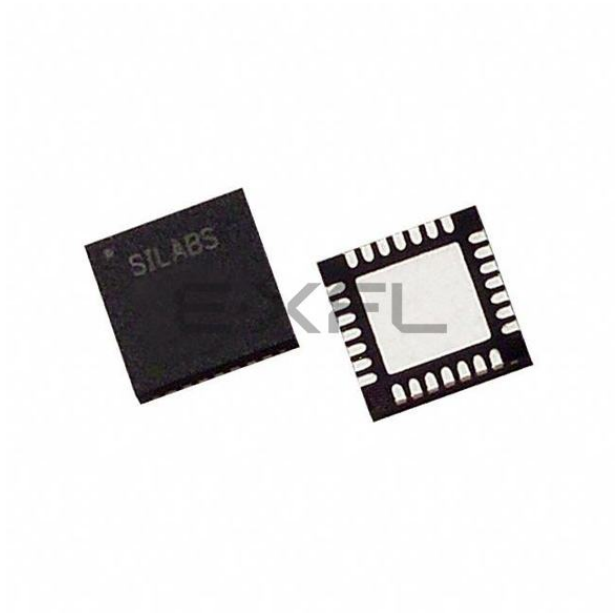

Welcome to E-XFL.COM

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	Not For New Designs
Core Processor	8051
Core Size	8-Bit
Speed	25MHz
Connectivity	UART/USART, USB
Peripherals	POR
Number of I/O	15
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1.5K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	28-VFQFN Exposed Pad
Supplier Device Package	28-QFN (5x5)
Purchase URL	https://www.e-xfl.com/product-detail/silicon-labs/c8051f326-gm

Figure 13.2. UART0 Timing Without Parity or Extra Bit 120
Figure 13.3. UART0 Timing With Parity 120
Figure 13.4. UART0 Timing With Extra Bit 120
Figure 13.5. Typical UART Interconnect Diagram 121
Figure 13.6. UART Multi-Processor Mode Interconnect Diagram 122
14. Timers
Figure 14.1. T0 Mode 0 Block Diagram 128
Figure 14.2. T0 Mode 2 Block Diagram 129
Figure 14.3. T0 Mode 3 Block Diagram 130
15. C2 Interface
Figure 15.1. Typical C2 Pin Sharing 137

List of Tables

1. System Overview
Table 1.1. Product Selection Guide 13
2. Absolute Maximum Ratings
Table 2.1. Absolute Maximum Ratings 23
3. Global DC Electrical Characteristics
Table 3.1. Global DC Electrical Characteristics 24
4. Pinout and Package Definitions
Table 4.1. Pin Definitions for the C8051F326/7 25
Table 4.2. QFN-28 Package Dimensions 29
Table 4.3. QFN-28 PCB Land Pattern Dimesions 30
5. Voltage Regulator (REG0)
Table 5.1. Voltage Regulator Electrical Specifications 31
6. CIP-51 Microcontroller
Table 6.1. CIP-51 Instruction Set Summary 37
Table 6.2. Special Function Register (SFR) Memory Map 43
Table 6.3. Special Function Registers 43
Table 6.4. TMOD.3 Control of /INT0 49
Table 6.5. Interrupt Summary 50
7. Reset Sources
Table 7.1. Reset Electrical Characteristics 62
8. Flash Memory
Table 8.1. Flash Electrical Characteristics 64
9. External RAM
10. Oscillators
Table 10.1. Typical USB Full Speed Clock Settings 76
Table 10.2. Typical USB Low Speed Clock Settings 76
Table 10.3. Internal Oscillator Electrical Characteristics 78
11. Port Input/Output
Table 11.1. Port I/O DC Electrical Characteristics (C8051F326) 85
Table 11.2. Port I/O DC Electrical Characteristics (C8051F327) 85
12. Universal Serial Bus Controller (USB0)
Table 12.1. Endpoint Addressing Scheme 88
Table 12.2. USB0 Controller Registers 93
Table 12.3. FIFO Configurations 95
Table 12.4. USB Transceiver Electrical Characteristics 115
13. UART0
Table 13.1. Baud Rate Generator Settings for Standard Baud Rates 119
14. Timers
Table 14.1. Timer Modes 127
Table 14.2. Timer 0 Operation 128
15. C2 Interface

1. System Overview

C8051F326/7 devices are fully integrated mixed-signal system-on-a-chip MCUs. Highlighted features are listed below. Refer to Table 1.1 for specific product feature selection.

- High-speed pipelined 8051-compatible microcontroller core (up to 25 MIPS)
- In-system, full-speed, non-intrusive debug interface (on-chip)
- Universal serial bus (USB) function controller with three fixed-function endpoint pipes, integrated transceiver, and 256B FIFO RAM
- Supply voltage regulator
- Precision programmable 12 MHz internal oscillator and 4x clock multiplier
- 16k kB of on-chip Flash memory
- 1536 total bytes of on-chip RAM (256 + 1 k + 256 USB FIFO)
- Enhanced UART, serial interfaces implemented in hardware
- Two general-purpose 16-bit timers
- On-chip power-on reset, VDD monitor, and missing clock detector
- 15 Port I/O (5 V tolerant)

With on-chip power-on reset, VDD monitor, voltage regulator, and clock oscillator, C8051F326/7 devices are truly stand-alone System-on-a-Chip solutions. The Flash memory can be reprogrammed in-circuit, providing non-volatile data storage, and also allowing field upgrades of the 8051 firmware. User software has complete control of all peripherals, and may individually shut down any or all peripherals for power savings.

The on-chip Silicon Laboratories 2-Wire (C2) Development Interface allows non-intrusive (uses no on-chip resources), full speed, in-circuit debugging using the production MCU installed in the final application. This debug logic supports inspection and modification of memory and registers, setting breakpoints, single stepping, run and halt commands. All analog and digital peripherals are fully functional while debugging using C2. The two C2 interface pins can be shared with user functions, allowing in-system debugging without occupying package pins.

Each device is specified for 2.7–5.25 V operation over the industrial temperature range (–40 to +85 °C). For voltages above 3.6 V, the on-chip Voltage Regulator must be used. A minimum of 3.0 V is required for USB communication. The Port I/O and $\overline{\text{RST}}$ pins are tolerant of input signals up to 5 V. C8051F326/7 are available in two 28-pin QFN packages with different pinouts. The RoHS compliant devices are marked with a -GM suffix in the part number. The port I/O on C8051F326 devices is powered from a separate I/O supply allowing it to interface to low voltage logic.

Table 1.1. Product Selection Guide

Ordering Part Number	MIPS (Peak)	Flash Memory	RAM	Calibrated Internal Oscillator	USB	Supply Voltage Regulator	UART	Timers (16-bit)	Digital Port I/Os	Separate I/O Supply	Package
C8051F326-GM	25	16k	1536	✓	✓	✓	✓	2	15	✓	QFN-28
C8051F327-GM	25	16k	1536	✓	✓	✓	✓	2	15	—	QFN-28

Figure 1.3. Typical Connections for the C8051F326

Figure 1.4. Typical Connections for the C8051F327

1.1.3. Additional Features

The C8051F326/7 SoC family includes several key enhancements to the CIP-51 core and peripherals to improve performance and ease of use in end applications.

The extended interrupt handler provides 8 interrupt sources into the CIP-51. An interrupt driven system requires less intervention by the MCU, giving it more effective throughput. The interrupt sources are very useful when building multi-tasking, real-time systems.

Seven reset sources are available: power-on reset circuitry (POR), an on-chip VDD monitor (forces reset when power supply voltage drops below V_{RST} as given in Table 7.1 on page 62), the USB controller (USB bus reset or a VBUS transition), a Missing Clock Detector, a forced software reset, an external reset pin, and an errant Flash read/write protection circuit. Each reset source except for the POR, Reset Input Pin, or Flash error may be disabled by the user in software.

The internal oscillator is factory calibrated to 12 MHz ±1.5%, and the internal oscillator period may be user programmed in ~0.25% increments. An additional low-frequency oscillator is also available which facilitates low power operation. A clock recovery mechanism allows the internal oscillator to be used with the 4x Clock Multiplier as the USB clock source in Full Speed mode; the internal oscillator can also be used as the USB clock source in Low Speed mode. An external CMOS clock may also be used with the 4x Clock Multiplier. The system clock may be configured to use the internal oscillator, external clock, low-frequency oscillator, or the Clock Multiplier output divided by 2. If desired, the system clock source may be switched on-the-fly between oscillator sources. The external clock and internal low-frequency oscillator can be extremely useful in low power applications, allowing the MCU to run from a slow (power saving) clock source, while periodically switching to the high-frequency internal oscillator as needed.

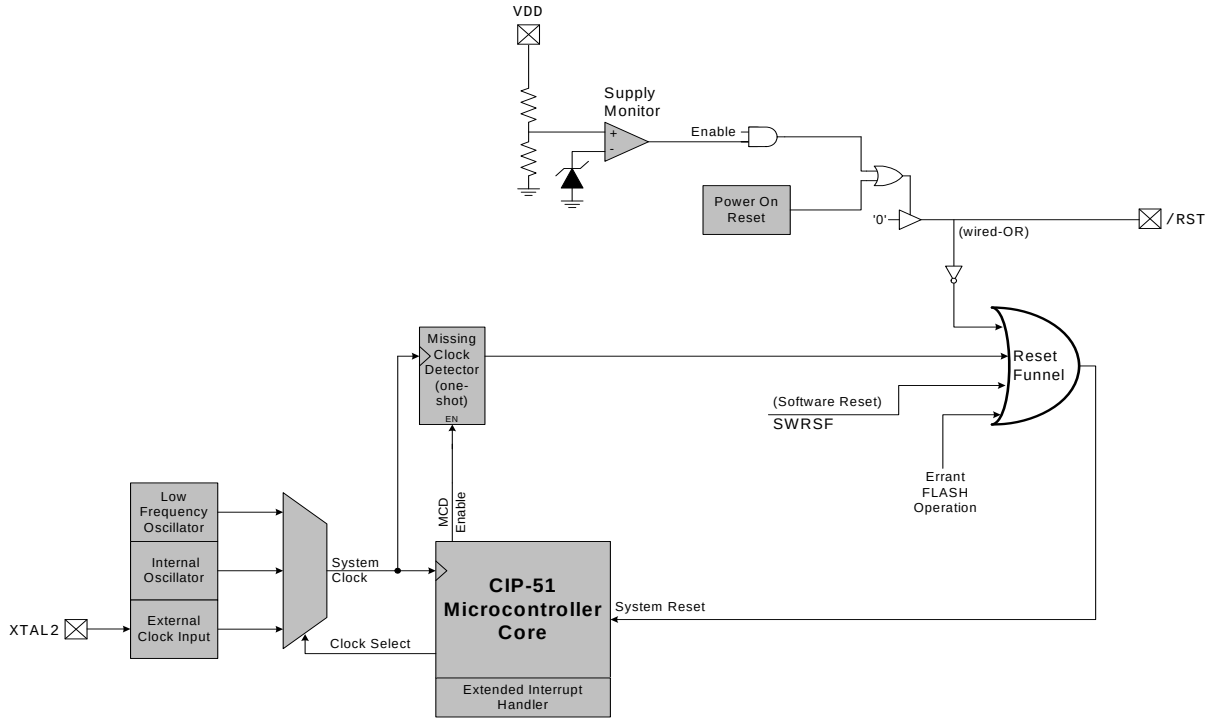

Figure 1.6. On-Chip Clock and Reset

1.5. On-Chip Debug Circuitry

C8051F326/7 devices include on-chip Silicon Laboratories 2-Wire (C2) debug circuitry that provides non-intrusive, full speed, in-circuit debugging of the production part *installed in the end application.*

The Silicon Laboratories' debugging system supports inspection and modification of memory and registers, breakpoints, and single stepping. No additional target RAM, program memory, timers, or communications channels are required. All the digital and analog peripherals are functional and work correctly while debugging. All the peripherals (except for the USB) are stalled when the MCU is halted, during single stepping, or at a breakpoint in order to keep them synchronized.

The C8051F326DK development kit provides all the hardware and software necessary to develop application code and perform in-circuit debugging with the C8051F326/7 MCUs. The kit includes a Windows development environment, a serial adapter for connecting to the C2 port, and a target application board. All of the necessary communication cables and a wall-mount power supply are also supplied with the development kit. The Silicon Laboratories debug environment is a vastly superior configuration for developing and debugging embedded applications compared to standard MCU emulators, which use on-board "ICE Chips" and target cables and require the MCU in the application board to be socketed. The Silicon Laboratories debug environment enhances ease of use and preserves the performance of on-chip peripherals.

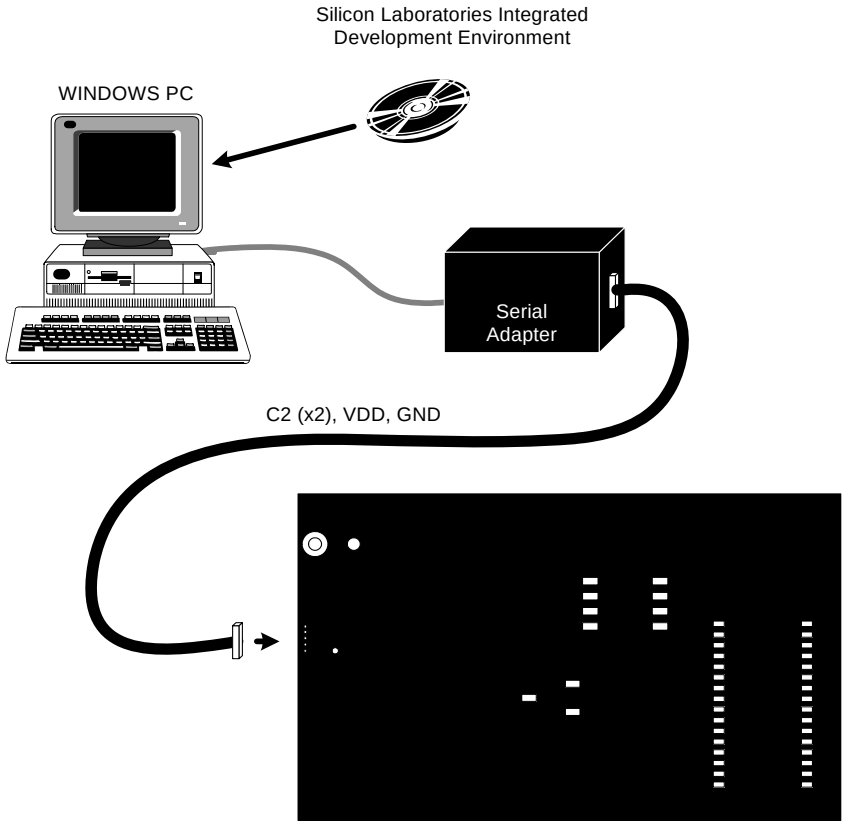

Figure 1.9. Development/In-System Debug Diagram

Performance
The CIP-51 employs a pipelined architecture that greatly increases its instruction throughput over the standard 8051 architecture. In a standard 8051, all instructions except for MUL and DIV take 12 or 24 system clock cycles to execute, and usually have a maximum system clock of 12 MHz. By contrast, the CIP-51 core executes 70% of its instructions in one or two system clock cycles, with no instructions taking more than eight system clock cycles.

With the CIP-51's maximum system clock at 25 MHz, it has a peak throughput of 25 MIPS. The CIP-51 has a total of 109 instructions. The table below shows the total number of instructions that for execution time.

Clocks to Execute	1	2	2/3	3	3/4	4	4/5	5	8
Number of Instructions	26	50	5	14	7	3	1	2	1

Programming and Debugging Support
In-system programming of the Flash program memory and communication with on-chip debug support logic is accomplished via the Silicon Laboratories 2-Wire Development Interface (C2). Note that the re-programmable Flash can also be read and changed a single byte at a time by the application software using the MOVC and MOVX instructions. This feature allows program memory to be used for non-volatile data storage as well as updating program code under software control.

The on-chip debug support logic facilitates full speed in-circuit debugging, allowing the setting of hardware breakpoints, starting, stopping and single stepping through program execution (including interrupt service routines), examination of the program's call stack, and reading/writing the contents of registers and memory. This method of on-chip debugging is completely non-intrusive, requiring no RAM, Stack, timers, or other on-chip resources. C2 details can be found in Section “15. C2 Interface” on page 135.

The CIP-51 is supported by development tools from Silicon Laboratories and third party vendors. Silicon Laboratories provides an integrated development environment (IDE) including editor, macro assembler, debugger and programmer. The IDE's debugger and programmer interface to the CIP-51 via the C2 interface to provide fast and efficient in-system device programming and debugging. Third party macro assemblers and C compilers are also available.

6.1. Instruction Set

The instruction set of the CIP-51 System Controller is fully compatible with the standard MCS-51™ instruction set. Standard 8051 development tools can be used to develop software for the CIP-51. All CIP-51 instructions are the binary and functional equivalent of their MCS-51™ counterparts, including opcodes, addressing modes and effect on PSW flags. However, instruction timing is different than that of the standard 8051.

6.1.1. Instruction and CPU Timing

In many 8051 implementations, a distinction is made between machine cycles and clock cycles, with machine cycles varying from 2 to 12 clock cycles in length. However, the CIP-51 implementation is based solely on clock cycle timing. All instruction timings are specified in terms of clock cycles.

Due to the pipelined architecture of the CIP-51, most instructions execute in the same number of clock cycles as there are program bytes in the instruction. Conditional branch instructions take one less clock cycle to complete when the branch is not taken as opposed to when the branch is taken. Table 6.1 is the CIP-51 Instruction Set Summary, which includes the mnemonic, number of bytes, and number of clock cycles for each instruction.

6.3. Interrupt Handler

The CIP-51 includes an extended interrupt system supporting a total of 8 interrupt sources with two priority levels. The allocation of interrupt sources between on-chip peripherals and external inputs pins varies according to the specific version of the device. Each interrupt source, with the exception of USB0, has one or more associated interrupt-pending flag(s) located in an SFR. USB0 interrupt sources are located in the USB registers. See Section “12.8. Interrupts” on page 101 for more details about the USB interrupt. When a peripheral or external source meets a valid interrupt condition, the associated interrupt-pending flag is set to logic 1.

If interrupts are enabled for the source, an interrupt request is generated when the interrupt-pending flag is set. As soon as execution of the current instruction is complete, the CPU generates an LCALL to a predetermined address to begin execution of an interrupt service routine (ISR). Each ISR must end with an RETI instruction, which returns program execution to the next instruction that would have been executed if the interrupt request had not occurred. If interrupts are not enabled, the interrupt-pending flag is ignored by the hardware and program execution continues as normal. (The interrupt-pending flag is set to logic 1 regardless of the interrupt's enable/disable state.)

Each interrupt source can be individually enabled or disabled through the use of an associated interrupt enable bit in an SFR (IE-EIE2). However, interrupts must first be globally enabled by setting the EA bit (IE.7) to logic 1 before the individual interrupt enables are recognized. Setting the EA bit to logic 0 disables all interrupt sources regardless of the individual interrupt-enable settings.

Note: Any instruction which clears the EA bit should be immediately followed by an instruction which has two or more opcode bytes. For example:

```
// in 'C':

EA = 0;    // clear EA bit

EA = 0;    // ... followed by another 2-byte opcode


; in assembly:

CLR  EA    ; clear EA bit

CLR  EA    ; ... followed by another 2-byte opcode
```

If an interrupt is posted during the execution phase of a "CLR EA" opcode (or any instruction that clears the EA bit), and the instruction is followed by a single-cycle instruction, the interrupt may be taken. If the EA bit is read inside the interrupt service routine, it will return a '0'. When the "CLR EA" opcode is followed by a multi-cycle instruction, the interrupt will not be taken.

Some interrupt-pending flags are automatically cleared by the hardware when the CPU vectors to the ISR. However, most are not cleared by the hardware and must be cleared by software before returning from the ISR. If an interrupt-pending flag remains set after the CPU completes the return-from-interrupt (RETI) instruction, a new interrupt request will be generated immediately and the CPU will re-enter the ISR after the completion of the next instruction.

6.3.1. MCU Interrupt Sources and Vectors

The MCU supports 8 interrupt sources. Software can simulate an interrupt by setting any interrupt-pending flag to logic 1. If interrupts are enabled for the flag, an interrupt request will be generated and the CPU will vector to the ISR address associated with the interrupt-pending flag. MCU interrupt sources, associated vector addresses, priority order and control bits are summarized in Table 6.5 on page 50. Refer to the data sheet section associated with a particular on-chip peripheral for information regarding valid interrupt conditions for the peripheral and the behavior of its interrupt-pending flag(s).

SFR Definition 7.2. RSTSRC: Reset Source

R/W	R	R	R/W	R	R/W	R/W	R	Reset Value
USBRSF	FERROR	—	SWRSF	—	MCDRSF	PORSF	PINRSF	Variable
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xEF

Bit7: USBRSF: USB Reset Flag
0: **Read:** Last reset was not a USB reset; **Write:** USB resets disabled.
1: **Read:** Last reset was a USB reset; **Write:** USB resets enabled.

Bit6: FERROR: Flash Error Indicator.
0: Source of last reset was not a Flash read/write/erase error.
1: Source of last reset was a Flash read/write/erase error.

Bit5: Unused. Read = 0. Write = don't care.

Bit4: SWRSF: Software Reset Force and Flag.
0: **Read:** Source of last reset was not a write to the SWRSF bit; **Write:** No Effect.
1: **Read:** Source of last was a write to the SWRSF bit; **Write:** Forces a system reset.

Bit3: Unused. Read = 0. Write = don't care.

Bit2: MCDRSF: Missing Clock Detector Flag.
0: **Read:** Source of last reset was not a Missing Clock Detector timeout; **Write:** Missing Clock Detector disabled.
1: **Read:** Source of last reset was a Missing Clock Detector timeout; **Write:** Missing Clock Detector enabled; triggers a reset if a missing clock condition is detected.

Bit1: PORSF: Power-On / VDD Monitor Reset Flag.
This bit is set anytime a power-on reset occurs. Writing this bit selects/deselects the VDD monitor as a reset source. **Note: writing '1' to this bit before the VDD monitor is enabled and stabilized can cause a system reset.** See register VDM0CN (Figure 7.1).
0: **Read:** Last reset was not a power-on or VDD monitor reset; **Write:** VDD monitor is not a reset source.
1: **Read:** Last reset was a power-on or VDD monitor reset; all other reset flags indeterminate; **Write:** VDD monitor is a reset source.

Bit0: PINRSF: HW Pin Reset Flag.
0: Source of last reset was not $\overline{\text{RST}}$ pin.
1: Source of last reset was $\overline{\text{RST}}$ pin.

Note: Do not use read-modify-write instructions on this register.

SFR Definition 10.2. OSCICL: Internal Oscillator Calibration

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
—	—	—	OSCCAL					Variable
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
								0xB3

Bits4–0: OSCCAL: Oscillator Calibration Value
These bits determine the internal oscillator period.

Note: If the sum of the reset value of OSCCAL and ΔOSCCAL is greater than 31 or less than 0, then the device will not be capable of producing the desired frequency.

Note: The contents of this register are undefined when Clock Recovery is enabled. See Section "12.4. USB Clock Configuration" on page 94 for details on Clock Recovery.

11.1. Port I/O Initialization

Port I/O initialization consists of the following steps:

Step 1. Select if the port pin will be used as an output or input.
Step 2. If output, select the output mode: open-drain or push-pull.
Step 3. Configure the PnMDOUT and Pn latches according to the desired input or output configuration.
Step 4. Select if /SYSCLK will appear on the P0.0 output and configure GPIOCN.0.
Step 5. Enable Global Inputs (INPUTEN = ‘1).

Port pins can be used as digital inputs or outputs. To configure a Port pin as a digital input, write ‘0’ to the corresponding bit in register PnMDOUT, and write ‘1’ to the corresponding Port latch (register Pn). When a Port pin is read, the actual voltage at the pin is used to determine a logic 0 or logic 1 value; the Port latch is write-only.

Digital output pins can be configured to open-drain or push-pull. In open drain mode (corresponding bit in PnMDOUT is set to ‘0’), the low output driver is turned on when the Port latch is a logic 0 and turned off when the Port latch is a logic 1. The high output driver is always off, regardless of the Port latch setting. In open drain mode, an output port pin becomes a high impedance input when the Port latch is a logic 1. An external pullup resistor is recommended if the pin is intended for use as an output. This mode is useful when interfacing to 5V logic.

Each port pin has an internal weak pullup that is enabled when the WEAKPUD bit ‘0’, the port output mode is configured as open-drain, and the port latch is a logic 1 (pin is a high impedance input). The weak pullup is disabled if the pin is configured to push-pull mode or the Port latch is a logic 0 to avoid unnecessary power dissipation.

In push-pull mode (corresponding bit in PnMDOUT is set to ‘1’), one of the output drivers will always remain on. When the Port latch is a logic 0, the low output driver is turned on and the high output driver is off. When the Port latch is a logic 1, the low output driver is turned off and the high output driver is turned on. Note that in push-pull mode, the voltage at the port pin will reflect the logic level of the output Port latch. This mode cannot be used to drive logic levels higher than VIO or VDD.

After each port pin is properly configured as an input or output, special signals can be routed to select port pins. Special signals include /SYSCLK on P0.0, XTAL2 clock input on P0.3, UART TX on P0.4, and UART RX on P0.5. The /SYSCLK signal can be routed to P0.0 by setting GPIOCN.0 to ‘1’. The XTAL2 clock input is always routed to P0.3. The UART TX signal is always enabled, and ANDed with the P0.4 latch. When using the UART, the P0.4 Port latches should be logic ‘1’ to allow the UART to control the TX pin. If the Port latch is written ‘0’ at any time, the TX signal will be forced to a logic 0. When the UART is not used, the value of the TX signal is parked at logic 1 and P0.4 can be used as GPIO.

Important Note: Setting the INPUTEN bit in GPIOCN to ‘1’ globally enables digital inputs. Until global inputs are enabled, all port pins on the device remain as output only and cannot be used to sense the logic level on the port pin. INPUTEN must be set to ‘1’ in order to use UART RX, XTAL2, or the /INT0 input.

11.2. General Purpose Port I/O

Port0, Port2, and Port3 are accessed through corresponding special function registers (SFRs) that are both byte addressable and bit addressable. When writing to a Port, the value written to the SFR is latched to maintain the output data value at each pin. When reading, the logic levels of the Port's input pins are returned if INPUTEN is set to ‘1’. The exception to this is the execution of the read-modify-write instructions. The read-modify-write instructions when operating on a Port SFR are the following: ANL, ORL, XRL, JBC, CPL, INC, DEC, and DJNZ. The MOV, CLR and SETB instructions are also read-modify-write when the destination is an individual bit in a Port SFR. For these instructions, the value of the register (not the pin) is read, modified, and written back to the SFR.

SFR Definition 11.4. P2: Port2

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
—	—	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							(bit addressable)	0xA0

Bits7–6: Unused. Read = 00b. Write = don't care.
Bits5–0: P2.[5:0]
Write - Output appears on I/O pins.
0: Logic Low Output.
1: Logic High Output (high impedance if corresponding P2MDOUT.n bit = 0).
Read - Always reads '0' if INPUTEN = '0'. Otherwise, directly reads Port pin.
0: P2.n pin is logic low.
1: P2.n pin is logic high.

SFR Definition 11.5. P2MDOUT: Port2 Output Mode

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
—	—							00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
								0xA6

Bits7–6: Unused. Read = 00b. Write = don't care.
Bits5–0: Output Configuration Bit for P2.5–2.0:
0: P2.0 Output is open-drain.
1: P2.0 Output is push-pull.

SFR Definition 11.6. P3: Port3

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
—	—	—	—	—	—	—	P3.0	11111111
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address:
							(bit addressable)	0xB0

Bits7–1: Unused. Read = 0000000b. Write = don't care.
Bit0: P3.0
Write - Output appears on I/O pins.
0: Logic Low Output.
1: Logic High Output (high impedance if corresponding P3MDOUT.n bit = 0).
Read - Always reads '0' if INPUTEN = '0'. Otherwise, directly reads Port pin.
0: P3.n pin is logic low.
1: P3.n pin is logic high.

SFR Definition 11.7. P3MDOUT: Port3 Output Mode

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	Reset Value
—	—	—	—	—	—	—		00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	SFR Address: 0xA7

Bits7–1: Unused. Read = 0000000b. Write = don't care.
Bit0: Output Configuration Bit for P3.0:
0: P3.0 Output is open-drain.
1: P3.0 Output is push-pull.

12.3. USB Register Access

The USB0 controller registers listed in Table 12.2 are accessed through two SFRs: USB0 Address (USB0ADR) and USB0 Data (USB0DAT). The USB0ADR register selects which USB register is targeted by reads/writes of the USB0DAT register. See Figure 12.2.

Endpoint control/status registers are accessed by first writing the USB register INDEX with the target endpoint number. Once the target endpoint number is written to the INDEX register, the control/status registers associated with the target endpoint may be accessed. See the “Indexed Registers” section of Table 12.2 for a list of endpoint control/status registers.

Important Note: The USB clock must be active when accessing USB registers.

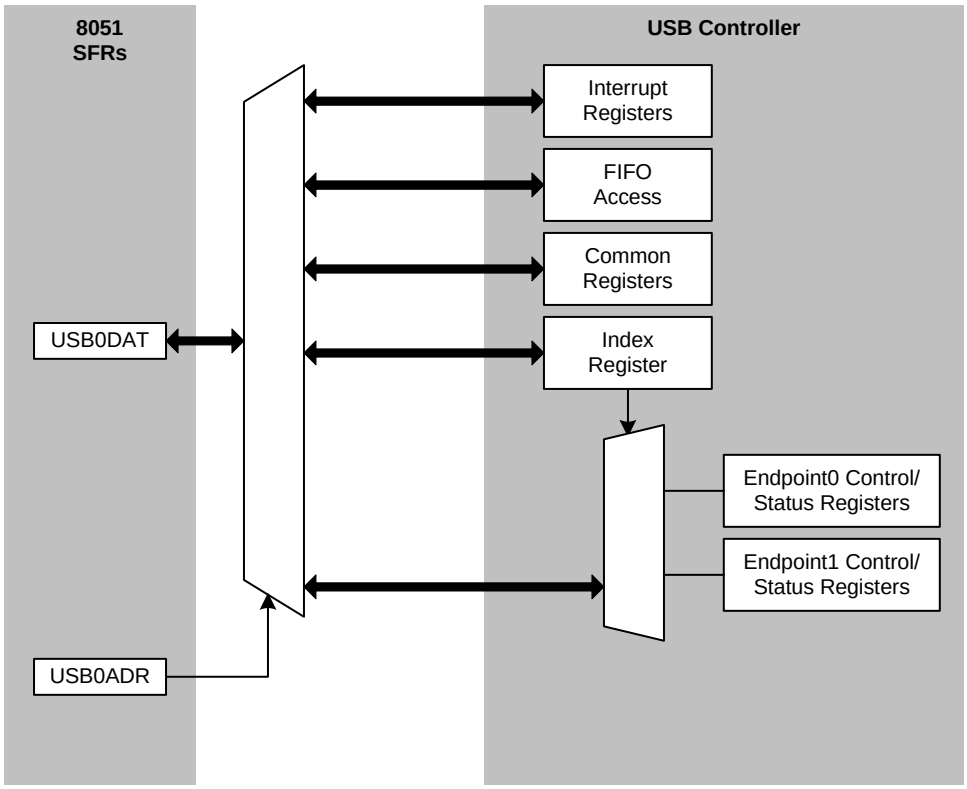

Figure 12.2. USB0 Register Access Scheme

Table 12.2. USB0 Controller Registers

USB Register Name	USB Register Address	Description	Page Number
Interrupt Registers			
IN1INT	0x02	Endpoint0 and Endpoint1 IN Interrupt Flags	101
OUT1INT	0x04	Endpoint1 OUT Interrupt Flag	101
CMINT	0x06	Common USB Interrupt Flags	102
IN1IE	0x07	Endpoint0 and Endpoint1 IN Interrupt Enables	102
OUT1IE	0x09	Endpoint1 OUT Interrupt Enable	103
CMIE	0x0B	Common USB Interrupt Enable	103
Common Registers			
FADDR	0x00	Function Address	97
POWER	0x01	Power Management	99
FRAMEL	0x0C	Frame Number Low Byte	100
FRAMEH	0x0D	Frame Number High Byte	100
INDEX	0x0E	Endpoint Index Selection	92
CLKREC	0x0F	Clock Recovery Control	94
FIFOn	0x20-0x21	Endpoints0-1 FIFOs	96
Indexed Registers			
E0CSR	0x11	Endpoint0 Control / Status	106
EINCSRL		Endpoint IN Control / Status Low Byte	110
EINCSRH	0x12	Endpoint IN Control / Status High Byte	111
EOUTCSRL	0x14	Endpoint OUT Control / Status Low Byte	113
EOUTCSRH	0x15	Endpoint OUT Control / Status High Byte	114
E0CNT	0x16	Number of Received Bytes in Endpoint0 FIFO	107
EOUTCNTL		Endpoint OUT Packet Count Low Byte	114
EOUTCNTH	0x17	Endpoint OUT Packet Count High Byte	114

12.5. FIFO Management

256 bytes of on-chip XRAM are used as FIFO space for USB0. This FIFO space is split between Endpoint0 and Endpoint1 as shown in Figure 12.3. FIFO space allocated for Endpoint1 is split into an IN and an OUT endpoint.

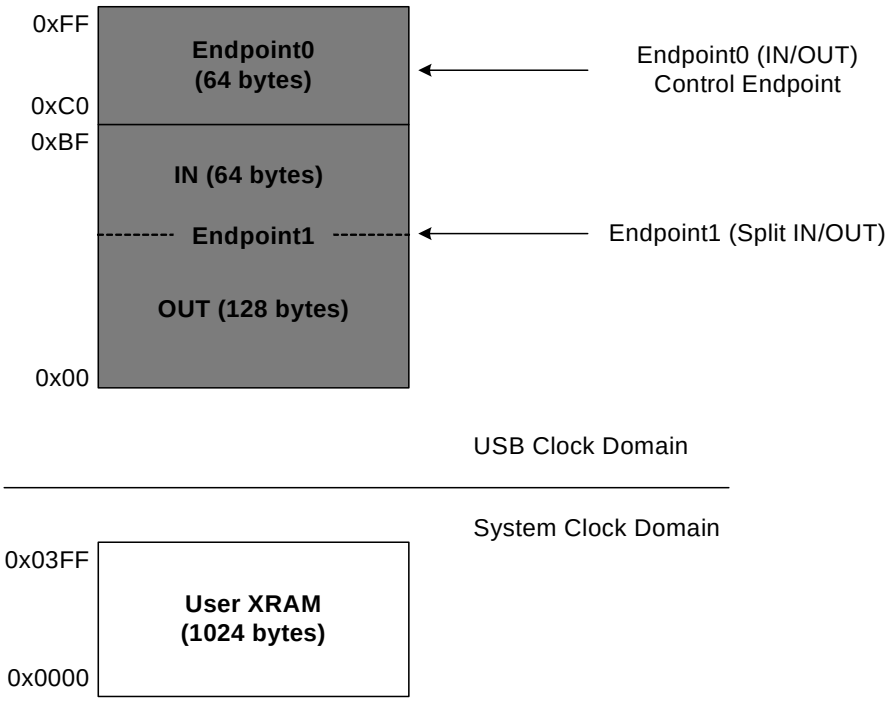

Figure 12.3. USB FIFO Allocation

12.5.1. FIFO Split Mode

The FIFO space for Endpoint1 is split such that the upper 64 bytes of the FIFO space is used by the IN endpoint, and the lower 128 bytes is used by the OUT endpoint.

The FIFO space for Endpoint0 is not split. The 64 byte FIFO space forms a single IN *or* OUT FIFO. Endpoint0 can transfer data in one direction at a time. The endpoint direction (IN/OUT) is determined by the DIRSEL bit in the corresponding endpoint's EINCSRH register (see Figure 12.20).

12.5.2. FIFO Double Buffering

The Endpoint1 FIFO can be configured for double-buffered mode. In this mode, the maximum packet size is halved and the FIFO may contain two packets at a time. This mode is only available for Endpoint1. Double buffering may be enabled for the IN Endpoint and/or the OUT endpoint. See Table 12.3 for a list of maximum packet sizes for each FIFO configuration.

Table 12.3. FIFO Configurations

Endpoint Number	Split Mode Enabled?	Maximum IN Packet Size (Double Buffer Disabled / Enabled)	Maximum OUT Packet Size (Double Buffer Disabled / Enabled)
0	N/A	64	
1	Y	64 / 32	128 / 64

USB Register Definition 12.18. E0CNT: USB0 Endpoint 0 Data Count

R	R	R	R	R	R	R	R	Reset Value
-	E0CNT							00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	USB Address: 0x16

Bit7: Unused. Read = 0. Write = don't care.

Bits6–0: E0CNT: Endpoint 0 Data Count

This 7-bit number indicates the number of received data bytes in the Endpoint 0 FIFO. This number is only valid while bit OPRDY is a '1'.

USB Register Definition 12.21. EOUTCSRL: USB0 OUT Endpoint Control Low Byte

W	R/W	R/W	R/W	R	R/W	R	R/W	Reset Value
CLRDT	STSTL	SDSTL	FLUSH	DATERR	OVRUN	FIFOFUL	OPRDY	00000000
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	USB Address: 0x14

Bit7: CLRDT: Clear Data Toggle
Write: Software should write ‘1’ to this bit to reset the OUT endpoint data toggle to ‘0’.
Read: This bit always reads ‘0’.

Bit6: STSTL: Sent Stall
Hardware sets this bit to ‘1’ when a STALL handshake signal is transmitted. This flag must be cleared by software.

Bit5: SDSTL: Send Stall
Software should write ‘1’ to this bit to generate a STALL handshake. Software should write ‘0’ to this bit to terminate the STALL signal. This bit has no effect in ISO mode.

Bit4: FLUSH: FIFO Flush
Writing a ‘1’ to this bit flushes the next packet to be read from the OUT endpoint FIFO. The FIFO pointer is reset and the OPRDY bit is cleared. If the FIFO contains multiple packets, software must write ‘1’ to FLUSH for each packet. Hardware resets the FLUSH bit to ‘0’ when the FIFO flush is complete.

Bit3: DATERR: Data Error
In ISO mode, this bit is set by hardware if a received packet has a CRC or bit-stuffing error. It is cleared when software clears OPRDY. This bit is only valid in ISO mode.

Bit2: OVRUN: Data Overrun
This bit is set by hardware when an incoming data packet cannot be loaded into the OUT endpoint FIFO. This bit is only valid in ISO mode, and must be cleared by software.
0: No data overrun.
1: A data packet was lost because of a full FIFO since this flag was last cleared.

Bit1: FIFOFUL: OUT FIFO Full
This bit indicates the contents of the OUT FIFO. If double buffering is enabled for the endpoint (DBIEN = ‘1’), the FIFO is full when the FIFO contains two packets. If DBIEN = ‘0’, the FIFO is full when the FIFO contains one packet.
0: OUT endpoint FIFO is not full.
1: OUT endpoint FIFO is full.

Bit0: OPRDY: OUT Packet Ready
Hardware sets this bit to ‘1’ and generates an interrupt when a data packet is available. Software should clear this bit after each data packet is unloaded from the OUT endpoint FIFO.

13.2. Data Format

UART0 has a number of available options for data formatting. Data transfers begin with a start bit (logic low), followed by the data bits (sent LSB-first), a parity or extra bit (if selected), and end with one or two stop bits (logic high). The data length is variable between 5 and 8 bits. A parity bit can be appended to the data, and automatically generated and detected by hardware for even, odd, mark, or space parity. The stop bit length is selectable between 1 and 2 bit times, and a multi-processor communication mode is available for implementing networked UART buses. All of the data formatting options can be configured using the SMOD0 register, shown in SFR Definition 13.2. Figure 13.2 shows the timing for a UART0 transaction without parity or an extra bit enabled. Figure 13.3 shows the timing for a UART0 transaction with parity enabled (PE0 = 1). Figure 13.4 is an example of a UART0 transaction when the extra bit is enabled (XBE0 = 1). Note that the extra bit feature is not available when parity is enabled, and the second stop bit is only an option for data lengths of 6, 7, or 8 bits.

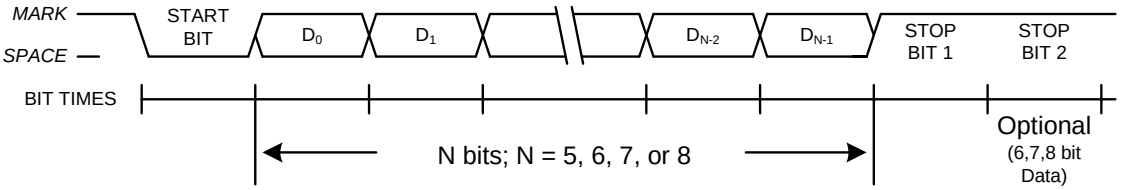

Figure 13.2. UART0 Timing Without Parity or Extra Bit

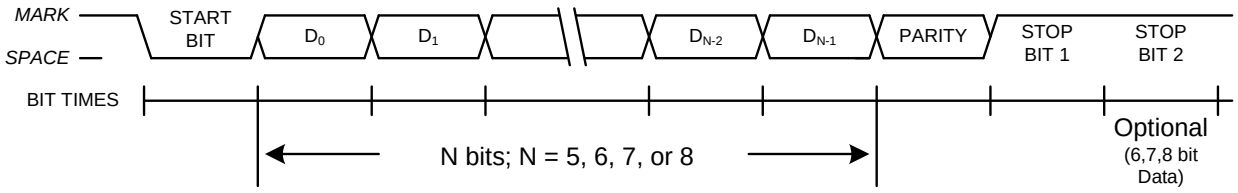

Figure 13.3. UART0 Timing With Parity

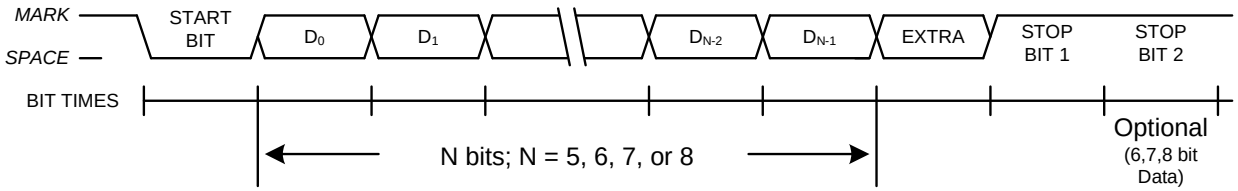

Figure 13.4. UART0 Timing With Extra Bit

DOCUMENT CHANGE LIST

Revision 0.5 to Revision 1.0

- Updated Section “1. System Overview” on page 13 and Table 1.1, “Product Selection Guide,” on page 13.
  - Changed “-GQ” references to “-GM”
- Added Figure 1.3. "Typical Connections for the C8051F326" on page 16 and Figure 1.4. "Typical Connections for the C8051F327" on page 16.
- Changed Figure 4.5. "Typical C8051F327 QFN-28 Landing Diagram" on page 31 to show ground connection on Pin 3.
- Replaced TBDs with values in Table 5.1, “Voltage Regulator Electrical Specifications,” on page 31.
- Replaced TBDs with values in Table 7.1, “Reset Electrical Characteristics,” on page 62.
- Moved USB Active characteristics from Table 3.1, “Global DC Electrical Characteristics,” on page 24 to Table 12.4, “USB Transceiver Electrical Characteristics,” on page 115.
- Added port information to Figure 11.1. "Port I/O Functional Block Diagram" on page 79.
- Added read/write state description to bits 7–6 in SFR Definition 11.4. “P2: Port2” on page 83.
- Clarified description of read state for bits 7–3 in USB Register Definition 12.10. “FRAMEH: USB0 Frame Number High” on page 100.
- Clarified description of read state for bits 7–2 in USB Register Definition 12.24. “EOUTCNTH: USB0 OUT Endpoint Count High” on page 114.
- Standardized descriptions for “unused” and “reserved” bits in SFR Definitions throughout document.

Revision 1.0 to Revision 1.1

- Updated package and land pattern drawings.