Analog Devices Inc./Maxim Integrated - DS89C420-MCL Datasheet

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	8051
Core Size	8-Bit
Speed	33MHz
Connectivity	EBI/EMI, SIO, UART/USART
Peripherals	Power-Fail Reset, WDT
Number of I/O	32
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	-
Oscillator Type	External
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.600", 15.24mm)
Supplier Device Package	40-PDIP
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/ds89c420-mcl

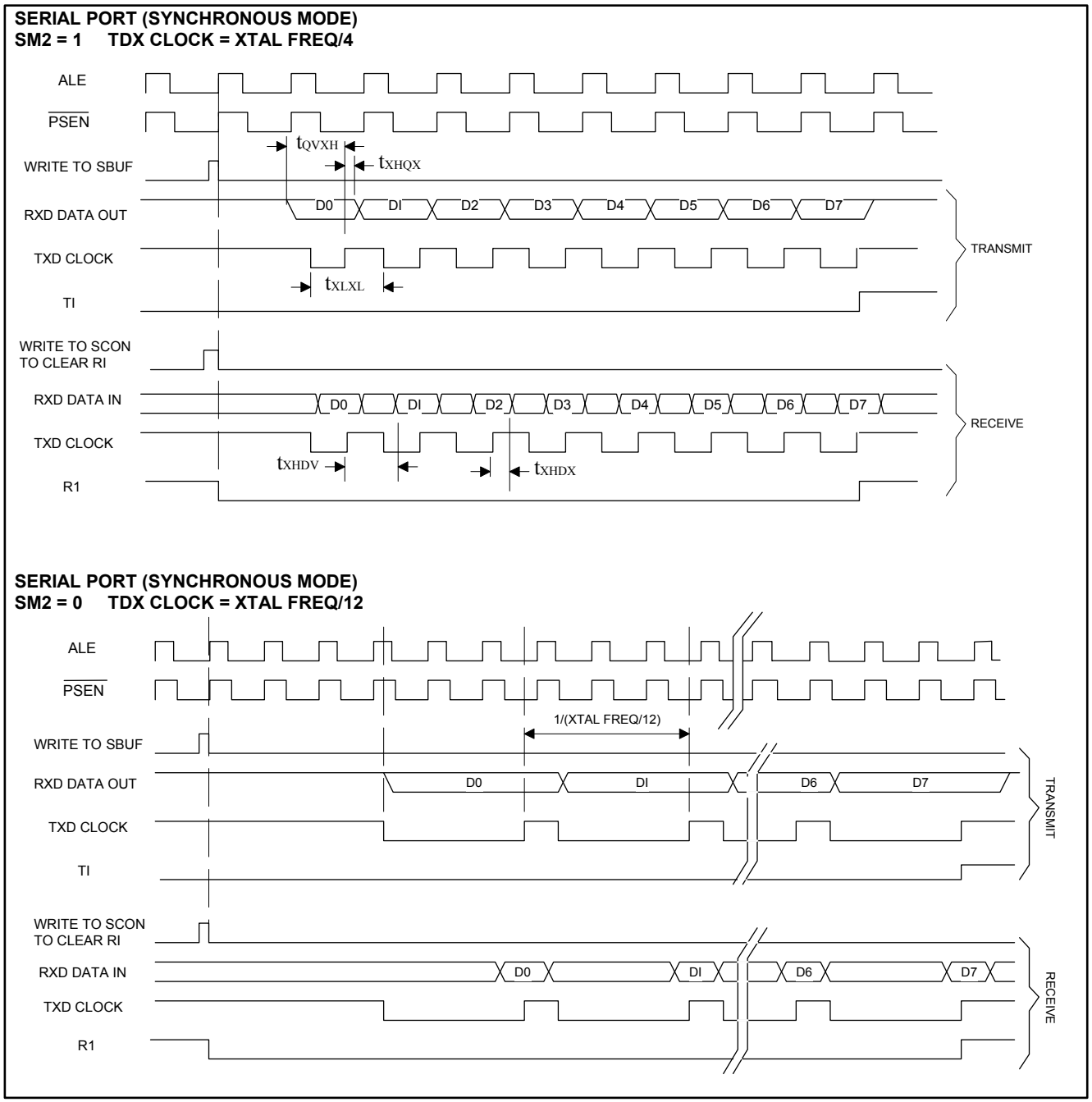
AC CHARACTERISTICS

(V_{CC} = 4.5V to 5.5V; T_A = -40°C to +85°C)* (Figure 1, Figure 2, and Figure 3)

PARAMETER		SYMBOL	1 CYCLE PAGE MODE 1		2 CYCLE PAGE MODE 1		4 CYCLE PAGE MODE 1		PAGE MODE 2		NON-PAGE MODE		UNITS
			MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
System Clock (Note 1)	External Oscillator (25MHz, 33MHz)	1 / t_{CLCL}	0	25	0	25	0	25	0	25	0	25	MHz
			0	33	0	33	0	33	0	33	0	33	
	External Crystal (25MHz, 33MHz)		1	25	1	25	1	25	1	25	1	25	
			1	33	1	33	1	33	1	33	1	33	
ALE Pulse Width (Note 2)		t_{LHLL}	$0.5t_{CLCL}$ - 2 + t_{STC3}		t_{CLCL} - 2 + t_{STC3}		$2t_{CLCL}$ - 4 + t_{STC3}		$1.5t_{CLCL}$ - 5 + t_{STC3}		$1.5t_{CLCL}$ - 5 + t_{STC3}		ns
Port 0 Instruction Address Valid to ALE Low		t_{AVLL}							t_{CLCL} - 2		$0.5t_{CLCL}$ - 2		ns
Port 2 Instruction Address Valid to ALE Low		t_{AVLL2}	$0.5t_{CLCL}$ - 4		$0.5t_{CLCL}$ - 4		$1.5t_{CLCL}$ - 5		$0.5t_{CLCL}$ - 2		t_{CLCL} - 2		ns
Port 0 Data AddressValid to ALE Low		t_{AVLL3}							t_{CLCL} - 2 + t_{STC3}		$0.5t_{CLCL}$ - 2 + t_{STC3}		ns
Program Address Hold After ALE Low		t_{LLAX}	$0.5t_{CLCL}$ - 8		$1.5t_{CLCL}$ - 8		$2.5t_{CLCL}$ - 8		$0.5t_{CLCL}$ - 8		$0.5t_{CLCL}$ - 8		ns
Address Hold After ALE Low MOVX Write		t_{LLAX2}	$0.5t_{CLCL}$ - 8 + t_{STC4}		$1.5t_{CLCL}$ - 8 + t_{STC4}		$2.5t_{CLCL}$ - 8 + t_{STC4}		$0.5t_{CLCL}$ - 8 + t_{STC4}		$0.5t_{CLCL}$ - 8 + t_{STC4}		ns
Address Hold After ALE Low MOVX Read		t_{LLAX3}	$0.5t_{CLCL}$ - 8 + t_{STC4}		$1.5t_{CLCL}$ - 8 + t_{STC4}		$2.5t_{CLCL}$ - 8 + t_{STC4}		$0.5t_{CLCL}$ - 8 + t_{STC4}		$0.5t_{CLCL}$ - 8 + t_{STC4}		ns
ALE Low to Valid Instruction In		t_{LLIV}								$2.5t_{CLCL}$ - 20		$2.5t_{CLCL}$ - 20	ns
ALE Low to $\overline{PSEN}$ Low		t_{LLPL}							$1.5t_{CLCL}$ - 6		$0.5t_{CLCL}$ - 6		ns
$\overline{PSEN}$ Pulse Width for Program Fetch		t_{PLPH}	t_{CLCL} - 5		t_{CLCL} - 5		$2t_{CLCL}$ - 5		t_{CLCL} - 5		$2t_{CLCL}$ - 5		ns

PARAMETER	SYMBOL	1 CYCLE PAGE MODE 1		2 CYCLE PAGE MODE 1		4 CYCLE PAGE MODE 1		PAGE MODE 2		NON-PAGE MODE		UNITS
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
$\overline{PSEN}$ Low to Valid Instruction In	t_{PLIV}		t_{CLCL} - 18		t_{CLCL} - 18		$2t_{CLCL}$ - 18		t_{CLCL} - 18		$2t_{CLCL}$ - 18	ns
Input Instruction Hold After $\overline{PSEN}$	t_{PXIX}	0		0		0		0		0		ns
Input Instruction Float After $\overline{PSEN}$	t_{PXIZ}								t_{CLCL} - 5		t_{CLCL} - 5	ns
Port 0 Address to Valid Instruction In	t_{AVIV0}								$1.5t_{CLCL}$ - 20		$3t_{CLCL}$ - 20	ns
Port 2 Address to Valid Instruction In	t_{AVIV2}		t_{CLCL} - 18		$1.5t_{CLCL}$ - 18		$2.5t_{CLCL}$ - 18		$3t_{CLCL}$ - 20		$3.5t_{CLCL}$ - 20	ns
$\overline{PSEN}$ Low to Port 0 Address Float	t_{PLAZ}								0		0	ns
$\overline{RD}$ Pulse Width (P3.7) (Note 2)	t_{RLRH}	t_{CLCL} - 5 + t_{STC1}		t_{CLCL} - 5 + t_{STC1}		$2t_{CLCL}$ - 5 + t_{STC1}		$2t_{CLCL}$ - 5 + t_{STC1}		$2t_{CLCL}$ - 5 + t_{STC1}		ns
$\overline{WR}$ Pulse Width (P3.6) (Note 2)	t_{WLWH}	t_{CLCL} - 5 + t_{STC1}		t_{CLCL} - 5 + t_{STC1}		$2t_{CLCL}$ - 5 + t_{STC1}		$2t_{CLCL}$ - 5 + t_{STC1}		$2t_{CLCL}$ - 5 + t_{STC1}		ns
$\overline{RD}$ (P3.7) Low to Valid Data In (Note 2)	t_{RLDV}		t_{CLCL} - 15 + t_{STC1}		t_{CLCL} - 15 + t_{STC1}		$2t_{CLCL}$ - 15 + t_{STC1}		$2t_{CLCL}$ - 15 + t_{STC1}		$2t_{CLCL}$ - 15 + t_{STC1}	ns
Data Hold After $\overline{RD}$ (P3.7)	t_{RHDX}	0		0		0		0		0		ns
Data Float After $\overline{RD}$ (P3.7)	t_{RHDZ}								t_{CLCL} - 5		t_{CLCL} - 5	ns
MOVX ALE Low to Input Data Valid (Note 2)	t_{LLDV}								$2.5t_{CLCL}$ - 20 + t_{STC1}		$2.5t_{CLCL}$ - 20 + t_{STC1}	ns
Port 0 Address to Valid Data In (Note 2)	t_{AVDV0}								$3t_{CLCL}$ - 20 + t_{STC1}		$3t_{CLCL}$ - 20 + t_{STC1}	ns

Figure 4. Serial Port Timing



PIN DESCRIPTION

PIN			NAME	FUNCTION		
DIP	PLCC	TQFP				
40	12, 44	6, 38	V_{CC}	V_{CC} - +5V		
20	1, 22, 23, 34	16, 17, 28, 39	GND	Logic Ground		
9	10	4	RST	External Reset. The RST input pin is bidirectional and contains a Schmitt trigger to recognize external active-high reset inputs. The pin also employs an internal pulldown resistor to allow for a combination of wire-ORed external reset sources. An RC is not required for power-up, since the device provides this function internally.		
19	21	15	XTAL1	XTAL1, XTAL2. The crystal oscillator pins XTAL1 and XTAL2 provide support for fundamental mode parallel resonant, AT cut crystals. XTAL1 also acts as an input if there is an external clock source in place of a crystal. XTAL2 serves as the output of the crystal amplifier.		
18	20	14	XTAL2			
29	32	26	$\overline{PSEN}$	Program Store Enable. This signal is commonly connected to optional external program memory as a chip enable. $\overline{PSEN}$ provides an active-low pulse and is driven high when external program memory is not being accessed. In 1-cycle page mode 1, $\overline{PSEN}$ remains low for consecutive page hits.		
30	33	27	ALE/$\overline{PROG}$	Address Latch Enable. Functions as a clock to latch the external address LSB from the multiplexed address/data bus on Port 0. This signal is commonly connected to the latch enable of an external 373 family transparent latch. In default mode, ALE has a pulse width of 1.5 XTAL1 cycles and a period of four XTAL1 cycles. In page mode, the ALE pulse width is altered according to the page mode selection. In traditional 8051 mode, ALE is high when using the EMI reduction mode and during a reset condition. ALE can be enabled by writing ALEON = 1 (PMR.2). Note that ALE operates independently of ALEON during external memory accesses. As an alternate mode, this pin ($\overline{PROG}$) is used to execute the parallel program function.		
39	43	37	P0.0 (AD0)	Port 0 (AD0–7), I/O. Port 0 is an open-drain 8-bit, bidirectional I/O port. As an alternate function, Port 0 can function as the multiplexed address/data bus to access off-chip memory. During the time when ALE is high, the LSB of a memory address is presented. When ALE falls to a logic 0, the port transitions to a bidirectional data bus. This bus is used to read external program memory and read/write external RAM or peripherals. When used as a memory bus, the port provides weak pullups for logic 1 outputs. The reset condition of Port 0 is three-state. Pullup resistors are required when using Port 0 as an I/O port.		
38	42	36	P0.1 (AD1)			
37	41	35	P0.2 (AD2)			
36	40	34	P0.3 (AD3)			
35	39	33	P0.4 (AD4)			
34	38	32	P0.5 (AD5)			
33	37	31	P0.6 (AD6)			
32	36	30	P0.7 (AD7)			
1–8	2–9	40–44, 1–3	P1.0–P1.7	Port 1, I/O. Port 1 functions as both an 8-bit, bidirectional I/O port and an alternate functional interface for timer 2 I/O, new external interrupts, and new serial port 1. The reset condition of port 1 is with all bits at logic 1. In this state, a weak pullup holds the port high. This condition also serves as an input state, since any external circuit that writes to the port overcomes the weak pullup. When software writes a 0 to any port pin, the DS89C420 activates a strong pulldown that remains on until either a 1 is written or a reset occurs. Writing a 1 after the port has been at 0 causes a strong transition driver to turn on, followed by a weaker sustaining pullup. Once the momentary strong driver turns off, the port again becomes the output high (and input) state. The alternate functions of Port 1 are outlined below.		
				PORT	**ALTERNATE**	**FUNCTION**
1	2	40		P1.0	T2	External I/O for Timer/Counter 2
2	3	41		P1.1	T2EX	Timer 2 Capture/Reload Trigger
3	4	42		P1.2	RXD1	Serial Port 1 Receive
4	5	43		P1.3	TXD1	Serial Port 1 Transmit
5	6	44		P1.4	INT2	External Interrupt 2 (Positive Edge Detect)
6	7	1		P1.5	$\overline{INT3}$	External Interrupt 3 (Negative Edge Detect)
7	8	2		P1.6	INT4	External Interrupt 4 (Positive Edge Detect)
8	9	3		P1.7	$\overline{INT5}$	External Interrupt 5 (Negative Edge Detect)

COMPATIBILITY

The DS89C420 is a fully static CMOS 8051-compatible microcontroller similar to the DS87C520 in functional features, but with much higher performance. In most cases the DS89C420 can drop into an existing socket for the 8xC51 family to improve the operation significantly. While remaining familiar to 8051 family users, it has many new features. The DS89C420 runs the standard 8051 family instruction set and is pin compatible with DIP, PLCC, and TQFP packages. In general, software written for existing 8051-based systems works without DS89C420 modification, with the exception of critical timing routines, since the DS89C420 performs its instructions much faster than the original for any given crystal selection.

The DS89C420 provides three 16-bit timer/counters, two full-duplex serial ports, and 256 bytes of direct RAM plus 1kB of extra MOVX RAM. I/O ports can operate as in standard 8051 products. Timers default to a 12 clock-per-cycle operation to keep their timing compatible with original 8051 family systems. However, timers are individually programmable to run at the new 1 clock-per-cycle if desired. The DS89C420 provides several new hardware features implemented by new SFRs.

PERFORMANCE OVERVIEW

The DS89C420 features a completely redesigned high-speed 8051-compatible core and allows operation at a higher clock frequency, but the updated core does not have the dummy memory cycles that are present in a standard 8051. A conventional 8051 generates machine cycles using the clock frequency divided by 12. In the DS89C420, the same machine cycle takes 1 clock. Thus, the fastest instructions execute 12 times faster for the same crystal frequency (and actually 24 times faster for the INC data pointer instruction). It should be noted that this speed improvement reduces when using external memory access modes that require more than 1 clock per cycle.

Improvement of individual programs depends on the actual instructions used. Speed-sensitive applications make the most use of instructions that are 12 times faster. However, the sheer number of 12-to-1 improved op codes makes dramatic speed improvements likely for any code. These architecture improvements produce instruction cycle times as low as 30ns (33MIPs). The dual data pointer feature also allows the user to eliminate wasted instructions when moving blocks of memory. The new page modes allow for increased efficiency in external memory accesses.

INSTRUCTION SET SUMMARY

All instructions perform the same functions as their 8051 counterparts. Their effect on bits, flags, and other status functions is also identical. However, the timing of each instruction is different in both absolute and relative number of clocks.

For absolute timing of real-time events, the timing of software loops can be calculated using information in the “Instruction Set” table of the *Ultra-High-Speed Flash Microcontroller User’s Guide*. However, counter/timers default to run at the older 12 clocks per increment. In this way, timer-based events occur at the standard intervals with software executing at higher speed. Timers optionally can run at lower numbers of clocks per increment to take advantage of faster processor operation.

The relative time of some instructions might be different in the new architecture than it was previously. For example, in the original architecture, the “MOVX A, @DPTR” instruction and the “MOV direct, direct” instruction used two machine cycles or 24 oscillator cycles. Therefore, they required the same amount of time. In the DS89C420, the MOVX instruction takes as little as two machine cycles or two oscillator cycles but the “MOV direct, direct” uses three machine cycles or three oscillator cycles. While both are faster than their original counterparts, they now have different execution times. This is because the DS89C420 usually uses one machine cycle for each instruction byte and requires one cycle for execution. The user concerned with precise program timing should examine the timing of each instruction to become familiar with the changes.

SPECIAL FUNCTION REGISTERS (SFRS)

All peripherals and operations that are not explicit instructions in the DS89C420 are controlled through SFRs. The most common features basic to the architecture are mapped to the SFRs. These include the CPU registers (ACC, B, and PSW), data pointers (DPTRs), stack pointer, I/O ports, timer/counters, and serial ports. In many cases, an SFR controls an individual function or reports the function’s status. The SFRs reside in register locations 80h–FFh and are only accessible by direct addressing. SFRs whose addresses end in 0h or 8h are bit-addressable.

All standard SFR locations from the 8051 are duplicated in the DS89C420 and several SFRs have been added for the unique features of the DS89C420. Most of these features are controlled by bits in SFRs located in unused locations in the 8051 SFR map. This allows for increased functionality while maintaining complete instruction set compatibility. Table 1 summarizes the SFRs and their locations. Table 2 specifies the default reset condition for all SFR bits.

DATA POINTERS

The data pointers (DPTR and DPTR1) are used to assign a memory address for the MOVX instructions. This address can point to a MOVX RAM location (on-chip or off-chip), or a memory-mapped peripheral. Two pointers are useful when moving data from one memory area to another, or when using a memory-mapped peripheral for both source and destination addresses. The user selects the active pointer through a dedicated SFR bit (Sel = DPS.0), or activates an automatic toggling feature for altering the pointer selection (TSL = DPS.5). An additional feature, if selected, provides automatic incrementing or decrementing of the current DPTR.

STACK POINTER

The stack pointer denotes the register location at the top of the stack, which is the last used value. The user can place the stack anywhere in the scratchpad RAM by setting the stack pointer to the desired location, although the lower bytes are normally used for working registers.

I/O PORTS

The DS89C420 offers four 8-bit I/O ports. Each I/O port is represented by an SFR location, and can be written or read. The I/O port has a latch that contains the value written by software.

COUNTER/TIMERS

Three 16-bit timer/counters are available in the DS89C420. Each timer is contained in two SFR locations that can be read or written by software. The timers are controlled by other SFRs described in the “SFR Bit Description” section of the *Ultra-High-Speed Flash Microcontroller User’s Guide*.

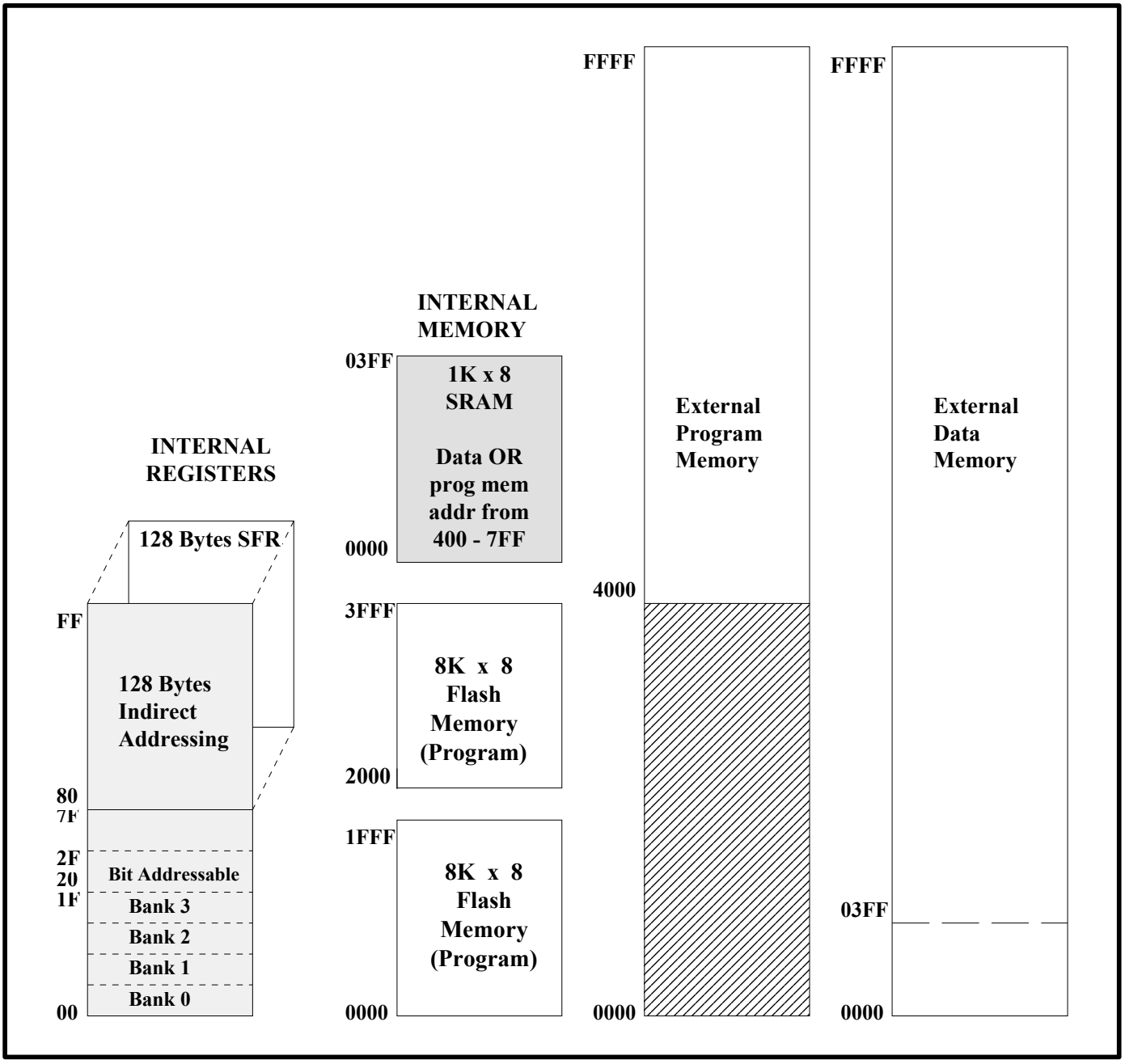
SERIAL PORTS

The DS89C420 provides two UARTs that are controlled and accessed by SFRs. Each UART has an address that is used to read and write the UART. The same address is used for read and write operations, which are distinguished by the instruction. Its own SFR control register controls each UART.

Table 1. Special Function Registers

REGISTER	ADDR	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
P0	80h	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
SP	81h	—	—	—	—	—	—	—	—
DPL	82h	—	—	—	—	—	—	—	—
DPH	83h	—	—	—	—	—	—	—	—
DPL1	84h	—	—	—	—	—	—	—	—
DPH1	85h	—	—	—	—	—	—	—	—
DPS	86h	ID1	ID0	TSL	AID	—	—	—	SEL
PCON	87h	SMOD_0	SMOD0	OFDF	OFDE	GF1	GF0	STOP	IDLE
TCON	88h	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
TMOD	89h	GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0
TL0	8Ah	—	—	—	—	—	—	—	—
TL1	8Bh	—	—	—	—	—	—	—	—
TH0	8Ch	—	—	—	—	—	—	—	—
TH1	8Dh	—	—	—	—	—	—	—	—
CKCON	8Eh	WD1	WD0	T2M	T1M	T0M	MD2	MD1	MD0
P1	90h	P1.7/$\overline{INT5}$	P1.6/INT4	P1.5/$\overline{INT3}$	P1.4/INT2	P1.3/TXD1	P1.2/RXD1	P1.1/T2EX	P1.0/T2
EXIF	91h	IE5	IE4	IE3	IE2	CKRY	RGMD	RGSL	BGS
CKMOD	96h	—	—	T2MH	T1MH	T0MH	—	—	—
SCON0	98h	SM0/FE_0	SM1_0	SM2_0	REN_0	TB8_0	RB8_0	TI_0	RI_0
SBUF0	99h	—	—	—	—	—	—	—	—
ACON	9Dh	PAGEE	PAGES1	PAGES0	—	—	—	—	—
P2	A0h	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0
IE	A8h	EA	ES1	ET2	ES0	ET1	EX1	ET0	EX0
SADDR0	A9h	—	—	—	—	—	—	—	—
SADDR1	AAh	—	—	—	—	—	—	—	—
P3	B0h	P3.7/$\overline{RD}$	P3.6/$\overline{WR}$	P3.5/T1	P3.4/T0	P3.3/$\overline{INT1}$	P3.2/$\overline{INT0}$	P3.1/TXD0	P3.0/RXD0
IP1	B1h	—	MPS1	MPT2	MPS0	MPT1	MPX1	MPT0	MPX0
IP0	B8h	—	LPS1	LPT2	LPS0	LPT1	LPX1	LPT0	LPX0
SADEN0	B9h	—	—	—	—	—	—	—	—
SADEN1	BAh	—	—	—	—	—	—	—	—
SCON1	C0h	SM0/FE_1	SM1_1	SM2_1	REN_1	TB8_1	RB8_1	TI_1	RI_1
SBUF1	C1h	—	—	—	—	—	—	—	—
ROMSIZE	C2h	—	—	—	—	PRAME	RMS2	RMS1	RMS0
PMR	C4h	CD1	CD0	SWB	CTM	4X/$\overline{2X}$	ALEON	DME1	DME0
STATUS	C5h	PIS2	PIS1	PIS0	—	SPTA1	SPRA1	SPTA0	SPRA0
TA	C7h	—	—	—	—	—	—	—	—
T2CON	C8h	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$
T2MOD	C9h	—	—	—	—	—	—	T2OE	DCEN
RCAP2L	CAh	—	—	—	—	—	—	—	—
RCAP2H	CBh	—	—	—	—	—	—	—	—
TL2	CCh	—	—	—	—	—	—	—	—
TH2	CDh	—	—	—	—	—	—	—	—
PSW	D0h	CY	AC	F0	RS1	RS0	OV	F1	P
FCNTL	D5h	$\overline{FBUSY}$	FERR	—	—	FC3	FC2	FC1	FC0
FDATA	D6h	—	—	—	—	—	—	—	—
WDCON	D8h	SMOD_1	POR	EPFI	PFI	WDIF	WTRF	EWT	RWT
ACC	E0h	—	—	—	—	—	—	—	—
EIE	E8h	—	—	—	EWDI	EX5	EX4	EX3	EX2
B	F0h	—	—	—	—	—	—	—	—
EIP1	F1h	—	—	—	MPWDI	MPX5	MPX4	MPX3	MPX2
EIP0	F8h	—	—	—	LPWDI	LPX5	LPX4	LPX3	LPX2

Figure 6. Memory Map



The reset default condition is a maximum on-chip program-memory address of 16kB. When accessing external program memory, the first 16kB would be inaccessible. To select a smaller effective program memory size, software must alter bits RMS2–RMS0. Altering these bits requires a timed access procedure as explained later.

Care should be taken so that changing the ROMSIZE register does not corrupt program execution. For example, assume that a DS89C420 is executing instructions from internal program memory near the 12kB boundary (~3000h) and that the ROMSIZE register is currently configured for a 16kB internal program space. If software reconfigures the ROMSIZE register to 4kB (0000h–0FFFh) in the current state, the device immediately jumps to external program execution because program code from 4kB to 16kB (1000h–3FFFh) is no longer located on-chip. This could result in code misalignment and execution of an invalid instruction. The recommended method is to modify the ROMSIZE register from a location in memory that is internal (or external) both before and after the operation. In the above example, the instruction that modifies the ROMSIZE register should be located below the

4kB (1000h) boundary or above the 16kB (3FFFh) boundary so that it is unaffected by the memory modification. The same precaution should be applied if the internal program memory size is modified while executing from external program memory.

For non-page mode operations, off-chip memory is accessed using the multiplexed address/data bus on P0 and the MSB address on P2. While serving as a memory bus, these pins are not I/O ports. This convention follows the standard 8051 method of expanding on-chip memory. Off-chip program memory access also occurs if the $\overline{EA}$ pin is logic 0. $\overline{EA}$ overrides all bit settings. The $\overline{PSEN}$ signal goes active (low) to serve as a chip enable or output enable when port 0 and port 2 fetch from external program memory.

The $\overline{RD}$ and $\overline{WR}$ signals are used to control the external data memory device. Data memory is accessed by MOVX instructions. The MOVX@Ri instruction uses the value in the designated working register to provide the LSB of the address, while port 2 supplies the address MSB. The MOVX@DPTR instruction uses one of the two data pointers to move data over the entire 64kB external data memory space. Software selects the data pointer to be used by writing to the SEL bit (DPS.0).

The DS89C420 also provides a user option for high-speed external memory access by reconfiguring the external memory interface into page mode operation.

Note: When using the original 8051 expanded bus structure, the throughput is reduced by 75% compared with that of internal operations. This is due to the CPU being stalled for three out of four clocks waiting for the data fetch, which takes four clocks. Page Mode 1 is the only external addressing mode where the CPU does not require stalls for external memory access, but page misses result in reduced external access performance.

ON-CHIP PROGRAM MEMORY

The processor can fetch the full on-chip program memory range automatically. The reset routines and all interrupt vectors are located in the lower 128 bytes of the on-chip program memory area.

On-chip program memory is logically divided into two 8kB flash memory banks and is designed to be programmed with the standard 5V V_{CC} supply by using a built-in program memory loader. It can also be programmed in standard flash or EPROM programmers. The DS89C420 incorporates a memory management unit (MMU) and other hardware to support any of the two programming methods. The MMU controls program and data memory access, and provides sequencing and timing controls for programming the on-chip program memory. There is also a separate security flash block that is used to support a standard three-level lock, a 64-byte encryption array, and other flash options.

SECURITY FEATURES

The DS89C420 incorporates a 64-byte encryption array, allowing the user to verify program codes while viewing the data in encrypted form. The encryption array is implemented in a security flash memory block that has the same electrical and timing characteristics as the on-chip program memory. Once the encryption array is programmed to non-FFh, the data presented in the verify mode is encrypted. Each byte of data is XNORed with a byte in the encryption array during verification.

A three-level lock restricts viewing of the internal program and data memory contents. By programming the three lock bits, the user can select a level of security as specified in Table 3. Once a security level is selected and programmed, the setting of the lock bits remains. Only a mass erase can erase these bits to allow reprogramming the security level to a less restricted protection.

Table 4. Parallel Programming Instruction Set

INSTRUCTION	P2.5:0, P1.7:0	P0.7:0	$\overline{PROG}$	P2.6	P2.7	P3.6	P3.7	OPERATION
Mass Erase	Don't care	Don't care	PL[1]	H	L	L	L	Mass erase the 16k x 8 program memory, the security block and the bank select. The contents of every memory location is returned to FFh.
Write Program Memory	ADDR	DIN	PL[3]	L	H	H	H	Program the 16k program memory.
Read Program Memory	ADDR	DOUT	H[4]	L	L	H	H	Verify the 16k program memory.
Write Encryption Array	ADDR	DIN	PL[3]	L	H	L	H	Program the 64 byte encryption array.
Write LB1	Don't care	Don't care	PL[3]	H	H	H	H	Program LB1 to logic 0.
Write LB2	Don't care	Don't care	PL[3]	H	H	L	L	Program LB2 and LB1 to 00b.
Write LB3	Don't care	Don't care	PL[3]	H	L	H	L	Program LB3, LB2, and LB1 to 000b.
Read Lock Bits	Don't care	DOUT	H[4]	L	L	L	H	Verify the lock bits. The lock bits are at address 40h and the three LSBs of the DOUT are the logic value of the lock bits LB3, LB2, and LB1, respectively.
Write Option Control Register	Don't care	DIN	PL[3]	L	H	L	L	Program the option control register. Bit 3 of the DIN represents the watchdog POR default setting.
Erase Option Control Register	Don't care	Don't care	PL[2]	H	L	L	H	Erase the option control register. This operation disables the watch-dog reset function on power-up.
Read Address 30, 31, 60, FC	ADDR	DOUT	H[4]	L	L	L	L	30h = Manufacturer ID 31h = Device ID 60h = Device extension FCh = Verify the option control register. Bit 3 of the DOUT is the logic value of the watchdog POR.

[1] Mass erase requires an active-low $\overline{PROG}$ pulse width of 828ms.
[2] Erase option control register requires an active-low $\overline{PROG}$ pulse width of 828ms.
[3] Byte program requires an active-low $\overline{PROG}$ pulse width of 100µs max.
[4] $\overline{PROG}$ is weakly pulled to a high internally.

Note 1: P3.2 is pulled low during programming to indicate Busy. P3.2 is pulled high again when programming is completed to indicate Ready.
Note 2: P3.0 is pulled high during programming to indicate an error.

DATA POINTER INCREMENT/DECREMENT AND OPTIONS

The DS89C420 incorporates a hardware feature to assist applications that require data pointer increment/decrement. Data pointer increment/decrement bits ID0 and ID1 (DPS.6 and DPS.7) define how the INC DPTR instruction functions in relation to the active DPTR (selected by the SEL bit). Setting ID0 = 1 and SEL = 0 enables the decrement operation for DPTR, and execution of the INC DPTR instruction decrements the DPTR contents by 1. Similarly, setting ID1 = 1 and SEL = 1 enables the decrement operation for DPTR1, and execution of the INC DPTR instruction decrements the DPTR1 contents by 1. With this feature, the user can configure the data pointers to operate in four ways for the INC DPTR instruction:

ID1	ID0	SEL = 0	SEL = 1
0	0	Increment DPTR	Increment DPTR1
0	1	Decrement DPTR	Increment DPTR1
1	0	Increment DPTR	Decrement DPTR1
1	1	Decrement DPTR	Decrement DPTR1

The SEL (DPS.0) bit always selects the active data pointer. The DS89C420 offers a programmable option that allows any instructions related to data pointer to toggle the SEL bit automatically. This option is enabled by setting the toggle-select-enable bit (TSL-DPS.5) to logic 1. Once enabled, the SEL bit is automatically toggled after the execution of one of the following five DPTR-related instructions:

```
INC DPTR
MOV DPTR #data16
MOVC A, @A+DPTR
MOVX A, @DPTR
MOVX @DPTR, A
```

The DS89C420 also offers a programmable option that automatically increases (or decreases) the contents of the selected data pointer by 1 after the execution of a DPTR-related instruction. The actual function (increment or decrement) is dependent upon the setting of the ID1 and ID0 bits. This option is enabled by setting the automatic increment/decrement enable (AID-DPS.4) to a logic 1 and is affected by one of the following three instructions:

```
MOVC A, @A+DPTR
MOVX A, @DPTR
MOVX @DPTR, A
```

EXTERNAL MEMORY

The DS89C420 executes external memory cycles for code fetches and read/writes of external program and data memory. A non-page external memory cycle is four times slower than the internal memory cycles (i.e., an external memory cycle contains four system clocks). For this reason, although a DS89C420 can be substituted for a ROM-less 8051 device (DS80C310, C320, etc.), there is no increase in execution speed.

However, a page mode external memory cycle can be completed in 1, 2, or 4 system clocks for a page hit and 2, 4, or 8 system clocks for a page miss, depending on user selection. The DS89C420 also supports a second page mode operation with a different external bus structure that provides for fast external code fetches but uses 4 system clock cycles for data memory access.

EXTERNAL PROGRAM MEMORY INTERFACE (NON-PAGE MODE)

Figure 8 shows the timing relationship for internal and external code fetches when CD1 and CD0 are set to 10b, assuming the microcontroller is in non-page mode for external fetches. Note that an external program fetch takes 4 system clocks, and an internal program fetch requires only 1 system clock.

As illustrated in Figure 8, ALE is deasserted when executing an internal memory fetch. The DS89C420 provides a programmable user option to turn on ALE during internal program memory operation. ALE is automatically enabled for code fetch externally, independent of the setting of this option.

$\overline{\text{PSEN}}$ is only asserted for external code fetches, and is inactive during internal execution.

EXTERNAL DATA MEMORY INTERFACE IN NON-PAGE MODE OPERATION

Just like the program memory cycle, the external data memory cycle is four times slower than the internal data memory cycle in non-page mode. A basic internal memory cycle contains one system clock and a basic external memory cycle contains four system clocks for non-page mode operation.

The DS89C420 allows software to adjust the speed of external data memory access by stretching the memory bus cycle. CKCON (8Eh) provides an application-selectable stretch value for this purpose. Software can change the stretch value dynamically by changing the setting of CKCON.2–CKCON.0. Table 5 shows the data memory cycle stretch values and their effects on the external MOVX-memory bus cycle and the control signal pulse width in terms of the number of oscillator clocks. A stretch machine cycle always contains four system clocks.

Figure 8. External Program Memory Access (Non-Page Mode and CD1:CD0 = 10)

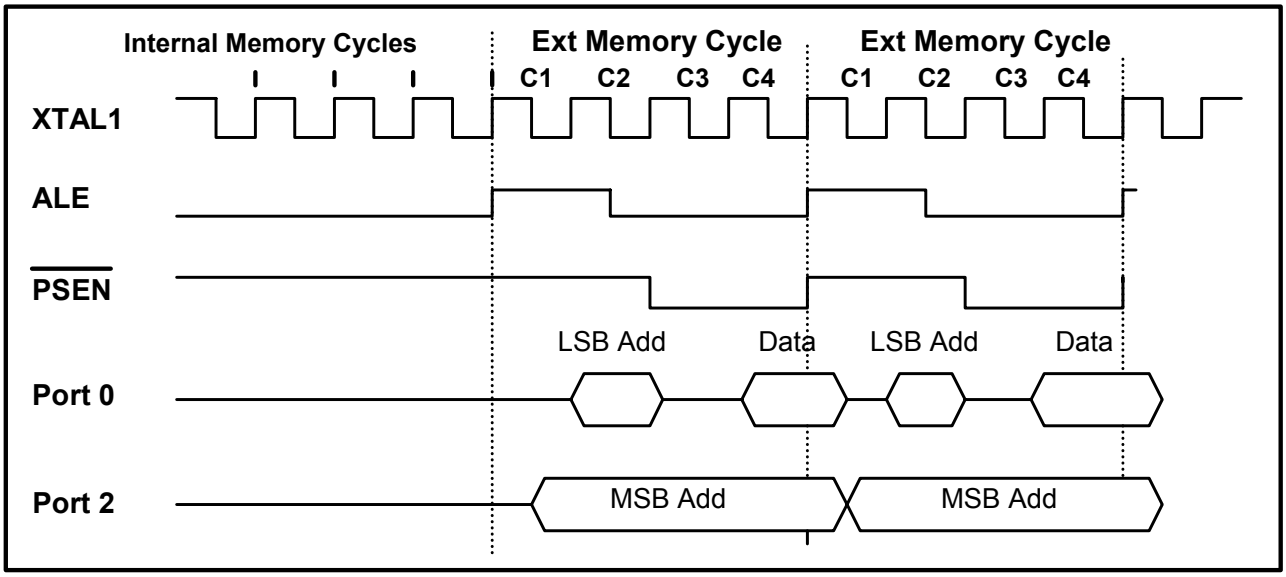


Table 5. Data Memory Cycle Stretch Values

MD2:MD0	STRETCH CYCLES	$\overline{RD}/\overline{WR}$ PULSE WIDTH (IN NUMBER OF OSCILLATOR CLOCKS)			
		4X/$\overline{2X}$, CD1, CD0 = 100	4X/$\overline{2X}$, CD1, CD0 = 000	4X/$\overline{2X}$, CD1, CD0 = X10	4X/$\overline{2X}$, CD1, CD0 = X11
000	0	0.5	1	2	2048
001	1	1	2	4	4096
010	2	2	4	8	8192
011	3	3	6	12	12288
100	7	4	8	16	16384
101	8	5	10	20	20480
110	9	6	12	24	24576
111	10	7	14	28	28672

As shown in Table 5, the stretch feature supports eight stretched external data-memory access cycles that can be categorized into three timing groups. When the stretch value is cleared to 000b, there is no stretch on external data memory access and a MOVX instruction is completed in two basic memory cycles. When the stretch value is set to 1, 2, or 3, the external data-memory access is extended by 1, 2, or 3 stretch machine cycles, respectively. Note that the first stretch value does not result in adding four system clocks to the $\overline{RD}/\overline{WR}$ control signals. This is because the first stretch uses one system clock to create additional setup time and one system clock to create additional address hold time. When using very slow RAM and peripherals, a larger stretch value (4–7) can be selected. In this stretch category, one stretch machine cycle (4 system clocks) is used to stretch the ALE pulse width, one stretch machine cycle is used to create additional setup, one stretch machine cycle is used to create additional hold time, and one stretch machine cycle is added to the $\overline{RD}$ or $\overline{WR}$ strobes.

Figure 9 and Figure 10 illustrate the timing relationship for external data-memory access in full speed (stretch value = 0), in the default stretch setting (stretch value = 1), and slow data-memory accessing (stretch value = 4) when the system clock is in divide by one mode (CD1:CD0 = 10b).

Figure 9. Non-Page Mode, External Data-Memory Access (Stretch = 0, CD1:CD2 = 10)

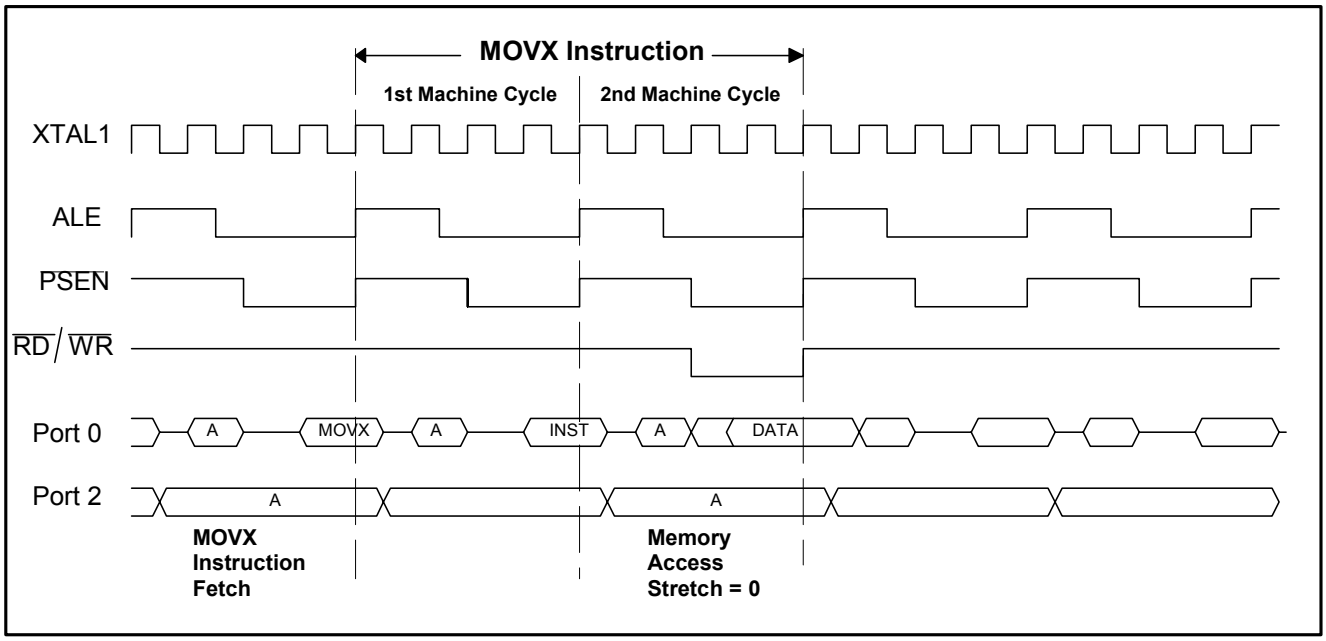


Figure 10. Non-Page Mode, External Data-Memory Access (Stretch = 1, CD1:CD2 = 10)

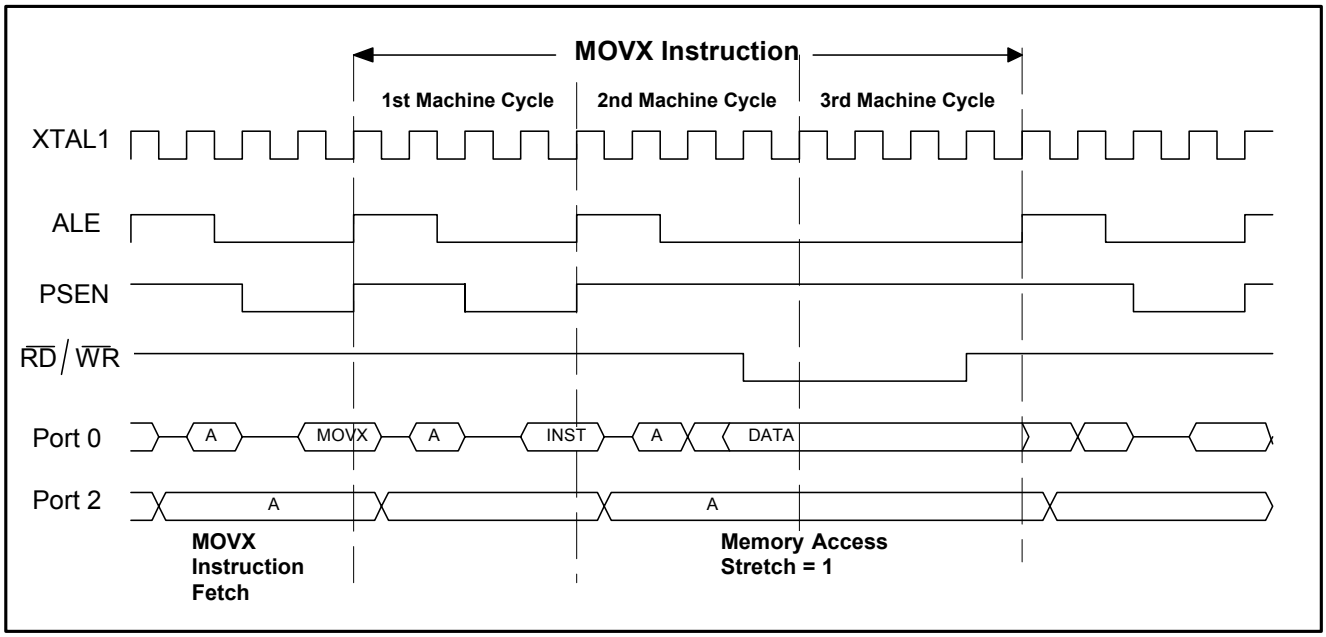


Figure 11. Page Mode 1, External Memory Cycle (CD1:CD0 = 10)

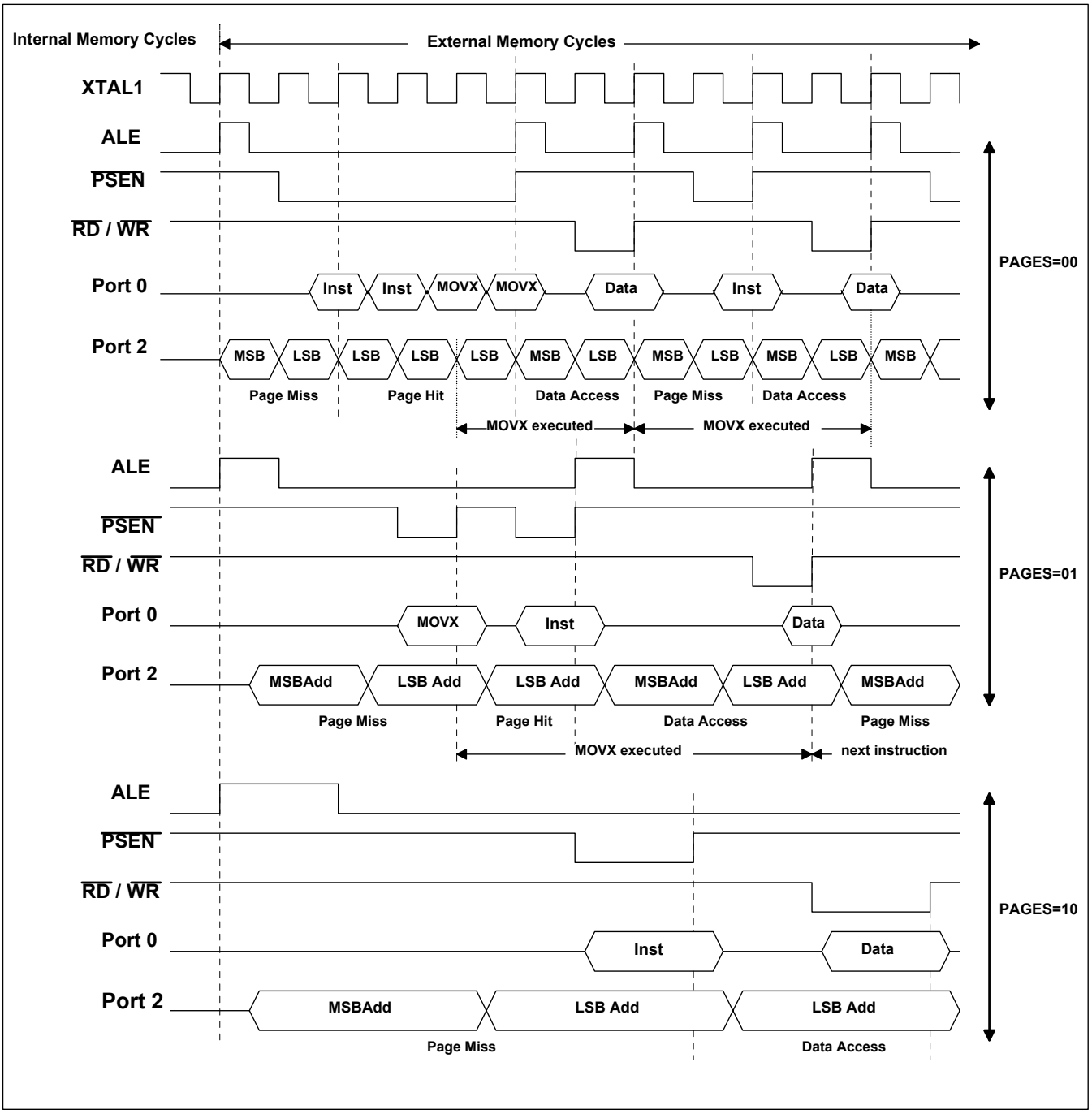


Figure 13. Page Mode 1, External Data Memory Access (Pages = 01, Stretch = 1, CD = 10)

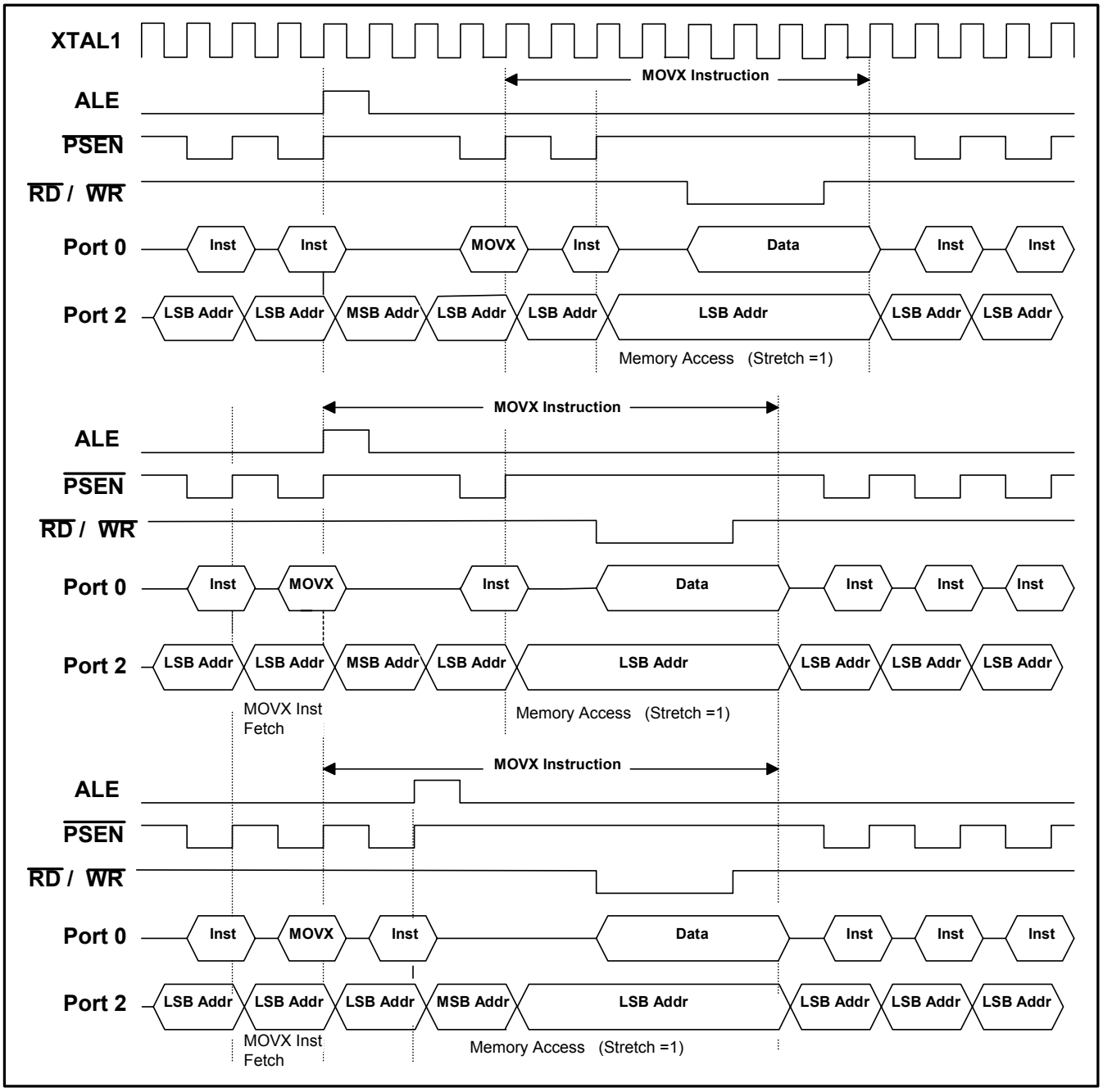


Figure 13 illustrates the external data-memory stretch cycle timing relationship when PAGEE = 1 and PAGES1:PAGES0 = 01. The stretch cycle shown is for a stretch value of 1 and is coincident with a page miss. Note that the first stretch value does not result in adding four system clocks to the $\overline{RD}/\overline{WR}$ control signals. This is because the first stretch uses one system clock to create additional setup and one system clock to create additional hold time.

timer/counters without auto-reload. Each timer can also be used as a counter of external pulses on the corresponding T0/T1 pin for 1-to-0 transitions. The timer mode (TMOD) register controls the operation mode. Each timer consists of a 16-bit register in 2 bytes, which can be found in the SFR map as TL0, TH0, TL1, and TH1. The timer control (TCON) register enables Timers 0 and 1.

Table 12. Timer Functions

FUNCTIONS	TIMER 0	TIMER 1	TIMER 2
Timer/Counter	13/16/8*/2 x 8 bit	13/16/8* bit	16 bit
Timer with Capture	No	No	Yes
External Control-Pulse Counter	Yes	Yes	No
Up/Down Auto-Reload Timer/Counter	No	No	Yes
Baud Rate Generator	No	Yes	Yes
Timer-Output Clock Generator	No	No	Yes

**8-bit timer/counter includes auto-reload feature; 2- x 8-bit mode does not.*

Timer 2 is a true 16-bit timer/counter that, with a 16-bit capture (RCAP2L and RCAP2H) register, is able to provide some unique functions like up/down auto-reload timer/counter and timer-output clock generation. Timer 2 (registers TL2 and TH2) is enabled by the T2CON register, and its mode of operation is selected by the T2MOD register.

Each timer has a selectable time base (Table 14). Following a reset, the timers default to divide-by-12 to maintain drop-in compatible with the 8051. If Timer 2 is used as a baud rate generator or clock output, its time base is fixed at divide by 2, regardless of the setting of its timer mode bits.

For details of operation, refer to “Programmable Timers” in the *Ultra-High-Speed Flash Microcontroller User’s Guide*.

TIMED ACCESS

The timed access function provides control verification to system functions. The timed access function prevents an errant CPU from making accidental changes to certain SFR bits that are considered vital to proper system operation. This is achieved by using software control when accessing the following SFR control bits:

WDCON.0	RWT	Reset Watchdog Timer
WDCON.1	EWT	Watchdog Reset Enable
WDCON.3	WDIF	Watchdog Interrupt Flag
WDCON.6	POR	Power-On Reset Flag
EXIF.0	BGS	Bandgap Select
ACON.5	PAGES0	Page Mode Select Bit 0
ACON.6	PAGES1	Page Mode Select Bit 1
ACON.7	PAGEE	Page Mode Enable
ROMSIZE.0	RMS0	Program Memory Size Select Bit 0
ROMSIZE.1	RMS1	Program Memory Size Select Bit 1
ROMSIZE.2	RMS2	Program Memory Size Select Bit 2
ROMSIZE.3	PRAME	Program RAM Enable
FCNTL.0	FC0	Flash Command Bit 0
FCNTL.1	FC1	Flash Command Bit 1
FCNTL.2	FC2	Flash Command Bit 2
FCNTL.3	FC3	Flash Command Bit 3

OSCILLATOR-FAIL DETECT

The DS89C420 incorporates an oscillator fail-detect circuit that, when enabled, causes a reset if the crystal oscillator frequency falls below 20kHz and holds the chip in reset with the ring oscillator operating. Setting the OFDE (PCON.4) bit to logic 1 enables the circuit. The OFDE bit is only cleared from logic 1 to logic 0 by a power-fail reset or by software. A reset caused by an oscillator failure also sets the OFDF (PCON.5) to logic 1. This flag is cleared by software or power-on reset. Note that this circuit does not force a reset when the oscillator is stopped by the software-enabled stop mode.

POWER MANAGEMENT MODE

Power management mode offers a software-controllable power-saving scheme by providing a reduced instruction cycle speed, which allows the DS89C420 to continue to operate while using an internally divided version of the clock source to save power. Power management mode is invoked by software setting the clock-divide control bits CD1 and CD0 (PMR.7-6) bits to 11b, which sets an operating rate of 1024 oscillator cycles for 1 machine cycle. On all forms of reset, the clock-divide control bits default to 10b, which selects 1 oscillator cycle per machine cycle.

Since the clock speed choice affects all functional logic including timers, the DS89C420 implements several hardware switchback features that allow the clock speed to automatically return to the divide-by-1 mode from a reduced cycle rate. Setting the SWB (PMR.5) bit to a 1 in software enables this switchback function.

When CD1 and CD0 are programmed to the divide-by-1024 mode and the SWB bit is also enabled, the system forces the clock-divide control bits to automatically reset to the divide-by-1 mode whenever the system detects an externally enabled (and allowed through nesting priorities) interrupt. The switchback occurs whenever one of the two conditions occur. The first switchback condition is initiated by the detection of a low on either $\overline{\text{INT0}}$, $\overline{\text{INT1}}$, $\overline{\text{INT3}}$ or $\overline{\text{INT5}}$, or a high on INT2 or INT4 when the respective pin has been programmed and allowed (through nesting priorities) to issue an interrupt. The second switchback condition occurs when either serial port is enabled to receive data and is found to have an active-low transition on the respective receive input pin. Serial port transmit activity also forces a switchback if the SWB is set. Note that the serial port activity, as related to the switchback, is independent of the serial port interrupt relationship. Any attempt to change the clock divider to the divide-by-1024 mode while the serial port is either transmitting or receiving has no effect, leaving the clock control in the divide-by-1 mode. Note also that the switchback interrupt relationship requires that the respective external interrupt source is allowed to actually generate an interrupt as defined by the priority of the interrupt and the state of the nested interrupts, before the switchback can actually occur. An interrupt by the serial port is not required, nor is the setting of serial port enable. Disabling external interrupts and serial port receive/transmission mode disable the automatic switchback mode. Clearing the SWB bit also disables the switchback, and all interrupt and serial port controls of the clock divider are disabled. All other clock modes ignore the switchback relationship and are unaffected by interrupts and serial port activity.

The basic divide-by-12 mode for the timers (TxMH, TxM = 00b), as well as the divide-by-32 and 64 for mode 2 on the serial ports, are maintained when running the processor with the oscillator divide ratio of 0.25, 0.5, and 1. Serial ports and timers track the oscillator cycles per machine cycle when the higher divide ratio of 1024 is selected, and require the switchback function to automatically return to the divide-by-1 mode for proper operation when a qualified event occurs. Table 14 summarizes the effect of clock mode on timer operation.

It is possible to enable a receive function on a serial port when incoming data is not present and then change to the higher divide ratio. An inactive serial port receive/transmit mode requires the receive input pin to remain high and all outgoing transmissions to be completed. During this inactive receive mode it is possible to change the clock-divide control bits from a divide-by-1 to a 1024 divide ratio. In the case when the serial port is being used to receive or transmit data it is very important to validate an attempted change in the clock-divide control bits (read CD1 and CD0 to verify write was allowed) before proceeding with low-power program functions.

An external reset by the RST pin unconditionally exits the processor from stop mode. If the BGS bit is set to logic 1, the bandgap provides a reset while in stop mode if V_{CC} should drop below the V_{RST} level. If BGS is 0, no reset is generated if V_{CC} drops below V_{RST}.

When the stop mode is removed, the processor waits for 65,536 clock cycles for the internal flash memory to warm up before starting normal execution. Also, the processor waits for the crystal warmup period if not using the ring oscillator.

SERIAL I/O

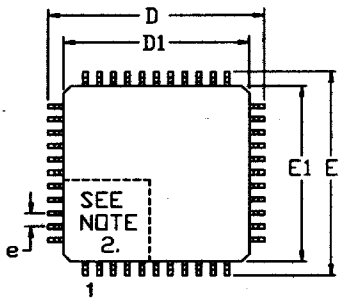
The DS89C420 provides a serial port (UART) that is identical to the 80C52. In addition, it includes a second hardware serial port that is a full duplicate of the standard one. This port optionally uses pins P1.2 (RXD1) and P1.3 (TXD1) and has duplicate control functions included in new SFR locations.

Both ports can operate simultaneously but can be at different baud rates or even in different modes. The second serial port has similar control registers (SCON1 at C0h, SBUF1 at C1h) as the original. The new serial port can only use timer 1 for timer-generated baud rates.

Control for serial port 0 is provided by the SCON0 register while its I/O buffer is SBUF0. Registers SCON1 and SBUF1 provide the same functions for the second serial port. A full description of the use and operation of both serial ports is in the *Ultra-High-Speed Flash Microcontroller User's Guide.*

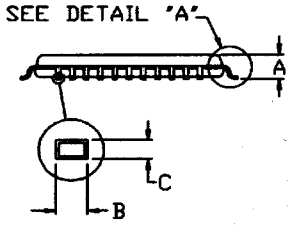
INSTRUCTION SET

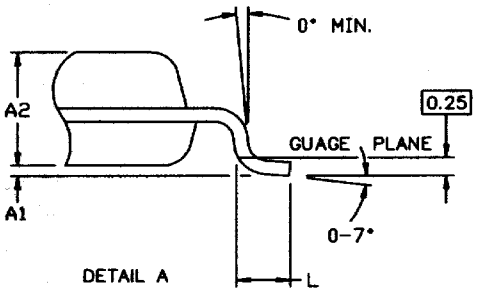
The DS89C420 instructions are 100% binary compatible with the industry standard 8051, and are only different in the number of machine cycles used for the instructions. Some special conditions and features should be considered when analyzing the DS89C420 instruction set. Full details are given in the *Ultra-High-Speed Flash Microcontroller User's Guide.*



NOTES:

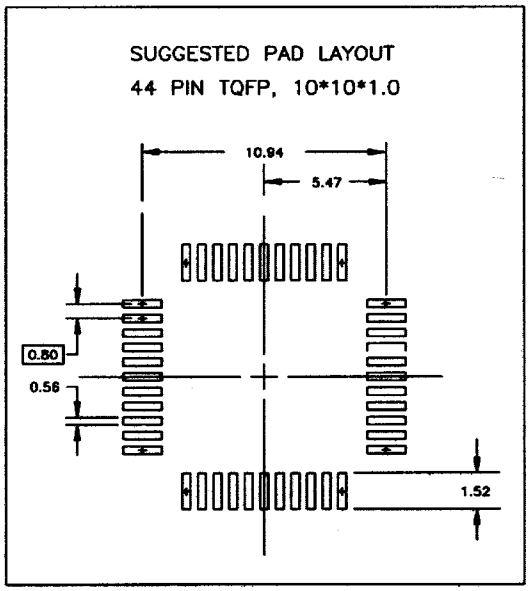
1. DIMENSIONS D1 AND E1 INCLUDE MOLD MISMATCH, BUT DO NOT INCLUDE MOLD PROTRUSION; ALLOWABLE PROTRUSION IS 0.25 MM PER SIDE.
2. DETAILS OF PIN 1 IDENTIFIER ARE OPTIONAL BUT MUST BE LOCATED WITHIN THE ZONE INDICATED.
3. ALLOWABLE DAMBAR PROTRUSION IS 0.08 MM TOTAL IN EXCESS OF THE B DIMENSION; AT MAXIMUM MATERIAL CONDITION. PROTRUSION NOT TO BE LOCATED ON LOWER RADIUS OR FOOT OF LEAD.
4. CONTROLLING DIMENSIONS: MILLIMETERS.





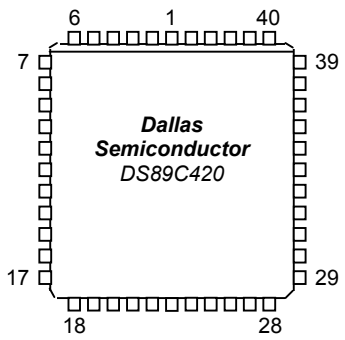
PKG	44-PIN	
DIM	MIN	MAX
A	–	1.20
A1	0.05	0.15
A2	0.95	1.05
D	11.80	12.20
D1	10.00 BSC	
E	11.80	12.20
E1	10.00 BSC	
L	0.45	0.75
e	0.80 BSC	
B	0.30	0.45
C	0.09	0.20

56-G4012-001



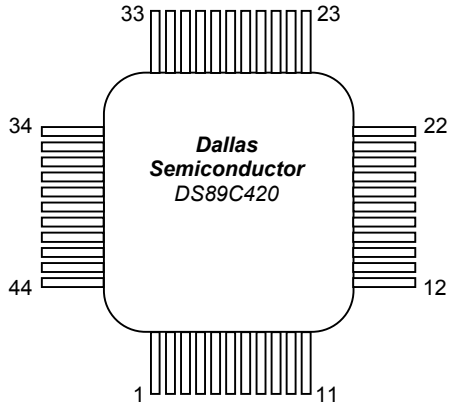
PIN CONFIGURATIONS

TOP VIEW



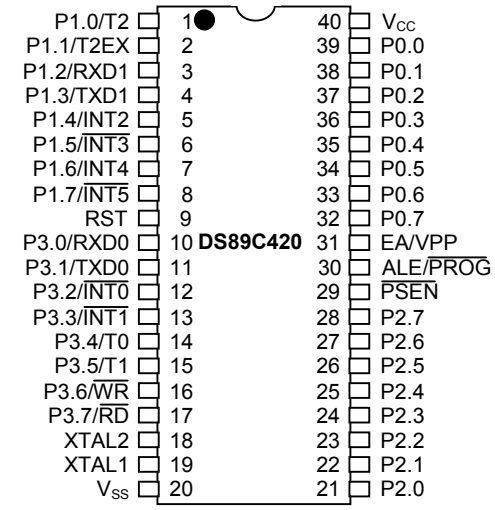
PLCC

TOP VIEW



TQFP

TOP VIEW



DIP