SECTION 7: POWER MANAGEMENT

The High–Speed Microcontroller has several features that relate to power consumption and management. They provide a combination of controlled operation in unreliable power applications and reduced power consumption in portable or battery powered applications. The range of features is shown below with details to follow.

POWER MANAGEMENT
EARLY WARNING POWER FAIL INTERRUPT
POWER FAIL/POWER ON RESET
BAND–GAP SELECT
WATCHDOG WAKE UP FROM IDLE

POWER SAVING
IDLE MODE
STOP MODE
RING WAKE UP FROM STOP
POWER MANAGEMENT MODES

PRECISION VOLTAGE MONITOR

The High–Speed Microcontroller uses a precision band–gap reference and other analog circuits to monitor the state of the power supply during power–up and power–down transitions. Other microcontroller systems would require external circuits to perform these functions. The band–gap reference provides a precise voltage to compare with V_{CC}. When V_{CC} begins to drop, the Power Monitor compares it to its reference. This enables the analog circuits to detect when V_{CC} passes through predetermined thresholds, V_{PFW} and V_{RST}. These are specified in the individual product data sheets.

EARLY WARNING POWER FAIL INTERRUPT

Devices which incorporate the precision voltage reference have the ability to generate a power–fail interrupt and/or reset in response to a low supply voltage. When V_{CC} reaches the V_{PFW} threshold, the microcontroller can generate an power–fail Interrupt. This early warning of supply voltage failure allows the system time to save critical parameters in nonvolatile memory and put external functions in a safe state.

The power–fail interrupt is optional and is enabled using the Enable Power Fail Warning Interrupt (EPFI) bit at WDCON.5. If enabled, V_{CC} dropping below V_{PFW} will cause the device to vector to address 33h. The Power–fail Interrupt status bit ,PFI (WDCON.4), will be set any time V_{CC} transitions below V_{PFW}. This flag is not cleared when V_{CC} is above V_{PFW}, and software should clear it immediately after reading it. As long as the condition exists, PFI will be immediately set again by hardware.

A typical application of the PFI is to place the device into a "safe mode" when a power loss appears imminent. When the interrupt occurs, the code vectors to location 33h. At this time, software can disable the interrupt, save any critical data, clear PFI, then continually poll the status of the power supply via the PFI flag. As long as PFI is set, power is still below V_{PFW}. If power returns to the proper level, PFI will not be set once cleared by software. This indicates a safe operating condition. If power continues to fall, a power–fail reset will be invoked automatically.

POWER–FAIL RESET

Devices which incorporate the power–fail reset will automatically invoke a reset when V_{CC} drops below V_{RST}. This will halt device operation, and place all outputs in their reset state. This state will continue to be held until V_{CC} drops below the voltage necessary to power the port pins. Because V_{RST} is lower than V_{PFW}, the microcontroller has the option to use the power–fail interrupt to place the device into a "safe" state before the device halts operation with a power–fail reset. This feature is automatic on devices which incorporate the power–fail reset feature, and cannot be disabled.

POWER ON RESET

When V_{CC} is applied to a system using the High–Speed Microcontroller, the device will hold itself in reset until power is within tolerance and stable. An internal band–gap reference provides a highly accurate and stable means of detecting power supply levels. It requires no external circuits to accomplish this. As power rises, the processor will stay in a reset state until $V_{CC} > V_{RST}$. As V_{CC} rises above V_{RST}, internal analog circuits will detect this and activate the on–chip crystal oscillator. On–chip hardware will then count 65536 oscillator clocks. During this count, V_{CC} must remain above V_{RST} or the process restarts. If an off–chip clock source is used, then clock counting still begins once $V_{CC} > V_{RST}$. This count period is used to make certain that power is within tolerance, and that the oscillator has time to stabilize. This provides a very controlled and predictable start–up condition.

Once the 65536 count period has elapsed, the reset condition is removed automatically, and software execution will begin at the reset vector location of 0000h. Software will be able to detect the power on reset condition using the Power–On Reset (POR) flag. POR is located at WDCON.6. This bit will be high to indicate that a Power–on Reset has occurred. It should then be cleared by software.

The complete power cycle operation is shown in Figure 7–1. Note that the interrupt threshold is fixed, but the interrupt itself is optional. Reset thresholds are also fixed and the reset operation is transparent. It requires no external components and no action by software to control reset operation.

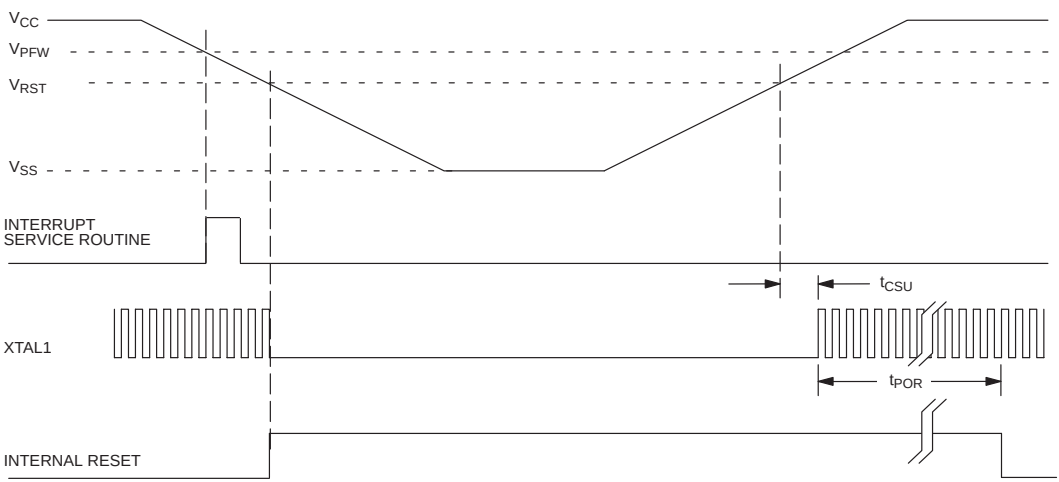
BAND–GAP SELECT

When present, the band–gap reference will provide a precise voltage reference for the power–fail monitor circuitry. The band–gap is normally disabled automatically upon entering Stop mode to provide the lowest power state. Since the band–gap is inactive, there can be no power–fail interrupt and no power–fail reset, similar to a traditional 8051.

If the use of the power–fail features are desired in Stop mode, the BGS bit (EXIF, 91h) may be used. When set to a logic 1 by software, the band–gap reference and associated power monitor circuits will remain active in Stop mode. The price of this feature is higher power supply current requirements. In Stop mode with the band–gap reference disabled (default), the processor draws approximately 1 μA. With the band–gap enabled, it draws approximately 50 μA.

BGS allows the user to decide whether the control circuitry and its associated power consumption are needed. If the application is such that power will not fail while in Stop or if it does not matter that power fails, the BGS should be set to 0 (default). If power can fail at any time and cause problems, the BGS should be set to 1.

POWER CYCLE OPERATION Figure 7–1



WATCHDOG WAKE UP

The Watchdog Wake up is more of an application than a feature. It allows a system to enter the Idle mode for power savings, then to wake up periodically to sample the external world. Idle mode is a low power state described below. Any of the programmable timers can perform this function, but the Watchdog allows a much longer period to be selected. At 12 MHz, the maximum Watchdog time–out is over 5.5 seconds. This contrasts with 0.78 seconds using the 16–bit timers. Software that uses the Watchdog as a wake up alarm should only enable the Watchdog Interrupt and not the Reset. Note that the Watchdog cannot be used to wake the system while in Stop mode since no clocks are running. Stop mode is described below.

POWER MANAGEMENT SUMMARY

The following is a summary of the power management bits and those that are useful or related. They are contained in the register locations WDCON;D8h, EIE;E8h, EXIF;91h, and PCON; 87h.

WDCON.6 POR – Power on Reset. Hardware will set this bit on a power up condition. Software can read it, but must clear it manually. This bit assists software in determining the cause of a reset.

WDCON.5 EPFI – Enable Power–fail Interrupt. Setting this bit to 1 enables the Power–fail interrupt. This will occur when V_{CC} drops to approximately 4.5 volts, and the processor vectors to location 33h. Setting this bit to a 0 turns off the Power–fail Interrupt.

WDCON.4 PFI – Power Fail Interrupt Flag. Hardware will set this bit to a 1 when a Power–fail condition occurs. Software must clear the bit manually. Writing a 1 to this bit will force an interrupt, if enabled.

WDCON.3 WDIF – Watchdog Interrupt Flag. If the Watchdog Interrupt is enabled (EIE.4), hardware will set this bit to indicate that the Watchdog Interrupt has occurred. If the interrupt is not enabled, this bit indicates that the time–out has passed. If the Watchdog Reset is enabled (WDCON.1), the user has 512 clocks to strobe the Watchdog prior to a reset. Software or any reset can clear this flag.

WDCON.2 WTRF – Watchdog Timer Reset Flag. Hardware will set this bit when the Watchdog Timer causes a reset. Software can read it, but must clear it manually. A Power–fail Reset will also clear the bit. This bit assists software in determining the cause of a reset. If EWT=0, the Watchdog Timer will have no affect on this bit.

WDCON.1 EWT – Enable Watchdog Timer Reset. Setting this bit will turn on the Watchdog Timer Reset function. The Interrupt will not occur unless the EWDI bit in the EIE register is set. A reset will occur according to the WD1 and WD0 bits in the CKCON register. Setting this bit to a 0 will disable the reset but leave the timer running.

WDCON.0 RWT – Reset Watchdog Timer. This bit serves as the strobe for the Watchdog function. During the time–out period, software must set the RWT bit if the Watchdog is enabled. Failing to set the RWT will cause a reset when the time–out has elapsed. There is no need to set the RWT bit to a 0 because it is self–clearing.

EIE.4 EWDI – Enable Watchdog Interrupt. Setting this bit in software enables the Watchdog Interrupt.

EXIF.0 BGS – Band–Gap Select. Setting this bit to a 1 will allow the use of the Band–gap voltage reference while in Stop mode. Since this function uses as much as 50 μA, the band–gap is optional in Stop mode. Setting this bit to a 0 will turn off the Band–gap while in Stop mode. When BGS=0, no Power–fail interrupt or Power–fail Reset will be available in Stop mode.

PCON.1 STOP. When this bit is set, the program stops execution, clocks are stopped, and the CPU enters power–down mode.

PCON.0 IDLE. Program execution halts leaving timers, serial ports, and clocks running.

EXIF.2 RGMD – Ring Oscillator Mode. Hardware will set this status bit to a 1 when the clock source is the Ring Oscillator. Hardware will set this status bit to a 0 when the crystal is the clock source. Refer to RGSL below for operation of the Ring Oscillator.

EXIF.1 RGSL – Ring Oscillator Select. When set to a 1 by software, the High–Speed Microcontroller will use a Ring Oscillator to come out of Stop mode without waiting for crystal start–up. This allows an instantaneous start–up when coming out of Stop mode. It is useful if software needs to perform a short task, then return to Stop. It is also useful if software must respond quickly to an external event. After the crystal has performed 65,536 cycles, hardware will switch to the crystal as its clock source. The RGMD status bit reports on this changeover. When RGSL is set to a 0, the High–Speed Microcontroller will delay software execution until after the 65,536 clock crystal startup time. RGSL is only cleared by a power–on reset and is not altered by other forms of reset.

POWER SAVING

The High–Speed Microcontroller is implemented using full CMOS circuitry for low power operation. It is fully static so the clock speed can be run down to DC. Like other CMOS, the power consumption is also a function of operating frequency. Although the High–Speed Microcontroller is designed for maximum performance, it also provides improved power versus work relationships compared with standard 8051 devices. These topics are discussed in detail below.

The High–Speed Microcontroller provides two modes (other than operating) that allow power conservation. They are similar, but have different merits and drawbacks. These modes are Idle and Stop. In the original 8051, the Stop mode is called Power Down. These modes are invoked in the same manner as the original 8051 series.

IDLE MODE

Idle mode suspends all CPU processing by holding the program counter in a static state. No program values are fetched and no processing occurs. This saves considerable power versus full operation. The virtue of Idle mode is that it uses half the power of the operating state, yet reacts instantly to any interrupt conditions. All clocks remain active so the timers, Watchdog, Serial Port, and Power Monitor functions are all working. Since all clocks are running, the CPU can exit the Idle state using any of the interrupt sources.

Software can invoke the Idle mode by setting the IDLE bit in the PCON register at location 87h. The bit is located at PCON.0. The instruction that executes this step will be the last instruction prior to freezing the program counter. Once in Idle, all resources are preserved. There are two ways to exit the Idle mode. First, any interrupt (that is enabled) will cause an exit. This will result in a jump to the appropriate interrupt vector. The IDLE bit in the PCON register will be cleared automatically. On returning from this vector using the RETI instruction, the next address will be the one immediately after the instruction that invoked the Idle state.

The Idle mode can also be removed using a reset. Any of the three reset sources can do this. On receiving the reset stimulus, the CPU will be placed in a reset state and the Idle condition cleared. When the reset stimulus is removed, software will begin execution as for any reset. Since all clocks are active, there will be no delay after the reset stimulus is removed. Note that if enabled, the Watchdog Timer continues to run during Idle and must be supported.

STOP MODE

Stop mode is the lowest power state that the High–Speed Microcontroller can enter. This is achieved by stopping all on–chip clocks, resulting in a fully static condition. No processing is possible, timers are stopped, and no serial communication is possible. Processor operation will halt on the instruction that sets the STOP bit. The internal amplifier that excites the external crystal will be disabled, halting crystal oscillation in Stop mode. Table 7–1 shows the state of the processor pins in Idle and Stop modes.

Stop mode can be exited in two ways. First, like the 8052 microcontrollers, a non–clocked interrupt such as the external interrupts or the power–fail interrupt can be used. Clocked interrupts such as the watchdog timer, internal timers, and serial ports will not operate in Stop mode. Note that the band–gap reference must be enabled in order to use the power–fail interrupt to exit Stop mode, which will increase Stop mode current. Processor operation will resume with the fetching of the interrupt vector associated with the interrupt that caused the exit from Stop mode. When the interrupt service routine is complete, an RETI will return the program to the instruction immediately following the one that invoked the Stop mode.

A second method of exiting Stop mode is with a reset. The watchdog timer reset is not available as a reset source because no timers are running in Stop mode. An external reset via the RST pin will unconditionally exit the device from Stop mode. If the BGS bit is set, the device will provide a reset while in Stop mode if V_{CC} should drop below the V_{RST} level. If the BGS bit is 0, then a dip in power below V_{RST} will not cause a reset. For example, if V_{CC} should drop to a level of V_{RST}–0.5V, then return to the full level, no reset will be generated. For this reason, use of the band–gap reference is recommended if a brownout condition is possible in Stop mode. If power fails completely (V_{CC}=0V), then a power on reset will still be performed when V_{CC} is reapplied regardless of the state of the BGS bit. Processor operation will resume execution from address 0000h like any other reset.

CRYSTAL RESUME FROM STOP MODE

If the microcontroller does not contain a ring oscillator, or if the RGSL bit is 0, a device exiting Stop mode must restart operation using the external crystal as a clock source. The device will experience a power–on reset delay of 65536 external clock cycles to allow the crystal to begin oscillation and the frequency to stabilize. Once this delay is complete, software will begin execution from either address 0000h or the appropriate interrupt vector, depending on the stimulus to exit Stop mode. The same 65536 external clock cycle delay is performed if an external crystal oscillator is used instead of an external crystal.

PIN STATES IN POWER SAVING MODES Table 7–1

DEVICE	MODE	ALE	PSEN	P0 (AD0–7)	P1	P2	P3
DS80C310 DS80C320	Idle or Stop	1	1	Latched[1]	Port data[2]	Latched[3]	Port data[2]
All Others Internal program execution	Idle or Stop	1	1	Port data[2]	Port data[2]	Port data[2]	Port data[2]
All Others External program execution	Idle	1	1	Latched[1]	Port data[2]	Latched[3]	Port data[2]
All Others External program execution	Stop	1	1	Port data[2]	Port data[2]	Port data[4]	Port data[2]

[1] Port exhibits opcode following instruction that sets the Stop bit. Port 0 is operating in true bidirectional mode, and will drive both a logic 1 and a logic 0.
[2] Port reflects data stored in corresponding Port SFR. Port 0 functions as an open–drain output in this mode.
[3] Port exhibits address MSB of opcode following instruction that sets the Stop bit.
[4] Port reflects data stored in corresponding Port SFR. In this mode, the port uses weak pullups. If a bit in the P2 SFR is a 1, the corresponding device pin will transition slowly to a high when the reset state is entered.

RING OSCILLATOR WAKE UP FROM STOP

A typical low power application is to keep the processor in Stop mode most of the time. Periodically, the system will wake up (using an external interrupt), take a reading of some condition, then return to sleep. The duration of full power operation is as short as possible. One disadvantage to this method is that the clock must be restarted prior to performing a meaningful operation. This start–up period is a waste of time and power since no work can be performed. The High–Speed Microcontroller provides an alternative.

If the Ring Select (RGSL) is enabled, the High–Speed Microcontroller can exit Stop mode running from an internal Ring Oscillator. Upon receipt of an interrupt , this oscillator can start instantaneously, allowing software execution to begin immediately while the oscillator is stabilizing. Once 65,536 clock cycles have been detected, the CPU will automatically switch to the normal oscillator as its clock source. However, if the required interrupt response is very short, the software can re–enter Stop mode before the crystal is even stable. In this case, Stop mode can be invoked and both oscillators will be stopped.

SPEED REDUCTION

The High–Speed Microcontroller is a fully CMOS 8051 compatible microcontroller. It can use significantly less power than other 8051 versions because it is more efficient. As an average, software will run 2.5 times faster on the High–Speed Microcontroller than on other 8051 derivatives. Thus the same job can be accomplished by slowing down the crystal by a factor of 2.5. For example, an existing 8051 design that runs at 12 MHz can run at approximately 4.8 MHz on the High–Speed Microcontroller. At this reduced speed, the High–Speed Microcontroller will have lower power consumption than an 8051, yet perform the same job.

Using the 2.5X factor, Table 7–2 shows the approximate speed at which the High–Speed Microcontroller can accomplish the same work as an 8051. The exact improvement will vary depending on the actual instruction mix. Available crystal speeds must also be considered. Refer to Section 16 for information on instruction timing.

CRYSTAL VS MIPS COMPARISON Table 7–2

ORIGINAL 8051 CRYSTAL SPEED	MIPS	HIGH–SPEED MICROCONTROLLER CRYSTAL SPEED
3.57 MHz	0.3	1.4 MHz
7.37 MHz	0.6	2.9 MHz
11.0592 MHz	0.9	4.4 MHz
14.318 MHz	1.2	5.7 MHz
16 MHz	1.3	6.4 MHz
20 MHz	1.6	8 MHz
24 MHz	2.0	9.6 MHz
33 MHz	2.7	13.2 MHz
40 MHz	3.3	16 MHz

POWER MANAGEMENT MODES

Power consumption in CMOS microcontrollers is a function of operating frequency. The Power Management Mode (PMM) feature, available with some members of the High–Speed Microcontroller family, allows software to dynamically match operating frequency and current consumption with the need for processing power. Instead of the default 4 clocks per machine cycle, power management mode 1 (PMM1) and power management mode 2 (PMM2) utilize 64 and 1024 clocks per cycle respectively to conserve power.

A number of special features have been added to enhance the function of the power management modes. The switchback feature allows the device to almost instantaneously return to divide by 4 mode upon acknowledgment of an external interrupt or a falling edge on a serial port receiver pin. The advantages of this become apparent when one calculates the increased interrupt service time of a device operating in PMM. In addition, a device operating in PMM would normally be unable to sample an incoming serial transmission to properly receive it. The switchback feature, explained below, allows a device to return to divide by 4 operation in time to receive incoming serial port data and process interrupts with no loss in performance.

The DS87C520 and DS87C530 incorporate a Status register (STATUS;C5h) to prevent the device from accidentally reducing the clock rate during the servicing of an external interrupt or serial port activity. This register can be interrogated to determine if a high priority, low priority, or power fail interrupt is in progress, or if serial port activity is occurring. Based on this information the software can delay or reject a planned change in the clock divider rate.

In addition, the DS87C520 and DS87C530 has the capability to operate from the internal ring oscillator during normal operation, not only during the crystal warm–up period. Table 7–3 summarizes the new control bits associated with the power management features.

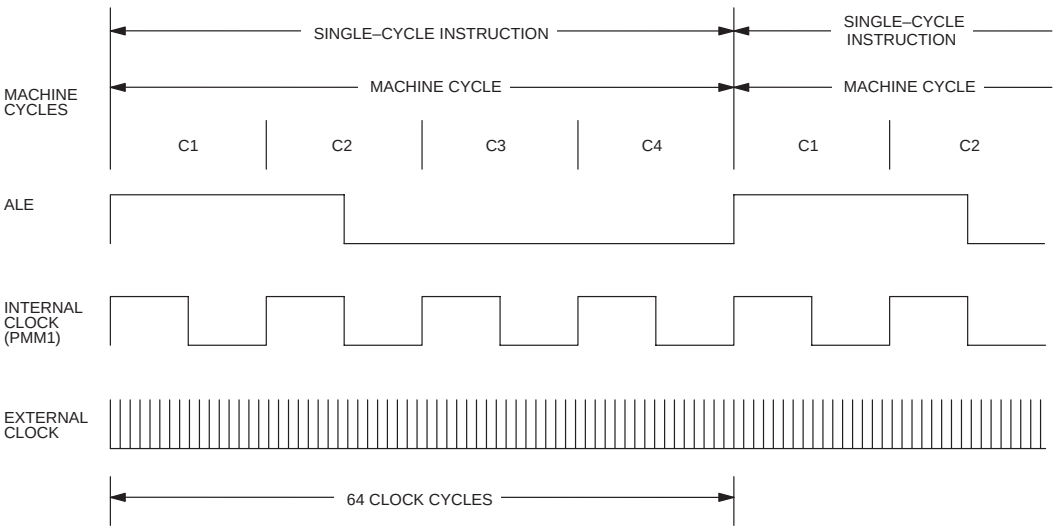
POWER MANAGEMENT AND STATUS BIT SUMMARY Table 7–3

BIT NAME	LOCATION	FUNCTION	RESET STATE	READ/WRITE ACCESS
CD1, CD0	PMR.7–6	Clock Divider Control CD1 CD0 Osc Cycles per Machine Cycle 0 0 Reserved 0 1 4 (Reset Default) 1 0 64 (PMM1) 1 1 1024 (PMM2)	0,1	Write: 0,1 anytime; 1,0 & 1,1 only when previously in 0,1 state. Unrestricted read.
SWB	PMR.5	Switchback Enable 0=Interrupts and serial port activity will not affect clock divider control bits 1=Enabled Interrupts and serial port activity will cause a switchback	0	Unrestricted
PIP	STATUS.7	Power Fail Interrupt Status 0=No power fail interrupt in progress 1=Power fail interrupt in progress	0	Read Only
HIP	STATUS.6	High Priority Interrupt Status 0=No high priority interrupt in progress 1=High priority interrupt in progress	0	Read Only
LIP	STATUS.5	Low Priority Interrupt Status 0=No low priority interrupt in progress 1=Low priority interrupt in progress	0	Read Only
SPTA1	STATUS.3	Serial Port 1 Transmitter Activity Status 0=Serial port 1 transmitter inactive 1=Serial port 1 transmitter active	0	Read Only
SPRA1	STATUS.2	Serial Port 1 Receiver Activity Status 0=Serial port 1 receiver inactive 1=Serial port 1 receiver active	0	Read Only
SPTA0	STATUS.1	Serial Port 0 Transmitter Activity Status 0=Serial port 0 transmitter inactive 1=Serial port 0 transmitter active	0	Read Only
SPRA0	STATUS.0	Serial Port 0 Receiver Activity Status 0=Serial port 0 receiver inactive 1=Serial port 0 receiver active	0	Read Only

POWER MANAGEMENT MODE TIMING

The two power management modes reduce power consumption by internally dividing the clock signal to the device, causing it to operate at a reduced speed. When PMM is invoked, the external crystal will continue to operate at full speed. The difference is that the device uses 16 (PMM1) or 256 (PMM2) external clocks to generate each internal clock cycle (C1, C2, C3 or C4) as opposed to 1 clock per internal clock cycle in divide by 4 mode. This translates to 64 or 1024 external clocks per machine cycle in PMM1 or PMM2, respectively. Relative timing relationships of all signals when the device is operating in PMM1 or PMM2 will remain the same as the 4 cycle timing. Note that all internal functions, on–board timers (including serial port baud rate generation), watchdog timer, and software timing loops will also run at the reduced speed. Most applications will not find it necessary to attend to this much detail, but the information is provided for calculating critical timings. Figure 7–2 demonstrates the internal timing relationships during PMM1.

INTERNAL TIMING RELATIONSHIPS IN PMM1 Figure 7–2



PMM1 and PMM2 are entered and exited by setting the Clock Rate Divider bits (PMR.7–6). In addition, it is possible use the switchback feature to effect a return to the divide by 4 mode from either power management mode. This allows both hardware and software to cause an exit from PMM. Entry to or exit from either PMM must be by the divide by 4 mode. This means that to switch from divide by 64 to divide by 1024 and vice versa, one must first switch back to divide by 4 mode. Attempts to execute an illegal speed change will be ignored and the bits will remain unchanged. It is the responsibility of the software to test for serial port activity before attempting to change speed, as a modification of the clock divider bits during a serial port operation will corrupt the data.

PMM AND PERIPHERAL FUNCTIONS

Timers 0, 1, and 2 will default on reset to a 12 clock per cycle operation to remain compatible with the original 8051 timing. The timers can be individually configured to run at machine cycle timing (divide by 4) by setting the relevant bits in the Clock Control Register (CKCON;8Eh). Because the timers derive their time base from the internal clock, timers 0, 1, and 2 operate at reduced clock rates during PMM. This will also affect the operation of the serial ports in PMM. In general, it is not possible to generate standard baud rates while in PMM, and the user is advised to avoid PMM or use the switchback feature if serial port operation is desired. Table 7–4 shows the effect of the clock divider value on timer operation.

EFFECT OF CLOCK MODES ON TIMER OPERATION Table 7–4

CD1	CD0	OSC. CYCLES PER MACHINE CYCLE	OSC. CYCLES PER TIMER 0/1/2 CLOCK		OSC. CYCLES PER TIMER 2 CLOCK, BAUD RATE GEN.		OSC. CYCLES PER SERIAL PORT CLOCK MODE 0		OSC. CYCLES PER SERIAL PORT CLOCK MODE 2	
			TxM=1	TxM=0	T2M=1	T2M=0	SM2=0	SM2=1	SMOD=0	SMOD=1
0	0	Reserved								
0	1	4	12	4	2	2	12	4	64	32
1	0	64 (PMM1)	192	64	32	32	192	64	1024	512
1	1	1024 (PMM2)	3072	1024	512	512	3072	1024	16,384	8192

SWITCHBACK

The switchback feature solves one of the most vexing dilemmas faced by power–conscious systems. Many applications are unable to use the Stop and Idle modes because they require constant computation. Traditionally, system designers could not reduce the operating speed below that required to process the fastest event. This meant that system architects would be forced to operate their systems at the highest rate of speed even when it was not required.

The switchback feature allows a system to operate at a relatively slow speed, and burst to a faster mode when required by an external event. When this feature is enabled by setting the Switchback Enable bit, SWB, (PMR.5), a qualified interrupt or serial port reception or transmission will cause the device to return to divide by 4 mode. A qualified interrupt is defined as an interrupt which has occurred and been acknowledged. This means that an interrupt must be enabled and also not blocked by a higher priority interrupt. After the event is complete, software can manually return the device to the appropriate PMM. The following sources can trigger a switchback:

external interrupt 0/1/2/3/4/5,
serial start bit detected, Serial Port 0/1,
transmit buffer loaded, Serial Port 0/1,
watchdog timer reset,
power–on reset,
external reset.

In the case of a serial port–initiated switchback, the switchback is not generated by the associated interrupt. This is because a device operating in PMM will not be able to correctly receive a byte of data to generate an interrupt. Instead, a switchback is generated by a serial port reception on the falling edge associated with the start bit, if the associated receiver enable bit (SCON0.4 or SCON1.4) is set. For serial port transmissions, a switchback is generated when the serial port buffer (SBUF0;99h or SBUF1;C1h) is loaded. This ensures the device will be operating in divide by 4 mode when the data is transmitted, and eliminates the need for a write to the CD1, CD0 bits to exit PMM before transmitting. The switchback feature is unaffected by the state of the serial port interrupt flags (RI_0, TI_0, RI_1, TI_1).

The timing of the switchback is dependent on the source. Interrupt–initiated switchbacks will occur at the start of the first C1 cycle following the event initiating the switchback. In PMM, each internal Cx cycle is 16 external clock cycles for PMM1 and 256 cycles for PMM2. If the current instruction in progress is a write to the IE, IP, EIE, or EIP registers, interrupt processing will be delayed until the completion of the following instruction. Serial transmit–initiated switchbacks occur at the start of the instruction following the MOV that loads SBUF0 or SBUF1. Serial reception–initiated switchbacks occur during the Cx cycle in which the falling edge was detected.

There are a few points that must be considered when using a serial port reception to generate a switchback. Under normal circumstances, noise on the line or an aborted transmission would cause the serial port to time–out and the data to be ignored. This presents a problem if the switchback is used, however, because a switchback would occur but there is no indication to the system that one has occurred. If PMM and serial port switchback functions are used in a noisy environment, the user is advised to periodically check if the device has accidentally exited PMM.

A similar problem can occur if multiprocessor communication protocols are used in conjunction with PMM. The High–Speed Microcontroller family supports both the use of the SM2 flag (SCON0.5 or SCON1.5), and the slave address recognition registers (SADDR0;A9h, SADDR1;AAh, SADEN0;B9h, SADEN1;BAh) for multiprocessor communications. The problem is that an invalid address which should be ignored by a particular processor will still generate a switchback. As a result it is not recommended to use a multiprocessor communication scheme in conjunction with PMM. If the system power considerations will allow for an occasional erroneous switchback, a polling scheme can be used to place the device back into PMM.

CLOCK SOURCE SELECTION

The High–Speed Microcontroller family supports three different clock sources for operation. As with most microcontrollers, the device can be clocked from an external crystal using the on–board crystal amplifier, or a clock source can supplied by an external oscillator. In addition, some members of the High–Speed Microcontroller family incorporate an on–board ring oscillator to provide a quick resumption from Stop mode. The ring oscillator is a low power digital oscillator internal to the microcontroller. When enabled, it provides an approximately 2–4 MHz clock source for device operation without external components. The ring oscillator is not as stable as an external crystal, and software should refrain from performing timing dependent operations, including serial port activity, while operating from the ring oscillator.

The ring oscillator provides many advantages to the designers of microcontroller–based systems. One is that it allows Dallas Semiconductor microcontrollers to perform a fast resume from Stop mode, eliminating the crystal warm–up delay when restarting the device. As an added feature, the DS87C520 and DS87C530 will also support extended operation from the ring oscillator, not only during the crystal warm–up period when resuming from Stop. All devices in the High–Speed Microcontroller family must begin operation following a power–on reset from an external clock source, either an external crystal or oscillator. Software can then disable the crystal and run from the lower power ring oscillator. The control and status bits which support the new and/or enhanced features are shown in Table 7–5.

CLOCK CONTROL AND STATUS BIT SUMMARY Table 7–5

BIT NAME	LOCATION	FUNCTION	RESET	WRITE ACCESS
XT/RG	EXIF.3	Crystal/Ring Clock Source Select. This bit is not present on the 80C320. 1=Select crystal or external clock as clock source, 0=Select ring oscillator as clock source	1	0 anytime; 1 when XTUP=1 & XTOFF=0
RGMD	EXIF.2	Ring Oscillator Mode Status. 1=Ring oscillator is current clock source, 0=Crystal or external clock is current clock source.	0	None
RGSL	EXIF.1	Ring Oscillator Select, Stop Mode. 1=Ring oscillator will be the clock source when resuming from Stop mode, 0=Crystal or external clock will be the clock source when resuming from Stop mode Note: Upon completion of crystal warm up period, DS80C320 devices will switch to crystal. DS87C520 and DS87C530 devices will switch to clock source designated by XT/$\overline{RG}$ bit		Unrestricted
XTOFF	PMR.3	Crystal Oscillator Disable. Disables crystal operation after ring mode has been selected. This bit is not present on the 80C320. 1=Crystal amplifier is disabled. 0=Crystal amplifier is enabled. Check XTUP for status.	0	0 anytime; 1 when XT/$\overline{RG}$ =0
XTUP	STATUS.4	Crystal Oscillator Warm Up Status. This bit is not present on the 80C320. 1=Oscillator warm up complete. 0=Oscillator warm up still in progress, crystal not available.	1	None

RING OSCILLATOR RESUME FROM STOP

To achieve the minimum power consumption during periods of processor inactivity, software can place the device into Stop mode. Such systems will typically resume operation using an external interrupt, perform some activity, and then return to Stop mode. Traditional designs which rely upon an external crystal as the clock source must incur the startup delay of the crystal when resuming from Stop mode. This is a waste of time and power as no work can be performed until the crystal has stabilized.

Although the ring oscillator provides an approximately 2–4 MHz clock source for device operation, it is not as stable as an external crystal. As a result, high accuracy timing operations should be avoided while running from the ring oscillator. This includes using the timers for pulse measurement, and the use of the serial ports in asynchronous modes (1, 2, 3). Serial ports operating in mode 0 are unaffected by the stability of the clock source as a separate synchronizing clock is generated.

If the Ring Oscillator Select bit, RGSL (EXIF.1) is set, the device will resume operation immediately using the internal ring oscillator as the clock source. The device will continue to run from the ring oscillator until the crystal warm–up period of 65,536 clock cycles (measured from the external source) has completed. At this time the device will switch to the clock source active before it entered Stop mode and continue operation. This allows software execution to begin immediately upon resuming from Stop mode. The current clock source is indicated by the Ring Oscillator Mode bit, RGMD (EXIF.2).

In Stop mode, enabled interrupts become true edge triggered interrupts, compared with the sampled edge detection used during normal operation. This means that external interrupts are more sensitive to noise in Stop mode than during normal operation. Applications should be carefully designed to ensure that noise will not cause an erroneous exit from Stop mode.

SWITCHING BETWEEN CLOCK SOURCES

DS87C520 and DS87C530 incorporate the ability to run the device from the ring oscillator after the crystal warm–up period has elapsed. Immediately following a reset (including initial power–up), all devices must operate from an external crystal or oscillator. At this point, software may switch to the ring oscillator by clearing the XT/$\overline{RG}$ bit (EXIF.3). If there is no expectation that the crystal oscillator will be needed soon, the crystal oscillator can be disabled by setting the Crystal Oscillator Disable Bit, XTOFF (PMR.3). Note that switching to the ring oscillator does not automatically disable the crystal amplifier, and thus it is possible to be operating the device from the ring oscillator and have the external crystal amplifier operating at the same time. In some cases this may be desired to take advantage of the low–frequency, low–power feature of the ring oscillator but still have the capability of quickly switching back to the external crystal to perform timing or serial port operations.

Switching from the ring oscillator to the crystal oscillator is more involved due to the startup delays inherent in the external crystal. To prevent an accidental disabling of the device, the XTUP bit must be set by internal hardware (indicating an enabled, stable crystal) before setting the XT/$\overline{RG}$ bit. The procedure to switch to the crystal oscillator when running from the ring oscillator is as follows:

1. Clear the Crystal Oscillator Disable Bit, XTOFF (PMR.3) to restart the crystal oscillator and start the crystal warm–up period.
2. Wait for the Crystal Oscillator Warm Up Status bit, XTUP (STATUS.4) to be set, indicating that the external crystal warm up period is complete. This will take 65,536 external clock cycles.
3. Set the Crystal Oscillator/Ring Oscillator Select Bit, XT/$\overline{RG}$ (EXIF.3) to select the crystal as the clock source.