied with power, and ***all*** ground pins (VSS and AVSS) must be properly grounded.
The DBG pin is open-drain and must always be connected to V_{DD} through an external pull-up resistor to ensure proper operation.

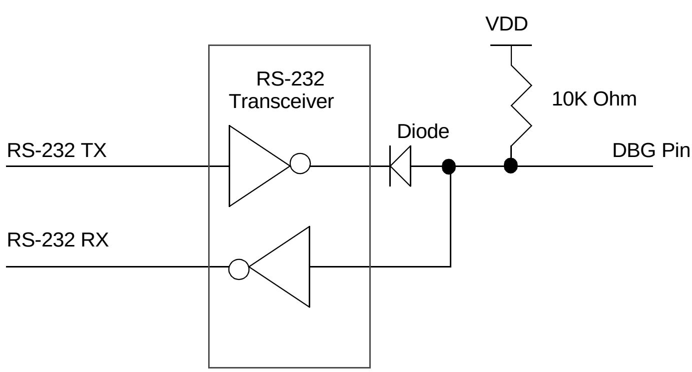

Figure 87. Interfacing the On-Chip Debugger's DBG Pin with an RS-232 Interface (1)

Flags Register

The Flags Register contains the status information regarding the most recent arithmetic, logical, bit manipulation or rotate and shift operation. 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 Binary-Coded Decimal (BCD) arithmetic.

The two remaining bits, User Flags (F1 and F2), are available as general-purpose status bits. User Flags are unaffected by arithmetic operations and must be set or cleared by instructions. The User Flags cannot be used with conditional Jumps. They are undefined at initial power-up and are unaffected by Reset. Figure 99 illustrates the flags and their bit positions in the Flags Register.

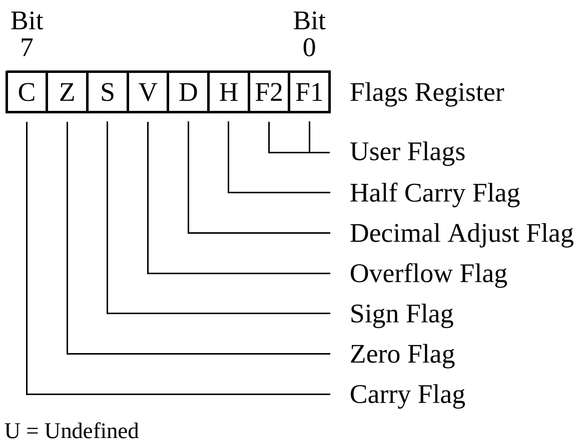

Figure 99. Flags Register

Interrupts, the Software Trap (TRAP) instruction, and Illegal Instruction Traps all write the value of the Flags Register to the stack. Executing an Interrupt Return (IRET) instruction restores the value saved on the stack into the Flags Register.

Lower Nibble (Hex)

Upper Nibble (Hex)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2																
3																
4																
5																
6																
7																
8																
9																
A			3.3 **CPC** r1,r2	3.4 **CPC** r1,Ir2	4.3 **CPC** R2,R1	4.4 **CPC** IR2,R1	4.3 **CPC** R1,IM	4.4 **CPC** IR1,IM	5.3 **CPCX** ER2,ER1	5.3 **CPCX** IM,ER1						
B																
C	3.2 **SRL** R1	3.3 **SRL** IR1														
D																
E																
F																

Figure 102. Second Opcode Map after 1FH