



# μPSD33XX (TURBO SERIES)

## Fast 8032 MCU With Programmable Logic

PRELIMINARY DATA

### FEATURES SUMMARY

#### 8-bit System On Chip for Embedded Control

The Turbo μPSD3300 Series combines a powerful, 8051-based microcontroller with a unique memory structure, programmable logic, and a rich peripheral mix to form the ideal SOC for embedded control. At its core is a fast, 4-cycle 8032 MCU with a 6-byte instruction prefetch queue and a 4-entry, fully associative branching cache to maximize MCU performance, enabling smaller loops of code to execute very quickly.

Code development is easily managed without a hardware In-Circuit Emulator by using the serial JTAG debug interface. JTAG is also used for In-System Programming (ISP), perfect for manufacturing and lab development. The 8032 core is coupled to Programmable System Device (PSD) architecture to optimize 8032 memory structure, offering two independent banks of Flash memory that can be placed at virtually any address within 8032 program or data space, and easily paged beyond 64K bytes using on-chip programmable decode logic. Dual Flash memory banks provide a robust solution for remote product updates in the field through In-Application Programming (IAP). Dual Flash banks also support EEPROM emulation, eliminating the need for external EEPROM chips.

A wide variety of Flash and SRAM memory sizes are available, some reaching the largest on the 8-bit MCU market today. General purpose programmable logic is included to build an endless variety of glue-logic, saving external chips. This SOC also provides a rich array of peripherals, including analog and supervisor functions.

#### ■ Fast Turbo 8032 MCU

- Advanced 8032 core: four clocks per instruction instruction pre-fetch; branching cache
- 10 MIPs peak performance @40MHz clock 5.0V V<sub>CC</sub> ... 8 MIPs peak @40MHz, 3.3V V<sub>CC</sub>
- 8032 core compatible with 3rd party tools
- Internal clock divider for low-power mode
- Three 8032 16-bit timers and external interrupts
- Dual XDATA pointers with auto incr & decr

- Programmable Counter Array (PCA)
  - Dual independent timer/counter blocks, each with three 16-bit timer/counters modules
  - Use any of the 6 modules as: 16-bit capture/compare, 16-bit timer/counter, 8/16 bit PWM.
- JTAG Debug and In-System Programming
  - Set Breakpoints, trace, single-step, display, modify memory, and SFRs; external event pin.
  - ISP the chip in 10-20sec, 8032 not involved.
- Programmable Logic, General Purpose
  - 16 Macrocells with architecture similar to industry standard 22V10 PLDs
  - Create shifters, state machines, chip-selects, glue-logic to keypads, panels, LCDs, others
  - Configure PLD with simple PSDsoft Express software ... download at no charge from web.
- Dual Flash Memories w/Memory Management
  - True READ-while-WRITE concurrent access
  - Main Flash size: 64K, 128K, or 256K Bytes
  - Secondary Flash size: 16K or 32K bytes
  - 100,000 min erase cycles, 15 year retention
  - On-chip programmable memory decode logic
- SRAM
  - 2K, 8K, or 32K Bytes; use as XDATA or code.
  - Capable of battery backup w/external battery.
- Peripheral Interfaces
  - (8) 10-bit ADC channels, 8 usec conversion time
  - I<sup>2</sup>C Master/Slave bus controller up to 800kHz
  - SPI Master bus controller, up to 10Mhz
  - Two standard UARTs with independent baud
  - IrDA protocol support up to 115K baud rate
  - 8032 address/data bus (80 pin package only)
  - Up to 46 I/O; eight can sink/source 10mA.
- Supervisor Functions
  - Watchdog timer, V<sub>CC</sub> monitor with 10ms Reset generator, Filtered Reset input

- Operating Range
  - 3.3V applications:  $V_{CC} = 3.3V \pm 10\%$
  - 5.0V applications:  $V_{CC}$ : Both  $5.0V \pm 10\%$  and  $3.3V \pm 10\%$  sources are required.
  - Temp:  $-40^{\circ}C$  to  $+85^{\circ}C$  (Industrial Range)

- TQFP Packaging
  - 52-pin (10x10mm) or 80-pin (12x12mm)

Figure 1. 52-lead, Thin, Quad, Flat Package

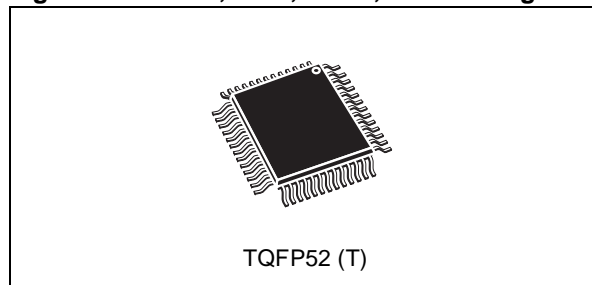


Figure 2. 80-lead, Thin, Quad, Flat Package

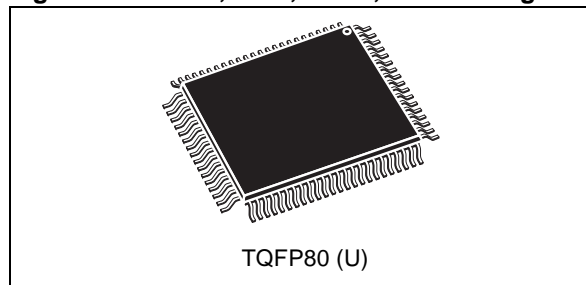


Table 1. μPSD33XX Device Selector Guide

Part No.	Main Flash Kbyte	2nd Flash Kbyte	SRAM Kbyte	PLD	I <sup>2</sup> C, SPI, Dual UART, IrDA	ADC, Supervisor	3 std timers, +6 prog timer/ PWM modules	GPIO	8032 Bus Pins	JTAG, ISP, & Debug	Pkg	Op V <sub>CC</sub>
μPSD 3312DV-40T6	64K	16K	2K	16 macro cells	Yes	Yes	Yes	Up to 37	No	yes	52-pin TQFP	3.3V±10%
μPSD 3312D-40T6	64K	16K	2K	16 macro cells	Yes	Yes	Yes	Up to 37	No	Yes	52-pin TQFP	3.3V & 5.0V±10%
μPSD 3333DV-40T6	128K	32K	8K	16 macro cells	Yes	Yes	Yes	Up to 37	No	Yes	52-pin TQFP	3.3V±10%
μPSD 3333D-40T6	128K	32K	8K	16 macro cells	Yes	Yes	Yes	Up to 37	No	Yes	52-pin TQFP	3.3V & 5.0V±10%
μPSD 3334DV-40T6 16	256K	32K	8K	16 macro cells	Yes	Yes	Yes	Up to 46	Yes	Yes	80-pin TQFP	3.3V±10%
μPSD 3334D-40U6	256K	32K	8K	16 macro cells	Yes	Yes	Yes	Up to 46	Yes	Yes	80-pin TQFP	3.3V & 5.0V±10%
μPSD 3354DV-40T6	256K	32K	32K	16 macro cells	Yes	Yes	Yes	Up to 37	No	Yes	52-pin TQFP	3.3V±10%
μPSD 3354D-40T6	256K	32K	32K	16 macro cells	Yes	Yes	Yes	Up to 37	No	Yes	52-pin TQFP	3.3V & 5.0V±10%
μPSD 3354DV-40T6 16	256K	32K	32K	16 macro cells	Yes	Yes	Yes	Up to 46	Yes	Yes	80-pin TQFP	3.3V±10%
μPSD 3354D-40U6	256K	32K	32K	16 macro cells	Yes	Yes	Yes	Up to 46	Yes	Yes	80-pin TQFP	3.3V & 5.0V±10%

TABLE OF CONTENTS

μPSD33XX HARDWARE DESCRIPTION ..... 12

MEMORY ORGANIZATION ..... 13

    Program Memory ..... 13

    Data Memory ..... 14

    RAM ..... 14

    XRAM-PSD ..... 14

8032 CORE PERFORMANCE ENHANCEMENTS ..... 14

    Instruction Pre-fetch Queue ..... 14

    Branch Cache (BC) ..... 14

MCU MODULE DISCRIPTION ..... 15

    Special Function Registers (SFRs) ..... 15

    Dual Data Pointers ..... 16

    Data Pointer Control Register, DPTC (85H) ..... 16

    Data Pointer Mode Register, DPTM (86H) ..... 17

    Debug Unit ..... 17

INTERRUPT SYSTEM ..... 18

    External Int0 ..... 18

    Timer 0 and 1 Inputs ..... 18

    Timer 2 Interrupt ..... 18

    I2C Interrupt ..... 18

    External Int1 ..... 18

    ADC Interrupt ..... 19

    PCA Interrupt ..... 19

    SPI Interrupt ..... 19

    UART Interrupt ..... 19

    Debug Interrupt ..... 19

    Interrupt Priority Structure ..... 21

    Interrupts Enable Structure ..... 21

POWER SAVINGS MODES ..... 23

    Idle Mode Function Activity ..... 23

    Idle Mode ..... 24

    Power-down Mode ..... 24

REDUCED FREQUENCY MODE ..... 25

CLOCK GENERATION ..... 25

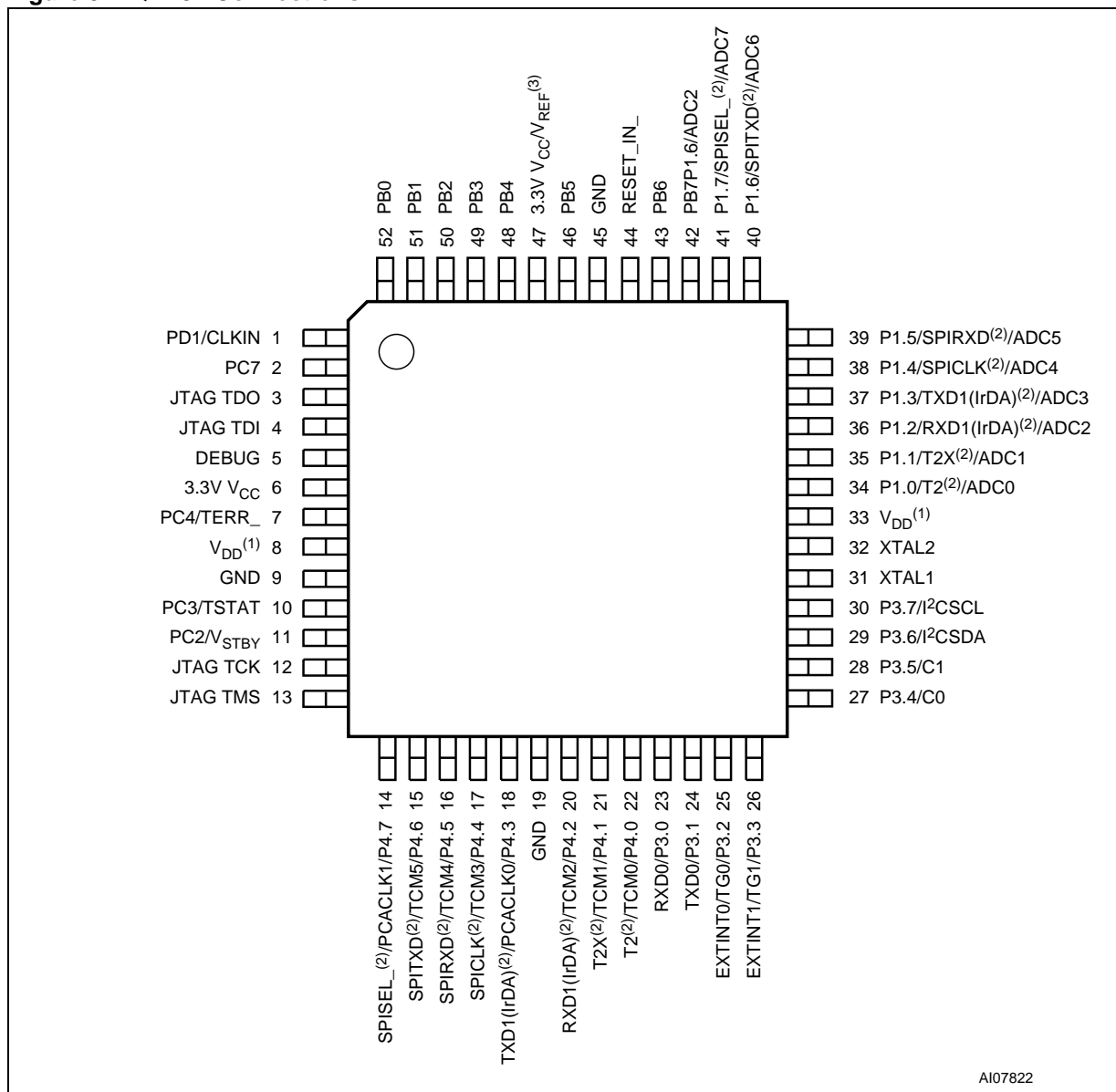
CPU CLOCK CONTROL REGISTER ..... 26

OSCILLATOR .....	27
I/O PORTS (MCU MODULE).....	27
Port 1 Register P1SFS0, Port 1 Register P1SFS1.....	28
Port 3 Configuration Register .....	28
Port 4 Configuration Register .....	28
Port 4 High Current Option.....	28
MCU MEMORY BUS INTERFACE .....	29
READ Bus Cycle (Code or XDATA) .....	29
WRITE Bus Cycle (XDATA).....	29
Bus Control Register (BUSCON) .....	29
SUPERVISORY .....	31
External Reset .....	31
Low V <sub>CC</sub> Voltage Reset.....	31
Watchdog Timer Overflow Reset .....	31
Debug Unit Reset.....	31
Reset Output .....	31
Power-up Reset .....	31
WATCHDOG TIMER .....	32
Watchdog Counter .....	32
WDT Registers.....	33
TIMER/COUNTERS (TIMER0, TIMER1, AND TIMER 2) .....	34
Timer 0 and Timer 1.....	34
Timer 2 .....	37
STANDARD SERIAL INTERFACE (UART) .....	41
Serial Port Control Register .....	41
IRDA INTERFACE TO INFRARED TRANSCEIVER .....	44
Baud Rate Select .....	45
I2C INTERFACE .....	46
I2C Registers Definition .....	46
Serial Status Register (S1STA) .....	48
Data Shift Register (S1DAT).....	48
Address Register (S1ADR).....	49
SPI (SYNCHRONOUS PERIPHERAL INTERFACE) .....	50
SPI Registers .....	50
Operation .....	53
SPI Configuration .....	53

Slave Select Output .....	54
Receive Operation .....	55
<b>ANALOG-TO-DIGITAL CONVERTOR (ADC) .....</b>	<b>56</b>
Port 1 ADC Channel Selects .....	56
<b>PROGRAMMABLE COUNTER ARRAY (PCA) WITH PWM .....</b>	<b>58</b>
PCA Block .....	58
Operation of TCM Modes .....	59
Capture Mode .....	59
Timer Mode .....	59
Toggle Mode .....	59
PWM Mode - (X8), Fixed Frequency .....	59
PWM Mode - (X8), Programmable Frequency .....	61
PWM Mode - Fixed Frequency, 16-bit .....	62
Writing to Capture/Compare Registers .....	62
Control Register Bit Definition .....	62
TCM Interrupts .....	63
<b>PSD MODULE .....</b>	<b>65</b>
Functional Overview .....	65
<b>PSD MODULE REGISTER DESCRIPTION AND ADDRESS OFFSET .....</b>	<b>67</b>
<b>PSD MODULE DETAILED OPERATION .....</b>	<b>68</b>
<b>MEMORY BLOCKS .....</b>	<b>68</b>
Primary Flash Memory and Secondary Flash memory Description .....	68
Memory Block Select Signals .....	68
Flash Memory Instructions .....	68
READ Flash Memory .....	70
Programming Flash Memory .....	71
Erasing Flash Memory .....	72
Specific Features .....	73
SRAM .....	74
Sector Select and SRAM Select .....	74
Page Register .....	77
<b>PLDs .....</b>	<b>78</b>
The Turbo Bit in PSD Module .....	78
Decode PLD (DPLD) .....	79
Complex PLD (CPLD) .....	80
Output Macrocell (OMC) .....	81
Product Term Allocator .....	82
Input Macrocells (IMC) .....	83

<b>I/O PORTS (PSD MODULE)</b> .....	<b>84</b>
General Port Architecture .....	84
Port Operating Modes .....	85
MCU I/O Mode .....	85
PLD I/O Mode .....	85
Address Out Mode .....	85
Peripheral I/O Mode .....	85
JTAG In-System Programming (ISP) .....	85
Port Configuration Registers (PCR) .....	87
Port Data Registers .....	88
Ports A and B – Functionality and Structure .....	89
Port C – Functionality and Structure .....	89
Port D – Functionality and Structure .....	89
<b>POWER MANAGEMENT</b> .....	<b>90</b>
PLD Power Management .....	92
PSD Chip Select Input (CSI, PD2) .....	92
Input Clock .....	92
Input Control Signals .....	92
<b>RESET TIMING AND DEVICE STATUS AT RESET</b> .....	<b>94</b>
Warm RESET .....	94
I/O Pin, Register and PLD Status at RESET .....	94
Reset of Flash Memory Erase and Program Cycles .....	94
<b>PROGRAMMING IN-CIRCUIT USING THE JTAG SERIAL INTERFACE</b> .....	<b>95</b>
Standard JTAG Signals .....	95
JTAG Extensions .....	95
Security and Flash Memory Protection .....	95
<b>INITIAL DELIVERY STATE</b> .....	<b>95</b>
<b>AC/DC PARAMETERS</b> .....	<b>96</b>
<b>MAXIMUM RATING</b> .....	<b>98</b>
<b>DC AND AC PARAMETERS</b> .....	<b>98</b>
<b>PART NUMBERING</b> .....	<b>117</b>
<b>PACKAGE MECHANICAL INFORMATION</b> .....	<b>118</b>
<b>REVISION HISTORY</b> .....	<b>122</b>

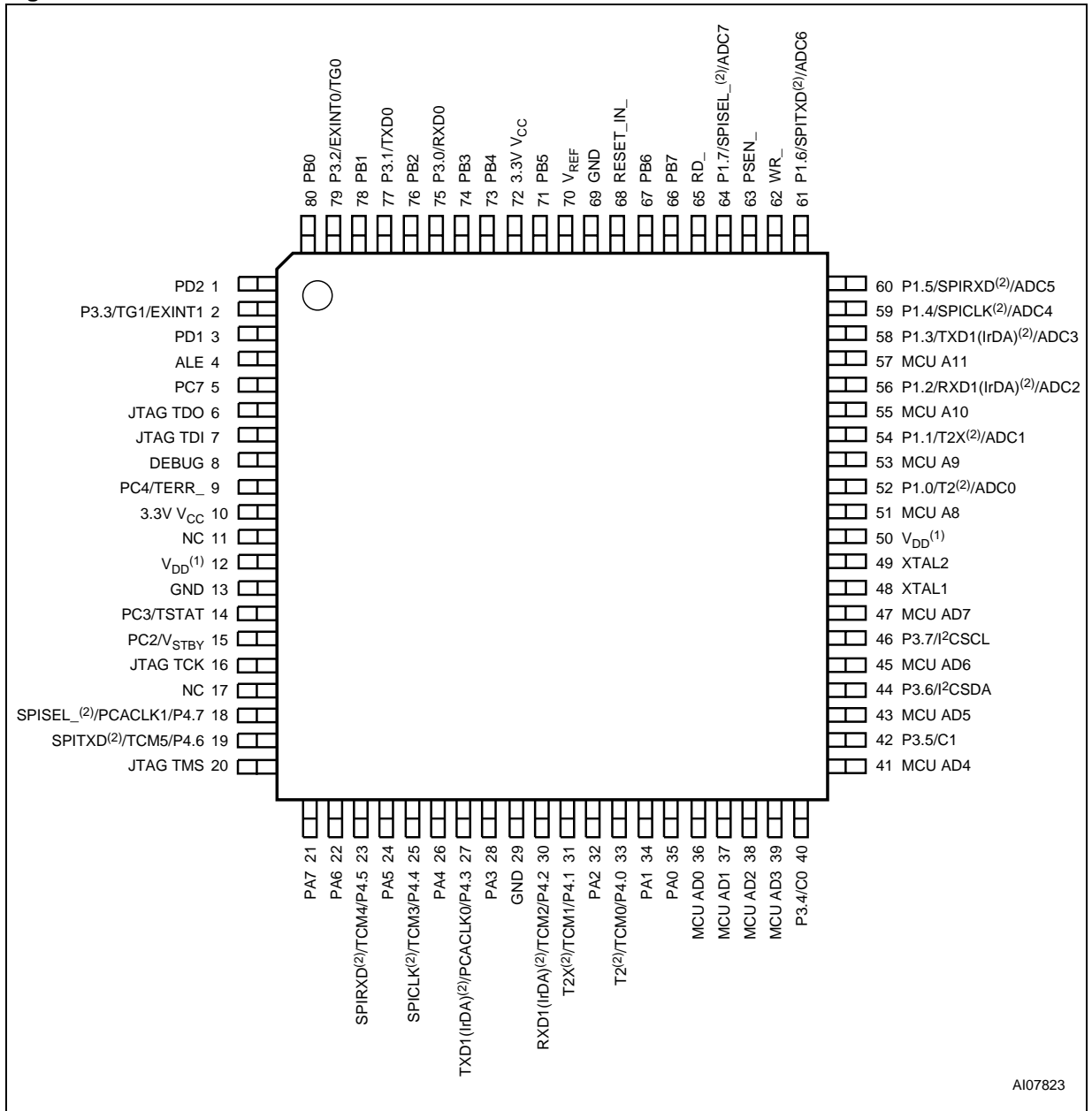
Figure 3. TQFP52 Connections



AI07822

- Note:
1. For 5V applications, V<sub>DD</sub> must be connected to a 5.0V source. For 3.3V applications, V<sub>DD</sub> must be connected to a 3.3V source.
  2. These signals can be used on one of two different ports (Port 1 or Port 4) for flexibility. Default is Port1.
  3. V<sub>REF</sub> and 3.3V V<sub>CC</sub> are shared in the 52-pin package only. ADC channels must use 3.3V as V<sub>REF</sub> for the 52-pin package.

Figure 4. TQFP80 Connections



A107823

Note: NC = Not Connected

1. For 5V applications, V<sub>DD</sub> must be connected to a 5.0V source. For 3.3V applications, V<sub>DD</sub> must be connected to a 3.3V source.
2. These signals can be used on one of two different ports (Port 1 or Port 4) for flexibility. Default is Port1.



**Table 2. Pin Descriptions**

Port Pin	Signal Name	80-Pin No.	52-Pin No.( <sup>1</sup> )	In/Out	Function		
					Basic	Alternate 1	Alternate 2
MCUAD0	AD0	36	N/A	I/O	External Bus Multiplexed Address/Data bus A0/D0		
MCUAD1	AD1	37	N/A	I/O	Multiplexed Address/Data bus A1/D1		
MCUAD2	AD2	38	N/A	I/O	Multiplexed Address/Data bus A2/D2		
MCUAD3	AD3	39	N/A	I/O	Multiplexed Address/Data bus A3/D3		
MCUAD4	AD4	41	N/A	I/O	Multiplexed Address/Data bus A4/D4		
MCUAD5	AD5	43	N/A	I/O	Multiplexed Address/Data bus A5/D5		
MCUAD6	AD6	45	N/A	I/O	Multiplexed Address/Data bus A6/D6		
MCUAD7	AD7	47	N/A	I/O	Multiplexed Address/Data bus A7/D7		
MCUA8	A8	51	N/A	O	External Bus, Addr A8		
MCUA9	A9	53	N/A	O	External Bus, Addr A9		
MCUA10	A10	55	N/A	O	External Bus, Addr A10		
MCUA11	A11	57	N/A	O	External Bus, Addr A11		
P1.0	T2 ADC0	52	34	I/O	General I/O port pin	Timer 2 Count input (T2)	ADC Channel 0 input (ADC0)
P1.1	T2EX ADC1	54	35	I/O	General I/O port pin	Timer 2 Trigger input (T2X)	ADC Channel 1 input (ADC1)
P1.2	RxD1 ADC2	56	36	I/O	General I/O port pin	UART1 or IrDA Receive (RxD1)	ADC Channel 2 input (ADC2)
P1.3	TXD1 ADC3	58	37	I/O	General I/O port pin	UART or IrDA Transmit (TxD1)	ADC Channel 3 input (ADC3)
P1.4	SPICLK ADC4	59	38	I/O	General I/O port pin	SPI Clock Out (SPICLK)	ADC Channel 4 input (ADC4)
P1.5	SPIRxD ADC6	60	39	I/O	General I/O port pin	SPI Receive (SPIRxD)	ADC Channel 5 input (ADC5)
P1.6	SPITxD ADC6	61	40	I/O	General I/O port pin	SPI Transmit (SPITxD)	ADC Channel 6 input (ADC6)
P1.7	SPISEL ADC7	64	41	I/O	General I/O port pin	SPI Slave Select (SPISEL)	ADC Channel 7 input (ADC7)
P3.0	RxD0	75	23	I/O	General I/O port pin	UART0 Receive (RxD0)	

Port Pin	Signal Name	80-Pin No.	52-Pin No.(1)	In/Out	Function		
					Basic	Alternate 1	Alternate 2
P3.1	TXD0	77	24	I/O	General I/O port pin	UART0 Transmit (TxD0)	
P3.2	EXINT0 TGO	79	25	I/O	General I/O port pin	Interrupt 0 input (EXTINT0)/Timer 0 gate control (TGO)	
P3.3	INT1	2	26	I/O	General I/O port pin	Interrupt 1 input (EXTINT1)/Timer 1 gate control (TG1)	
P3.4	C0	40	27	I/O	General I/O port pin	Counter 0 input (C0)	
P3.5	C1	42	28	I/O	General I/O port pin	Counter 1 input (C1)	
P3.6	I <sup>2</sup> CSDA	44	29	I/O	General I/O port pin	I <sup>2</sup> C Bus serial data (I <sup>2</sup> CSDA)	
P3.7	I <sup>2</sup> CSCL	46	30	I/O	General I/O port pin	I <sup>2</sup> C Bus clock (I <sup>2</sup> CSCL)	
P4.0	T2 TCM0	33	22	I/O	General I/O port pin	Program Counter Array0 PCA0-TCM0	Timer 2 Count input (T2)
P4.1	T2X TCM1	31	21	I/O	General I/O port pin	PCA0-TCM1	Timer 2 Trigger input (T2X)
P4.2	RXD1 TCM2	30	20	I/O	General I/O port pin	PCACLK0	UART1 or IrDA Receive (RxD1)
P4.3	TXD1 PCACLK0	27	18	I/O	General I/O port pin	PCACLK0	UART1 or IrDA Transmit (TxD1)
P4.4	SPICLK TCM3	25	17	I/O	General I/O port pin	Program Counter Array1 PCA1-TCM3	SPI Clock Out (SPICLK)
P4.5	SPIRXD TCM4	23	16	I/O	General I/O port pin	PCA1-TCM4	SPI Receive (SPIRXD)
P4.6	SPITXD	19	15	I/O	General I/O port pin	PCA1-TCM5	SPI Transmit (SPITxD)
P4.7	SPISEL PACCLK1	18	14	I/O	General I/O port pin	PCACLK1	SPI Slave Select (SPISEL)
V <sub>REF</sub>		70	N/A	I	Reference Voltage input for ADC		
RD <sub>-</sub>		65	N/A	O	READ Signal, external bus		
WR <sub>-</sub>		62	N/A	O	WRITE Signal, external bus		
PSEN <sub>-</sub>		63	N/A	O	PSEN Signal, external bus		
ALE		4	N/A	O	Address Latch signal, external bus		
RESET <sub>-</sub> IN <sub>-</sub>		68	44	I	Active low reset input		
XTAL1		48	31	I	Oscillator input pin for system clock		
XTAL2		49	32	O	Oscillator output pin for system clock		
DEBUG		8	5	I/O	I/O to the MCU Debug Unit		

Port Pin	Signal Name	80-Pin No.	52-Pin No.(1)	In/Out	Function		
					Basic	Alternate 1	Alternate 2
PA0		35	N/A	I/O	General I/O port pin		All Port A pins support: 1. PLD Macro-cell outputs, or 2. PLD inputs, or 3. Latched Address Out (A0-A7), or 4. Peripheral I/O Mode
PA1		34	N/A	I/O	General I/O port pin		
PA2		32	N/A	I/O	General I/O port pin		
PA3		28	N/A	I/O	General I/O port pin		
PA4		26	N/A	I/O	General I/O port pin		
PA5		24	N/A	I/O	General I/O port pin		
PA6		22	N/A	I/O	General I/O port pin		
PA7		21	N/A	I/O	General I/O port pin		
PB0		80	52	I/O	General I/O port pin		All Port B pins support: 1. PLD Macro-cell outputs, or 2. PLD inputs, or 3. Latched Address Out (A0-A7)
PB1		78	51	I/O	General I/O port pin		
PB2		76	50	I/O	General I/O port pin		
PB3		74	49	I/O	General I/O port pin		
PB4		73	48	I/O	General I/O port pin		
PB5		71	46	I/O	General I/O port pin		
PB6		67	43	I/O	General I/O port pin		
PB7		66	42	I/O	General I/O port pin		
JTAGTMS	TMS	20	13	I	JTAG pin (TMS)		
JTAGTCK	TCK	16	12	I	JTAG pin (TCK)		
PC2	V <sub>STBY</sub>	15	11	I/O	General I/O port pin	SRAM Standby voltage input (V <sub>STBY</sub> )	PLD Macrocell output, or PLD input
PC3	TSTAT	14	10	I/O	General I/O port pin	Optional JTAG Status (TSTAT)	PLD, Macrocell output, or PLD input
PC4	TERR	9	7	I/O	General I/O port pin	Optional JTAG Status (TERR)	PLD, Macrocell output, or PLD input
JTAGTDI	TDI	7	4	I	JTAG pin (TDI)		
JTAGTDO	TDO	6	3	O	JTAG pin (TDO)		
PC7		5	2	I/O	General I/O port pin		PLD, Macrocell output, or PLD input
PD1	CLKIN	3	1	I/O	General I/O port pin		1. PLD I/O 2. Clock input to PLD and APD
PD2	CSI	1	N/A	I/O	General I/O port pin		1. PLD I/O 2. Chip select of PSD Module
3.3V-V <sub>CC</sub>		10	6		V <sub>CC</sub> - MCU Module		
3.3V-V <sub>CC</sub>		72	47		V <sub>CC</sub> - MCU Module		
V <sub>DD</sub> 3.3V or 5V		12	8		V <sub>DD</sub> - PSD Module V <sub>DD</sub> - 3.3V for 3V V <sub>DD</sub> - 5V for 5V		
V <sub>DD</sub> 3.3V or 5V		50	33		V <sub>DD</sub> - PSD Module V <sub>DD</sub> - 3.3V for 3V V <sub>DD</sub> - 5V for 5V		
GND		13	9				
GND		29	19				
GND		69	45				
NC		11	N/A				
NC		17	N/A				

Note: 1. N/A = Signal Not Available on 52-pin package.

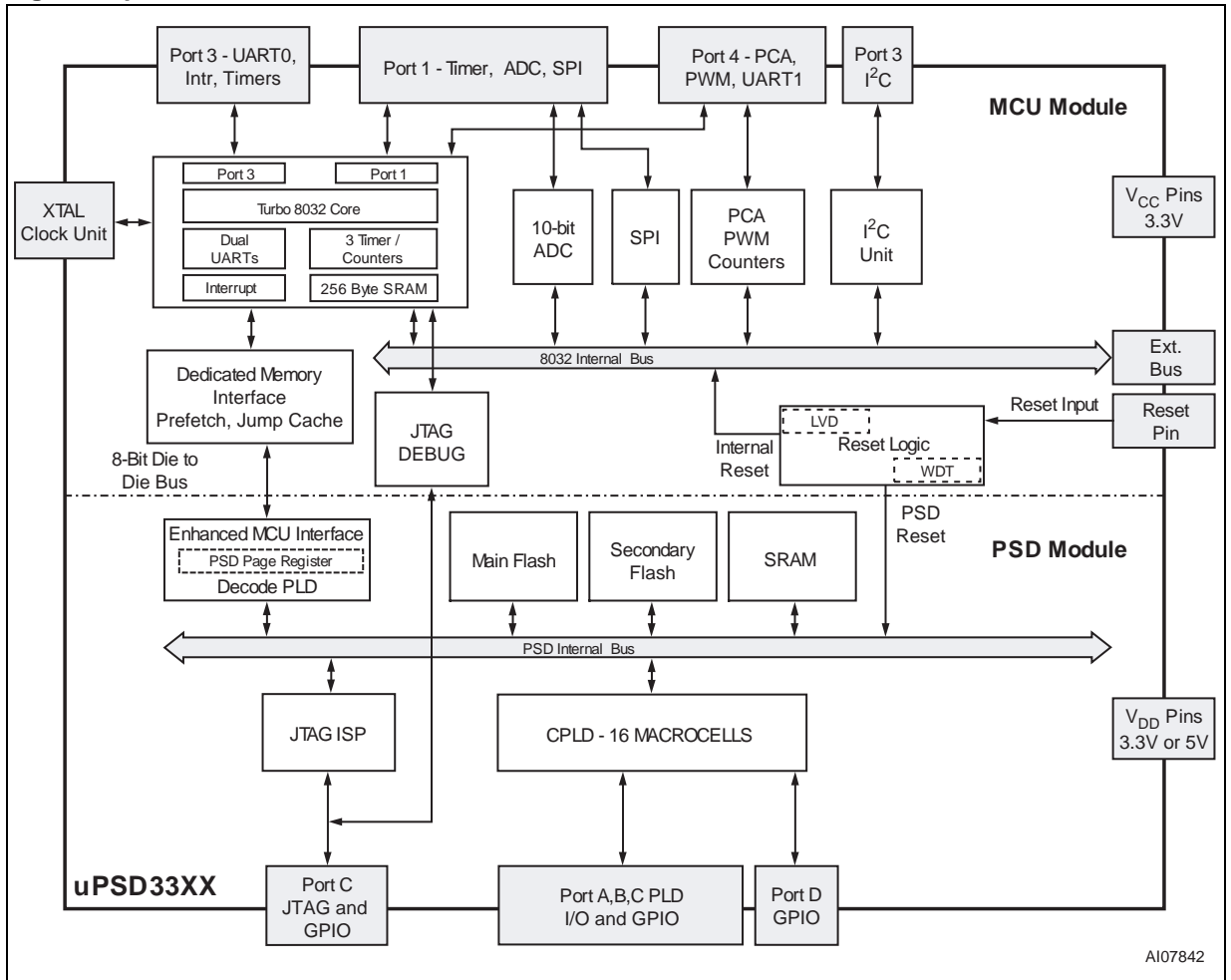
**μPSD33XX HARDWARE DESCRIPTION**

The μPSD33XX has a modular architecture with two main functional modules: the MCU Module and the PSD Module (see Figure 5). The MCU Module consists of a fast 4-cycle 8032 core, peripherals, and other system-supporting functions. The PSD Module provides configurable Program and Data memories to the 8032 CPU core. In addition, it has its own set of I/O ports and a PLD with 16 macrocells for general logic implementation. Ports A,B,C, and D are general purpose programmable I/O ports that have a port architecture which is different from I/O Ports in the MCU Module.

The PSD Module communicates with the MCU Module through the internal address, data bus (A0-A15, D0-D7) and control signals (RD\_, WR\_, PSEN\_, ALE, RESET\_). The user defines the Decoding PLD in the PSDsoft Development Tool and can map the resources in the PSD Module to any program or data address space.

The modules have their own separate power supply pins. The MCU module runs at 3.3V while the PSD module can be 3.3V or 5V, depending on the device. The MCU module can interface to a 5V PSD module as its I/O ports are 5V tolerant.

**Figure 5. μPSD33XX Functional Modules**



A107842

### MEMORY ORGANIZATION

The µPSD33XX Devices' fast 8032 Core has separate 64KB address spaces (see Figure 6) for Program memory and Data Memory. Program memory is where the 8032 executes instructions from. Data memory is used to hold data variables. Flash memory can be mapped in either program or data space. The Flash memory consists of two flash memory blocks: the main Flash and the Secondary Flash. Except during flash memory programming or update, Flash memory can only be read, not written to. A Page Register is used to access memory beyond the 64K bytes address space. Refer to the PSD Module for details on mapping of the Flash memory. The 8032 core has two types of data memory (internal and external) that can be read and written. The internal SRAM consists of 256 bytes, and includes the stack area.

The Special Function Registers (SFRs) occupy the upper 128 bytes of the internal SRAM, the registers can be accessed by Direct addressing only. There is a block of external SRAM in the µPSD33XX which resides in the PSD Module that can be mapped to any address space defined by the user.

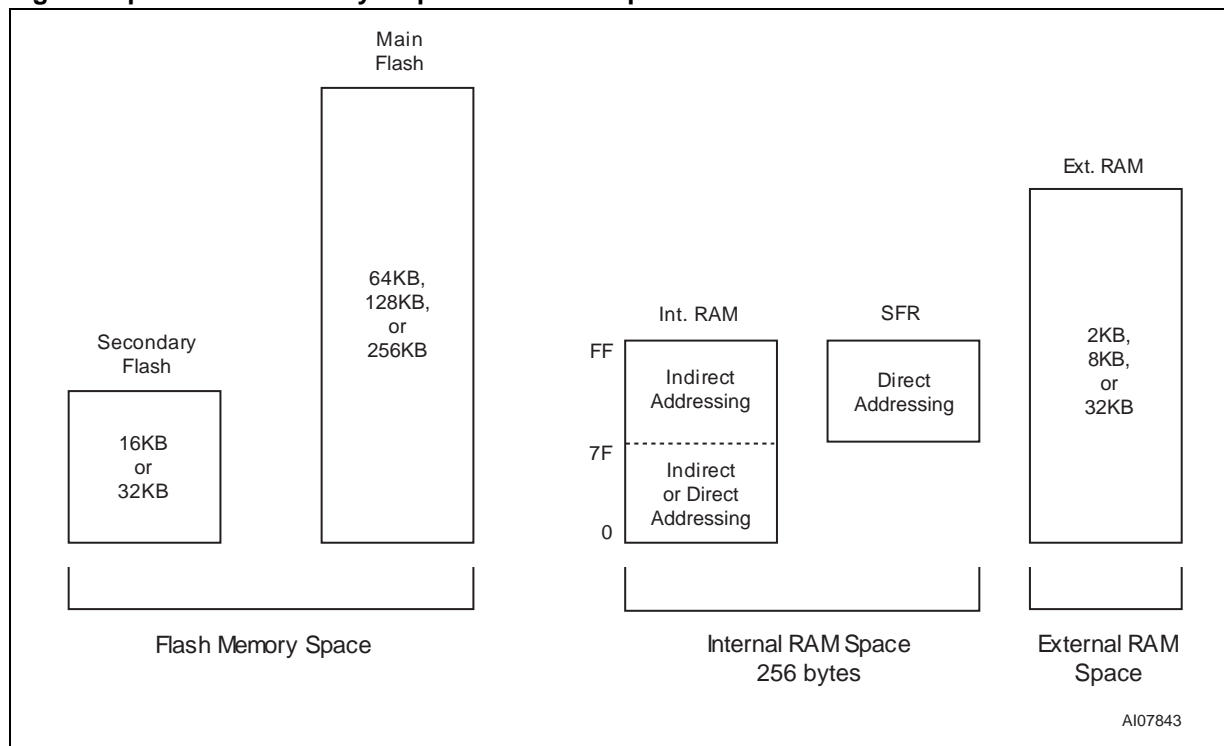
### Program Memory

The program memory consists of two Flash memory: Main Flash (64K, 128K, or 256K Bytes) and Secondary Flash (16K, 32K Bytes). The Flash memory can be mapped to any address space as defined by the user in the PSDsoft Tool. It can also be mapped to Data memory space during Flash memory update or programming.

After reset, the CPU begins execution from location 0000h. Each interrupt is assigned a fixed location in Program Memory. The interrupt causes the CPU to jump to that location, where it commences execution of the service routine. External Interrupt 0, for example, is assigned to location 0003h. If External Interrupt 0 is going to be used, its service routine must begin at location 0003h. If the interrupt is not going to be used, its service location is available as general purpose Program Memory.

The interrupt service locations are spaced at 8-byte intervals: 0003h for External Interrupt 0, 000Bh for Timer 0, 0013h for External Interrupt 1, 001Bh for Timer 1 and so forth. If an interrupt service routine is short enough (as is often the case in control applications), it can reside entirely within that 8-byte interval. Longer service routines can use a jump instruction to skip over subsequent interrupt locations, if other interrupts are in use.

Figure 6. µPSD33XX Memory Map and Address Space



## Data Memory

The internal data memory is divided into four physically separated blocks: 256 bytes of internal RAM, 128 bytes of Special Function Registers (SFRs), and XRAM-PSD in the PSD Module.

### RAM

Four register banks, each 8 registers wide, occupy locations 0 through 31 in the lower RAM area. Only one of these banks may be enabled at a time. The next 16 bytes, locations 32 through 47, contain 128 directly addressable bit locations. The stack depth is only limited by the available internal RAM space of 256 bytes.

## 8032 CORE PERFORMANCE ENHANCEMENTS

The μPSD33XX's CPU Core has implemented an Instruction Pre-fetch Queue (PFQ) and a Branch Cache (BC) to provide continuous code stream to the execution unit. The codes are fetched from PSD Module flash memory to be stored in the PFQ and Branch Cache.

### Instruction Pre-fetch Queue

The PFQ logic will fetch 8-bit data from the memories in the PSD Module. The logic will queue up to 6 bytes from the program instruction stream and will "feed" the requested bytes to the CPU core as the instructions are executed. The Pre-fetch Queue will make use of free memory bandwidth and try to keep the Pre-fetch Queue full with the next sequence of instructions. When a jump takes place (non sequential code access) the Pre-fetch Queue is cleared and will need to start over at from the new address. The Pre-fetch Queue must be able to "feed" instruction bytes to the CPU core so that it does not stall when the corresponding byte is in the queue. If the requested byte is not in the

### XRAM-PSD

The XRAM-PSD (2KB, 8KB, or 32KB) reside in the PSD Module and can be mapped to any address space

through the DPLD (Decoding PLD) as defined by the user in PSDsoft Development tool. The XRAM-PSD has a battery back-up feature that allows the data to be retained in the event of a power outage. The battery is connected to the Port C PC2 pin. This pin must be configured in PSDSoft to be battery back-up.

queue, then the CPU core will stall until the requested byte is read and given to the CPU and placed into the Pre-fetch Queue.

### Branch Cache (BC)

This is a 4-word cache that maintains the destination instructions (six bytes) from the previous 4 jumps. When the CPU takes a jump or interrupt that causes a non-sequential code access, the Branch Cache is checked to see if it has a match (address being jumped to). If the cache has a hit (match) then the corresponding word of data is transferred to the Pre-fetch Queue. At the same time the instruction bytes at the target address is supplied to the CPU. The Branch Cache must be able to supply the destination instruction to the CPU in time so that the CPU does not stall when there is a cache hit to an even or odd address. If there is a cache miss, then the CPU will stall until the destination words are fetched and placed into the Branch Cache. The Branch Cache will use LRU algorithm to replace cache entries.

**MCU MODULE DISCRIPTION**

This section provides a detail description of the MCU Module system functions and Peripherals, including:

- Special Function Registers
- Debug Unit
- Interrupts
- Power Saving Modes
- Oscillator and MCU Clock Generation
- I/O Ports

- MCU Bus Interface
- Supervisory Function (LVD and Watchdog)
- Timers/Counter
- UART
- IrDA Interface
- I<sup>2</sup>C Bus
- SPI Bus
- ADC
- Programmable Counter Array (PCA)

**Special Function Registers (SFRs)**

A map of the on-chip memory area called the Special Function Register (SFR) space is shown in Table 3. The SFRs can only be addressed directly in the address range from 80h to FFh. Sixteen addresses in the SFR space are both: byte- and bit-addressable. The bit-addressable SFRs are those

whose address ends in 0h and 8h (as indicated by \* in the table).

**Note:** In the SFRs not all of the addresses are occupied. Unoccupied addresses are not implemented on the chip and are reserved.

**Table 3. SFR Memory Map**

<b>F8</b>		CCON0		CCON2	CCON3					<b>FF</b>
<b>F0</b>	*B									<b>F7</b>
<b>E8</b>										<b>EF</b>
<b>E0</b>	*ACC						UDT1	UDT0		<b>E7</b>
<b>D8</b>	*SCON1	SBUF1		S1SETUP	S1CON	S1STA	S1DAT	S1ADR		<b>DF</b>
<b>D0</b>	*PSW		SPICLKD	SPISTAT	SPITDR	SPIRDR	SPICON0	SPICON1		<b>D7</b>
<b>C8</b>	*T2CON		RCAP2L	RCAP2H	TL2	TH2	IRDACON	DSTAT		<b>CF</b>
<b>C0</b>	*P4	CAPCOM L3	CAPCOM H3	CAPCOM L4	CAPCOM H4	CAPCOM L5	CAPCOM H5	PWMF1		<b>C7</b>
<b>B8</b>	*IP		PCAACL1	PCACH1	PCACON1	TCM MODE3	TCM MODE4	TCM MODE5		<b>BF</b>
<b>B0</b>	*P3	CAPCOM H1	CAPCOM L2	CAPCOM H2	PWMF0			IPA		<b>B7</b>
<b>A8</b>	*IE	TCM MODE0	TCM MODE1	TCM MODE2	CAPCOM L0	CAPCOM H0	WDKEY	CAPCOM L1		<b>AF</b>
<b>A0</b>	*P2		PCAACL0	PCACH0	PCACON0	PCASTA	WDRST	IEA		<b>A7</b>
<b>98</b>	*SCON0	SBUF0				BUSCON	DIR	DVR		<b>9F</b>
<b>90</b>	*P1	P3SFS	P4SFS0	P4SFS1	ADC0S	ADAT0	ADAT1	ACON		<b>97</b>
<b>88</b>	*TCON	TMOD	TL0	TL1	TH0	TH1	P1SFS0	P1SFS1		<b>8F</b>
<b>80</b>	*P0	SP	DPL	DPH		DPTC	PDTM	PCON		<b>87</b>

**Dual Data Pointers**

Data read access to the program memory and READ/WRITE access to the XRAM are executed using the data pointer DPTR as a 16-bit address register for indirect addressing mode. The DPTR consists of a high byte (DPH, 83H) and a low byte (DPL, 82H). Its intended function is to hold a 16-bit address. It may be manipulated as a 16-bit register or as two independent 8-bit registers.

The μPSD33XX has two data pointers (DPTR0 and DPTR1), one of which is selected by Bit DPSEL0 in the Data Pointer Control Register DPTC. After reset, these registers are set to “00H.” Only one DPTR is active at any time, and the selected PDTR resides in SFR address 83H and 82H. The DPTR which is not selected remains in the background and is not accessible by the CPU.

**Data Pointer Control Register, DPTC (85H)**

The control register allows the DPTR to be selected manually, or automatically switching between the two data pointers. Bit DPSEL0 selects one of two pointers. The automatic switching between DPTR0 and DPTR1 is controlled by Bit AT (Auto Toggle). When Bit AT is set, Bit DPSEL0 is toggled automatically every time after the DPTR is accessed. Detailed description for register DPTC is shown in Table 4 and Table 5.

The data pointer currently selected by the PSEL0 Bit can be modified, whereas the other data pointers are kept in the background and remain unchanged.

**Table 4. Data Pointer Control Register, DPTC, Bit Definition (85H, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	AT	–	–	–	–	–	DPSEL0

**Table 5. Data Pointer Control Register Details**

BIT	SYMBOL	RW	Definition
7	–		Reserved
6	AT	RW	0 = Manual Select Data Pointer 1 = Auto Toggle between DPTR0 and DPTR1
5-1	–		Reserved
0	DPSEL0	RW	0 = DPTR0 Selected 1 = DPTR1 Selected

Note: Standard increment instruction on Register DPTC can be used to toggle Bit DPSEL0.



**Data Pointer Mode Register, DPTM (86H)**

The µPSD33XX provides automatic increment or decrement of content of the working DPTR through the DPTM register. The content of the working DPTR is modified at the access time. De-

tailed description for DPTM is shown in Table 6 and Table 7.

The automatic decrement or increment function in the DPTM Register is effective only for the MOVX instruction.

**Table 6. Data Pointer Mode Register, DPTM Bit Definition (86H, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	–	–	–	MD11	MD10	MD01	MD00

**Table 7. Data Pointer Mode Register Details**

BIT	SYMBOL	RW	Definition
7-4	–		Reserved
3-2	MD[11:10]	RW	DPTR1 Mode Bits 00: DPTR1 No Change 01: Reserved 10: Auto Increment 11: Auto Decrement
1-0	MD[01:00]	RW	DPTR0 Mode Bits 00: DPTR0 No Change 01: Reserved 10: Auto Increment 11: Auto Decrement

**Debug Unit**

The MCU Module has a Debug Unit which supports debugging functions that are required in new PC board development. The JTAG port in the µPSD33XX is responsible for communications between the host development system and the Debug Unit. The basic debugging functions supported include:

- Halt or Start CPU execution
- Reset the CPU
- Single Step

- Four breakpoints, breaks on address/data
- Debug Interrupt to CPU at breakpoint
- Program tracing
- READ/WRITE to SFR, PC and Memory

The Debug pin can be configured in the host system to generate an output pulse for external triggering when a break condition is met. It can also be configured as an event input to the breakpoint logic in the Debug Unit. If not used, this pin should be pulled high.

## INTERRUPT SYSTEM

There are interrupt requests from 10 sources as follows:

- Debug Interrupt
- INT0 External Interrupt
- UART0 and UART1 Interrupt
- Timer 0 Interrupt
- I<sup>2</sup>C Interrupt
- INT1 External Interrupt
- ADC Interrupt
- Timer 1 Interrupt
- SPI Interrupt
- Timer 2 Interrupt
- PCA Interrupt

### External Int0

- The INT0 can be either level-active or transition-active depending on Bit IT0 in register TCON. The flag that actually generates this interrupt is Bit IE0 in TCON.
- When an external interrupt is generated, the corresponding request flag is cleared by the hardware when the service routine is vectored to only if the interrupt was transition activated.
- If the interrupt was level activated then the interrupt request flag remains set until the requested interrupt is actually generated. Then it has to deactivate the request before the interrupt service routine is completed, or else another interrupt will be generated.

### Timer 0 and 1 Inputs

- Timer 0 and Timer 1 Interrupts are generated by TF0 and TF1 which are set by an overflow

of their respective Timer/Counter registers (except for Timer 0 in Mode 3).

- These flags are cleared by the internal hardware when the interrupt is serviced.

### Timer 2 Interrupt

- Timer 2 Interrupt is generated by TF2 which is set by an overflow of Timer 2. This flag has to be cleared by the software - not by hardware.
- It is also generated by the T2EX signal (Timer 2 External Interrupt P1.1) which is controlled by EXEN2 and EXF2 Bits in the T2CON register.

### I<sup>2</sup>C Interrupt

- The interrupt of the I<sup>2</sup>C is generated by Bit INTR in the register S1STA.
- This flag is cleared by hardware.

### External Int1

- The INT1 can be either level-active or transition-active, depending on Bit IT1 in register TCON.  
The flag that actually generates this interrupt is Bit IE1 in TCON.
- When an external interrupt is generated, the corresponding request flag is cleared by the hardware when the service routine is vectored to, but only if the interrupt was transition-activated.
- If the interrupt was level-activated, then the interrupt request flag remains set until the requested interrupt is actually generated. It then has to deactivate the request before the interrupt service routine is completed, or else another interrupt will be generated.

**ADC Interrupt**

- The ADC unit generates an interrupt when conversion is completed and AINTEN Bit of the ACON register is set.
- After the interrupt is served, software needs to clear the interrupt flag AINTF.

**PCA Interrupt**

- Each of the 6 TCMs can generate a "match or capture" interrupt when enabled. The two 16-bit counters can also generate two counter overflow interrupts.
- The 8 PCA interrupts are "ORed" to generate one interrupt to the CPU.
- After serving the interrupt, the software has to clear the flag in the Status register.

**SPI Interrupt**

- The SPI can generate interrupt when the receive buffer is full and the transmission buffer is empty.
- An interrupt can also be generated at the end of transmission or receive overrun.

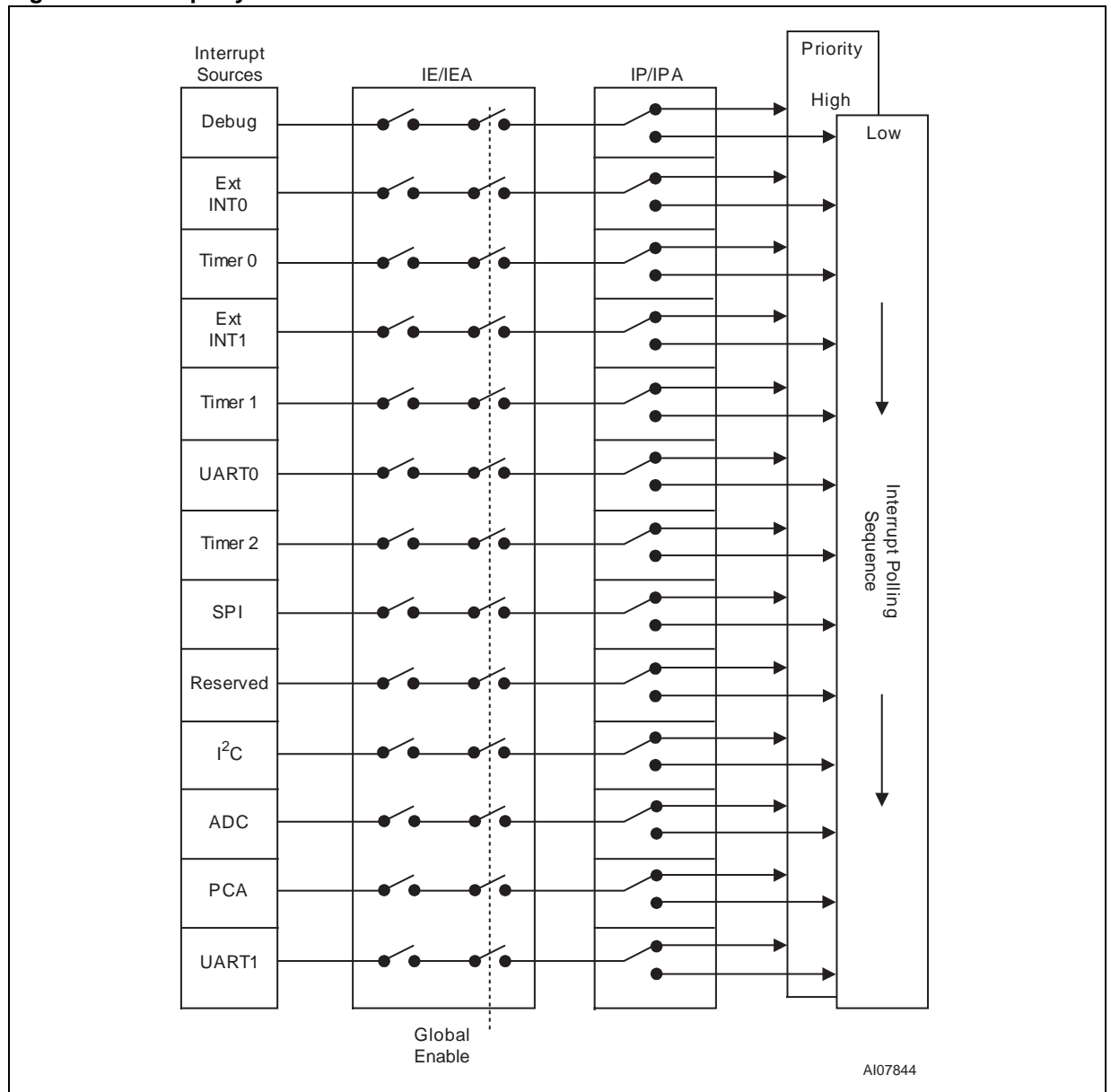
**UART Interrupt**

- The UART Interrupt is generated by RI (Receive Interrupt) or TI (Transmit Interrupt).
- When the UART Interrupt is generated, the corresponding request flag must be cleared with software. The interrupt service routine will have to check the various UART registers to determine the source and clear the corresponding flag.
- Both UARTs are identical, except for the additional interrupt controls in the Bit 4 of the additional interrupt control registers (A7H, B7H).

**Debug Interrupt**

- The Debug unit generates an interrupt when a Breakpoint condition is met.
- The interrupt has the highest priority.  
After the interrupt is served, the software needs to clear the interrupt flag in the status register.

Figure 7. Interrupt System



### Interrupt Priority Structure

Each interrupt source can be assigned one of two priority levels. Interrupt priority levels are defined by the Interrupt Priority special function registers IP and IPA.

0 = low priority

1 = high priority

A low priority interrupt may be halted by a high priority level interrupt. A high priority interrupt routine cannot be halted by any other interrupt source. If two interrupts of different priority occur simultaneously, the higher priority level request is serviced. If requests are of the same priority and are

received simultaneously, an internal polling sequence determines which request is serviced.

Thus, within each priority level, there is a second priority structure determined by the polling sequence.

### Interrupts Enable Structure

Each interrupt source can be individually enabled or disabled by setting or clearing a bit in the Interrupt Enable special function register IE and IEA. All interrupt sources can also be globally disabled by clearing Bit EA in the IE register.

**Table 8. Priority Level**

Priority	SOURCE	VECTOR ADDRESS
0 (Highest)	DEBUG	0063H
1	ExtlINT0	0003H
2	Timer 0	000BH
3	ExtlINT1	0013H
4	Timer 1	001BH
5	UART0	0023H
6	Timer 2 + EXF2	002BH
7	SPI	0053H
8	Reserved	0033H
9	I <sup>2</sup> C	0043H
10	ADC	003BH
11	PCA	005BH
12	UART1	004BH

**Table 9. Interrupt SFR**

SFR Addr	Reg Name	Bit Register Name								Reset Value	Comments
		7	6	5	4	3	2	1	0		
A7	IEA	ADC	SPI	PCA	ES1	–	–	EI <sup>2</sup> C		00	Interrupt Enable (2nd)
A8	IE	EA	EDB	ET2	ES0	ET1	EX1	ET0	EX0	00	Interrupt Enable
B7	IPA	PADC	PSPI	PPCA	PS1	–	–	PI <sup>2</sup> C		00	Interrupt Enable (2nd)
B8	IP	–	PDB	PT2	PS0	PT1	PX1	PT0	PX0	00	Interrupt Priority

**Table 10. Interrupt Enable SFR IE Bit Definition (A8H)**

BIT	SYMBOL	FUNCTION
7	EA	Disable all interrupts. 0 = No interrupt will be acknowledged 1 = Each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
6	EDB	Debug Unit Interrupt
5	ET2	Enable Timer 2 Interrupt
4	ES0	UART0 Interrupt
3	ET1	Enable Timer 1 Interrupt
2	EX1	Enable External Interrupt (INT1)
1	ET0	Enable Timer 0 Interrupt
0	EX0	Enable External Interrupt (INT0)

**Table 11. Interrupt Enable Addition SFR IEA Bit Definition (A7H)**

BIT	SYMBOL	FUNCTION
7	EADC	ADC Interrupt
6	ESPI	SPI Interrupt
5	EPCA	Programmable Counter Array Interrupt
4	ES1	UART1 Interrupt
3	–	Reserved
2	–	Reserved
1	EI <sup>2</sup> C	Enable I <sup>2</sup> C Interrupt
0	–	Reserved

**Table 12. Interrupt Priority Level SFR IP Bit Definition (B8H)**

BIT	SYMBOL	FUNCTION
7	–	Reserved
6	PDB	Debug Interrupt Level
5	PT2	Timer 2 Interrupt priority level
4	PS0	UART0 Interrupt priority level
3	PT1	Timer 1 Interrupt priority level
2	PX1	External Interrupt (INT1) priority level
1	PT0	Timer 0 Interrupt priority level
0	PX0	External Interrupt (INT0) priority level

**Table 13. Interrupt Priority Level Addition SFR IPA Bit Definition (B7H)**

BIT	SYMBOL	FUNCTION
7	PADC	ADC Interrupt priority level
6	PSPI	SPI Interrupt priority level
5	PPCA	PCA Interrupt level
4	PS1	UART1 Interrupt priority level
3	–	Not used
2	–	Not used
1	PI <sup>2</sup> C	I <sup>2</sup> C Interrupt priority level
0	–	Reserved

**POWER SAVINGS MODES**

Three software-selectable modes of reduced power consumption are implemented.

- Idle Mode
- Power-down Mode
- Reduced Frequency Mode

**Idle Mode Function Activity**

The following functions are switched off when the microcontroller enters the Idle Mode:

- CPU (halted - waiting for interrupt to exit halt)

The following functions remain active during Idle Mode (except when disabled by the control registers). Some of these functions may generate an interrupt or reset and thus terminate the Idle Mode.

- External Interrupts
- Timer 0, Timer 1 and Timer 2
- Watchdog Timer
- ADC
- I<sup>2</sup>C Bus Interface
- UART0 and UART1
- ADC
- SPI
- PCA (PWM)

**Table 14. Port Status at Power-saving Mode**

Mode	Ports 1, 3, 4	PCA	SPI	I <sup>2</sup> C	ADC
Idle	Maintain Data	Active	Active	Active	Active
Power-down	Maintain Data	Disable	Disable	Disable	Disable

**Table 15. Bus Signals at Power-down and Idle Mode**

Mode	ALE	PSEN_	RD_	WR_	AD0-7	A8-15
Idle	0	1	1	1	FF	FF
Power-down	0	1	1	1	FF	FF

**Idle Mode**

The instruction that sets PCON.0 is the last instruction executed in the normal operating mode before Idle Mode is activated. Once in the Idle Mode, the CPU status is preserved in its entirety: Stack pointer, Program counter, Program status word, Accumulator, RAM, and All other registers maintain their data during Idle Mode.

There are three ways to terminate the Idle Mode:

- Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware, terminating Idle Mode. The interrupt is serviced, and following return from interrupt instruction, RETI, the next instruction to be executed will be the one which follows the instruction that wrote a logic '1' to PCON.0.

- External hardware reset: the hardware reset is required to be active for two machine cycles to complete the RESET operation.
- Internal reset: the microcontroller restarts after 3 machine cycles in all cases.

**Power-down Mode**

The instruction that sets PCON.1 is the last executed prior to going into the Power-down Mode. Once in Power-down Mode, the oscillator is stopped. The contents of the on-chip RAM and the Special Function Register are preserved.

The Power-down Mode can be terminated by an external RESET.

**Table 16. PCON Register Bit Definition (87H, Reset Value 00H)**

BIT	SYMBOL	FUNCTION
7	SMOD0	Baud Rate Double Bit (UART0) 0 = No Doubling 1 = Doubling
6	SMOD1	Baud Rate Double Bit for 2nd UART (UART1) 0 = No Doubling 1 = Doubling
5	LVD	Disable LVD by setting this bit. 0 = Enable 1 = Disable
4	POR	Power-on reset sets this bit to '1.' See SUPERVISORY, page 31 for details. 0 = Clear with software 1 = Set by power-on reset generated by Supervisory circuit
3	RCLK1 <sup>(1)</sup>	Received Clock Flag (UART1)
2	TCLK1 <sup>(1)</sup>	Transmit Clock Flag (UART1)
1	PD	Activate Power-down Mode 0 = Exit from Power-down 1 = Enter into Power-down
0	IDL	Activate Idle Mode 0 = Exit from Idle Mode 1 = Enter into Idle Mode

Note: 1. See the T2CON Register (Table 38, page 38) for details of the flag description.



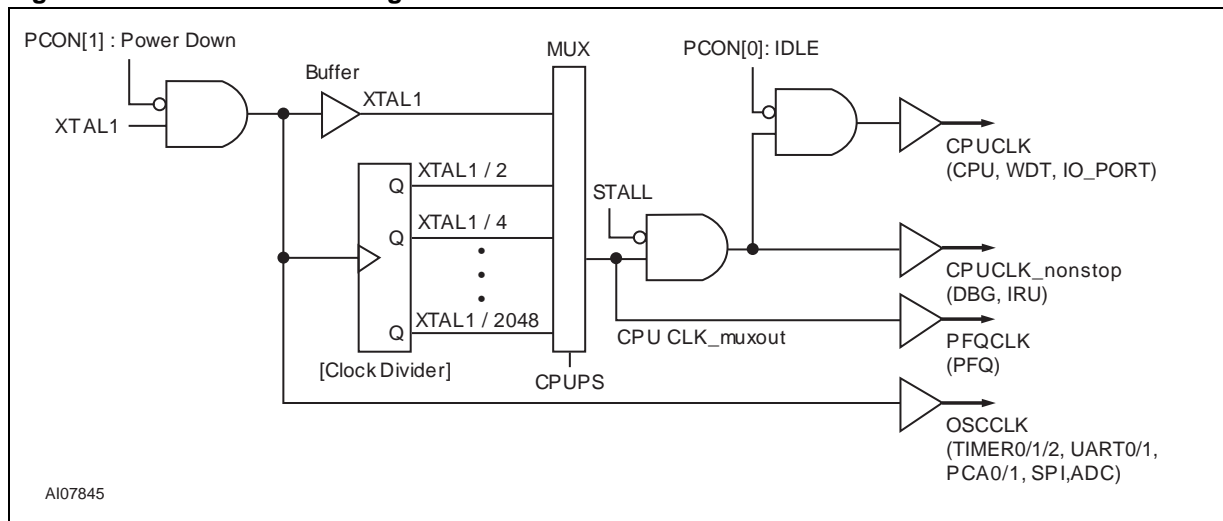
### REDUCED FREQUENCY MODE

The MCU consumes less power at lower clock frequency than at maximum frequency. The MCU can reduce the clock frequency by dividing the  $f_{OSC}$  with the divider as defined in CCON0 Register (see Figure 8). This mode allows the MCU to remain active while consuming less power at a slower speed. By changing back to the original di-

vider the MCU returns to normal mode. See CLOCK GENERATION, page 25 for more information.

In Reduced Frequency Mode, the Peripherals can still be functional at normal  $f_{OSC}$  frequency.

Figure 8. Clock Generation Logic



Note: 1. XTAL can be divided by 2 to 2048.

### CLOCK GENERATION

The clock unit uses the external crystal oscillator as a reference input clock, whose frequency is denoted by  $F_{OSC}$ . Based on the Frequency Selection Control registers (CCON0), the clock unit generates the following clocks:

**CPUCLK.** A clock for CPU and CPU-tightly-related peripherals (e.g., JTAG, WDT, IO\_PORT). This clock is disabled in both the Power-down Mode and the Idle Mode. The frequency of CPUCLK is  $f_{CPU}$ , which is obtained based on the  $f_{OSC}$ , CPUPS[2:0], and internal signals Idle, and Stall. When 'Idle' is 1,  $f_{CPU}$  is 0MHz. Otherwise,  $f_{CPU}$  is a function of  $f_{OSC}$ , CPUPS[2:0], and Stall. The Stall signal is generated from the Pre-fetch Queue (PFQ) block when the requested program code is not prepared in PFQ yet. CPUCLK is alive after a

breakpoint match or when the CPU is put in the halt state by the Debug Unit.

**CPUCLK\_nonstop.** A clock for CPU-loosely-related peripherals (e.g., Debug, Interrupt). This clock is the same as CPUCLK, except this clock is alive in Idle Mode.

**PFQCLK.** A clock for PFQ. This clock is same as CPUCLK except that this is alive even in Idle Mode or when the CPU is stalled.

**OSCCLK.** A clock for non-CPU related peripherals (e.g., TIMER0/1/2, UART0/1, PCA0/1, SPI). The frequency of OSCCLK is  $f_{OSC}$ , which is obtained from the external clock input. This clock is disabled only in the Power-down Mode.

The clock status is summarized in Table 17, page 26.

**Table 17. Clock Status**

Name	Modules	Power-saving Modes			Freq.
		NORMAL	IDLE	PD	
CPUCLK	CPU, JTAG, WDT, IO_PORT	ON	OFF	OFF	f <sub>CPU</sub>
CPUCLK_nonstop	Debug, Interrupt	ON		OFF	f <sub>CPU</sub>
PFQCLK	PFQ	ON		OFF	f <sub>CPU</sub>
JTAGCLK	JTAG	ON		OFF	f <sub>JTAG</sub>
OSCCLK <sup>(1)</sup>	TIMER0/1/2, UART0/1, PCA0/1, SPI, I <sup>2</sup> C, ADC	ON		OFF	f <sub>OSC</sub>

Note: 1. OSCCLK is the output of the crystal oscillator.

**CPU CLOCK CONTROL REGISTER**

The CPU Clock frequency is controlled by the CCON0 Register. The CPU is running at full frequency at power-up. The CPU Clock frequency can be changed any time by writing to the CPUPS Bits in the CCON0 register. After writing to the CCON0, the CPU Clock will be switched to the

new frequency immediately. The CPU Clock can be reduced by dividing the f<sub>OSC</sub> by 2 to 2048.

When the CPU is running at reduced clock frequency and the CPUAR Bit is set, it allows the CPU Clock to return to full frequency immediately when any interrupt occurs. This is achieved by automatically changing the CPUPS Bits to '000.'

**Table 18. CCON0 Register Bit Definition (0F9H, Reset Value 10H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reserved	Reserved	Reserved	DBGCE	CUPAR	CPUPS[2:0]		

**Table 19. CCON0 Register Bit Definition Details**

BIT	SYMBOL	RW	Definition
7-5	–		Reserved
4	DBGCE	RW	Debug Address Comparison Enable 0 = DBG Address Comparison is disabled 1 = DBG Address Comparison is enabled (Default during the reset period.) After reset, this bit is set to enable the Debug Unit's Address Comparison feature for debugging purposes. This bit should be set to '0' if the debugging function is not needed.
3	CPUAR	RW	Automatic CPU Clock Recovery 0 = There is no change of CPUPS[2:0] when an interrupt occurs. 1 = CPUPS[2:0] becomes 3'b000 whenever any interrupt occurs.
2:0	CPUPS	RW	CPUCLK Pre-Scaler 000: f <sub>CPU</sub> = f <sub>OSC</sub> (Default during the reset period) 001: f <sub>CPU</sub> = f <sub>OSC</sub> /2 010: f <sub>CPU</sub> = f <sub>OSC</sub> /4 011: f <sub>CPU</sub> = f <sub>OSC</sub> /8 100: f <sub>CPU</sub> = f <sub>OSC</sub> /16 101: f <sub>CPU</sub> = f <sub>OSC</sub> /32 110: f <sub>CPU</sub> = f <sub>OSC</sub> /1024 111: f <sub>CPU</sub> = f <sub>OSC</sub> /2048

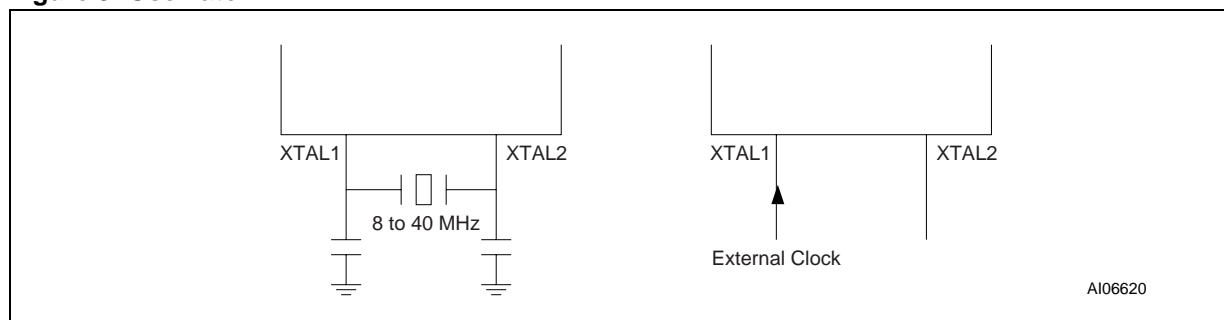
### OSCILLATOR

The oscillator circuit of the μPSD33XX Devices is a single stage, inverting amplifier in a Pierce oscillator configuration. The circuitry between XTAL1 and XTAL2 is basically an inverter biased to the transfer point. Either a crystal or ceramic resonator can be used as the feedback element to complete the oscillator circuit. Both are operated in parallel resonance. XTAL1 is the high gain amplifier input,

and XTAL2 is the output. To drive the μPSD33XX Devices externally, XTAL1 is driven from an external source and XTAL2 left open-circuit.

The μPSD33XX can run at maximum 40MHz clock. The CPU clock frequency can be configured in the CCON0 Register. However, the I<sup>2</sup>C bus requires a minimum 8MHz clock to be functional.

Figure 9. Oscillator



### I/O PORTS (MCU MODULE)

The MCU Module has three ports: Port 1, Port 3, and Port 4 (see Table 20). (Refer to the PSD MODULE, page 65 and I/O PORTS (PSD MODULE), page 84). The ports support:

- General Purpose I/O
- Alternate peripheral functions
- 5V tolerant
- High current on Port 4

The 80-pin μPSD33XX also has two ports in the MCU Module that are dedicated for the external MCU address and data bus. Ports 1, 3, and 4 are the same as in the standard 8032 microcontrollers,

with the exception of the additional special peripheral functions. All ports are bi-directional. Pins which are not configured as Alternate functions are normally bi-directional I/O.

The following SFR registers are used to control the mapping of alternate functions onto the I/O Port Bits (see Table 21 and Table 22). Port 1 and 4 alternate functions are controlled using the PXSFS1 registers. Port 3 alternate functions are controlled using the P3SFS register. After reset, the port SFR registers are cleared and are defaulted to general I/O.

Table 20. I/O Port Functions

Port Name	General Purpose I/O	Alternate 1 Function	Alternate 2 Function
Port 1	GPIO	Timer 2 - Pins 0, 1 UART1 - Pins 2, 3 SPI - Pins 4.. 7	ADC - Pins 0.. 7
Port 3	GPIO	UART0 - Pins 0, 1 Interrupt - Pins 2, 3 Timers - Pins 4, 5 I <sup>2</sup> C - Pins 6, 7	None
Port 4	GPIO	PCA0 - Pins 0.. 3 PCA1 - Pins 4-7	Timer 2 - Pins 0, 1 UART1 - Pins 2.. 3 SPI - Pins 4.. 7

**Port 1 Register P1SFS0, Port 1 Register P1SFS1**

P1SFS0 Register Bits 0.. 7 definition (see Table 21):

0 = Select pin as GPIO

1 = Select pin as Alternate function

When the bit in the P1SFS0 register is '1,' alternate peripheral functions are assigned to the pin. The new P1SFS1 register bit further selects which one of the alternate function to be enabled. Table 22 shows the alternate functions assigned to port 1 and how it can be selected.

**Table 21. Port 1 Register P1SFS0 (8EH, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0 = Port 1.7 1 = Alternate	0 = Port 1.6 1 = Alternate	0 = Port 1.5 1 = Alternate	0 = Port 1.4 1 = Alternate	0 = Port 1.3 1 = Alternate	0 = Port 1.2 1 = Alternate	0 = Port 1.1 1 = Alternate	0 = Port 1.0 1 = Alternate

**Table 22. Port 1 Register P1SFS1 (8FH, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2

**Port 3 Configuration Register**

Port 3 configuration is compatible to standard 8032, including alternate function pin assignments as shown in Table 23.

**Port 4 Configuration Register**

Port 4 has two registers: P4SFS0 Bit = 0 configures pin as GPIO and Bit = 1 configures pin as alternate function. P4SFS1 Bit determines which alternate function is to be enabled (see Table 24). The alternate 2 functions on Port 4 are the same as the alternate function 1 of Port 1. If an identical alternate function is assigned to both ports, the

Port 1 function has priority and the Port 4 function is disabled.

P4SFS0 Register Bits 0.. 7 definition:

0 = Select pin as GPIO

1 = Select pin as Alternate function

**Port 4 High Current Option**

Port 4 is a high current port (see Table 25). All 8 of the port pins are capable of a sink/source value of 10mA per pin in alternative function mode. The pins can sink 10mA in GPIO Mode. See the DC AND AC PARAMETERS, page 98, for VOL/VOH specification on Port 4.

**Table 23. Port 3 Register P3SFS (91H, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0 = Port 3.7 1 = Alternate	0 = Port 3.6 1 = Alternate	0 = Port 3.5 1 = Alternate	0 = Port 3.4 1 = Alternate	0 = Port 3.3 1 = Alternate	0 = Port 3.2 1 = Alternate	0 = Port 3.1 1 = Alternate	0 = Port 3.0 1 = Alternate

**Table 24. Port 4 Register P4SFS0 (92H, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0 = Port 4.7 1 = Alternate	0 = Port 4.6 1 = Alternate	0 = Port 4.5 1 = Alternate	0 = Port 4.4 1 = Alternate	0 = Port 4.3 1 = Alternate	0 = Port 4.2 1 = Alternate	0 = Port 4.1 1 = Alternate	0 = Port 4.0 1 = Alternate

**Table 25. Port 4 Register P4SFS1 (93H, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2	0 = Alternate1 1 = Alternate2



**MCU MEMORY BUS INTERFACE**

The MCU Module reads or writes to the PSD Module through the MCU memory bus. The Memory Bus is also available to external pins in the 80-pin package. The 8-bit MCU bus consists of standard 8032 bus signals.

**READ Bus Cycle (Code or XDATA)**

The READ bus cycle reads 8 bits per bus cycle and is identical for both the Program Fetch (PSEN) and Data Read (RD) in terms of timing and function. When the PSD Module is selected and either PSEN or RD is active, the PSD Module will drive data D0-D7 of the MCU bus. For program fetch, the MCU keeps the byte in the Pre-fetch Buffer. As for the XDATA READ bus cycle, the MCU routes the data byte to the CPU core directly. The READ bus cycle timing and length is controlled by the BUSCON Register.

**WRITE Bus Cycle (XDATA)**

The MCU writes one byte per bus cycle. The timing and length of the WRITE Bus Cycle is controlled by the BUSCON Register, which can be programmed by the software.

**Bus Control Register (BUSCON)**

The μPSD33XX has a programmable bus interface where the user can specify the length of a bus cycle. Based on the PQF Clock frequency, the number of clocks in a bus cycle can be changed to maximize data transfer rate (see Table 27). The μPSD33XX defaults to 6 PFQ clock for all READ and WRITE bus cycles after reset. Table 28 shows the minimum number of PFQ clocks in a bus cycle that are required for different PFQ Clock frequencies.

In addition, the BUSCON allows the user to set bits to turn on/off the Prefetch Queue and Branch Cache. In some real time applications, turning off the queue and cache provides determinable execution. The user may also wish to turn the queue and cache off during debugging.

**Table 26. BUSCON Register Bit Definition (9DH, Reset Value 2BH)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EPFQ	EBC	WRW1	WRW0	RDW1	RDW0	CW1	CW0

**Table 27. Number of PFQ Clocks Required to Optimize Bus Transfer Rate**

PFQ Clock Frequency	Code Data		READ Data		WRITE Data	
	3V <sup>(1)</sup>	5V <sup>(1)</sup>	3V <sup>(1)</sup>	5V <sup>(1)</sup>	3V <sup>(1)</sup>	5V <sup>(1)</sup>
25-40MHz	5	4	5	4	5	4
8-24MHz	3	3	4	4	4	4

Note: 1. V<sub>DD</sub> of the PSD Module

**Table 28. BUSCON Register Bit Definition Details**

Register Bit	Definition
CW1.. CW0	Code Fetch bus cycle: When PFQ fetches code, 2'b00: The code read from memory takes 3 PFQCLK clocks 2'b01: The code read from memory takes 4 PFQCLK clocks 2'b10: The code read from memory takes 5 PFQCLK clocks 2'b11: The code read from memory takes 6 PFQCLK clocks (default)
RDW1.. RDW0	XDATA READ bus cycle: 2'b00: The code read from XDATA takes 4 PFQCLK clocks 2'b01: The code read from XDATA takes 5 PFQCLK clocks 2'b10: The code read from XDATA takes 6 PFQCLK clocks (default) 2'b11: The code read from XDATA takes 7 PFQCLK clocks
WRW1.. WRW0	XDATA WRITE bus cycle 2'b00: The code read from XDATA takes 4 PFQCLK clocks 2'b01: The code read from XDATA takes 5 PFQCLK clocks 2'b10: The code read from XDATA takes 6 PFQCLK clocks (default) 2'b11: The code read from XDATA takes 7 PFQCLK clocks
EBC	Enable Branch Cache 0 = BC is disabled (default) 1 = BC is enabled
EPFQ	Enable Prefetch Queue 0 = PFQ is disabled (default) 1 = PFQ is enabled

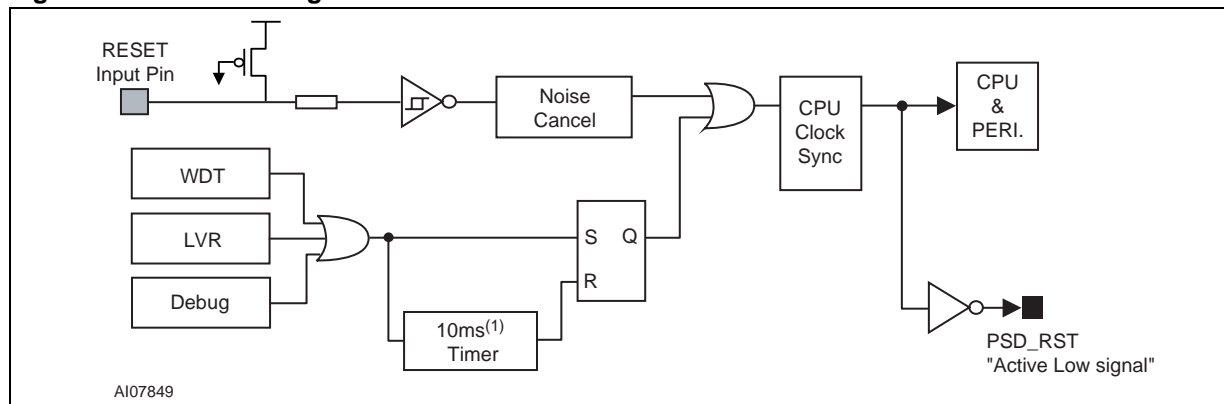
## SUPERVISORY

There are four ways to invoke a reset and initialize the μPSD33XX Devices:

- Via the external RESET pin
- Via the internal LVR Block.
- Via Watch Dog timer

Each RESET source will cause an internal reset signal to be active. The CPU responds by executing an internal reset and puts the internal registers in a defined state. This internal reset is also routed as an active low reset input to the PSD Module.

Figure 10. RESET Configuration



Note: 1. 10ms at 40MHz, 50ms at 8MHz.

### External Reset

The RESET pin is connected to a Schmitt trigger for noise reduction. A RESET is accomplished by holding the RESET pin LOW for at least 1ms at power-up while the oscillator is running. Refer to AC specification on other RESET timing requirements.

### Low V<sub>CC</sub> Voltage Reset

An internal reset is generated by the LVR circuit when the V<sub>CC</sub> drops below the reset threshold. After V<sub>CC</sub> returns to the reset threshold, the RESET signal will remain asserted for 10ms before it is released. On initial power-up the LVR is enabled (default). After power-up the LVR can be disabled via the LVREN Bit in the PCON Register.

**Note:** The LVR logic is functional in both the Idle and Power-down Modes. The reset circuit resides in the MCU Module which operates at 3.3V V<sub>CC</sub>. The reset threshold will always be: 2.5V +/-0.2V for all μPSD33XX devices.

This logic supports approximately 0.1V of hysteresis and 1μs noise-cancelling delay.

### Watchdog Timer Overflow Reset

The Watchdog timer generates an internal reset when its 24-bit counter overflows. See WATCHDOG TIMER, page 32 for details.

### Debug Unit Reset

The Debug Unit can generate a reset to the Supervisory circuit for debugging purpose. Under normal operation, this reset source is disabled.

### Reset Output

The output of the reset logic resets the MCU, it also drives a PSD\_Reset signal (active low) which is connected to the Reset Input on the PSD Module. The output of the reset logic remains asserted for a minimum of approximately 10ms. This time base is calculated by counting the f<sub>OSC</sub> at 40Mhz to last a minimum of 10ms at 40Mhz. The time will be longer as the f<sub>OSC</sub> is lower.

### Power-up Reset

At power up, the internal reset generated by the LVD circuit is latched as a '1' in the PCON register (Bit POR). Software can read this bit and determine whether the last CPU reset is a power up or warm reset. This bit must be cleared with software.

**WATCHDOG TIMER**

The hardware watchdog timer (WDT) resets the μPSD33XX Devices when it overflows. The WDT is intended as a recovery method in situations where the CPU may be subjected to a software upset. To prevent a system reset the timer must be reloaded in time by the application software. If the processor suffers a hardware/software malfunction, the software will fail to reload the timer. This failure will result in a reset upon overflow thus preventing the processor running out of control.

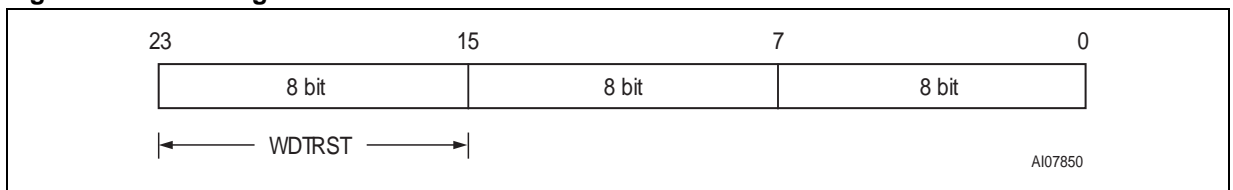
In the Idle Mode the watchdog timer and reset circuitry remain active. The WDT consists of a 24-bit counter, the Watchdog Timer RESET (WDRST)

SFR, and the Watchdog Key Register (WDKEY). Since the WDKEY register is loaded with 55h after reset, the Watchdog is disabled until it is enabled by the software.

**Watchdog Counter**

The 24-bit counter runs on machine cycle (4 f<sub>OSC</sub> clocks) and has a WDT reset period of about 1.6 seconds (see Figure 11). The 8th MSB of the counter is loaded from the Watch Dog Timer Clear Register (WDRST). By writing to the WDRST register, the user can change the WDT reset period. The 24-bit counter overflows when it reaches FFFFFFFh.

**Figure 11. Watchdog Counter**



$$\text{machine cycle} = (4T_{\text{OSC}} + T_{\text{STALL}})$$

$$\text{where, } T_{\text{OSC}} = \frac{1}{40\text{MHz}} = 0.025\mu\text{s} = 25\text{ns}$$

T<sub>STALL</sub>: the average waiting time due to PFQ/BC stall

For example, when t<sub>STALL</sub> is '0' and t<sub>OSC</sub> is 25ns, the total required cycle (to reach the overflow of the 24-bit counter that is clocked at machine cycle) is as below:

$$2^{24} \times 2^2 = 2^{26} = 2^6 \times 2^{20} = 64 \times (2^{10})^2 = 64 \text{ million (OSCcycle)}$$

Therefore, the reset period is:

$$\text{reset period} = 64\text{M} \times 25\text{ns} = 1.6\text{s}$$



## WDT Registers

### WDTKEY Register

- When this SFR is written as #55h, the WDT is disabled. Otherwise, writing any other values to the register will disable the WDT. Since the reset value of WDTKEY is #55h, the WDT is disabled at reset. The WDT is disabled after the reset that is generated by the WDT counter overflow.
- When the WDT is disabled, the 24-bit counter is cleared. Therefore, the new value of WDTRST must be written into the 24-bit counter after the WDT is enabled.
- In Idle Mode, the oscillator continues to run. To prevent the WDT from resetting the processor while in Idle, the user should

always set up a timer that will periodically exit Idle, service the WDT, and re-enter Idle Mode.

### WDTRST Register

- When this SFR is written as a value, the value is loaded into the upper 8 bits of 24-bit counter in WDT. And, the lower 16 bits are cleared.
- If the user write "WDTRST" by '04h, then the value of the 24-bit counter changes from 040000, 040001, 040002, ..., FFFFFFFF, then generates the WDT reset.

**Table 29. WDKEY: Watchdog Timer Key Register (0AEH, Reset Value 55H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WDKEY7	WDKEY6	WDKEY5	WDKEY4	WDKEY3	WDKEY2	WDKEY1	WDKEY0

**Table 30. WDKEY: Watchdog Timer Key Register Details**

Register Bit	Definition
WDKEY7.. 0	Enable or disable watchdog timer. Writing to WDKEY with data pattern 01010101 (= 55h) will disable the watchdog timer. Other data: enables the watchdog timer.

**Table 31. WDRST: Watchdog Timer Clear Register (0A6H, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WDRST7	WDRST6	WDRST5	WDRST4	WDRST3	WDRST2	WDRST1	WDRST0

**Table 32. WDRST: Watchdog Timer Clear Register Details**

Register Bit	Definition
WDRST[7.. 0]	To reset watchdog timer, write any value to this register. This value is loaded to the 8th most significant bits of the 24-bit counter.

**TIMER/COUNTERS (TIMER0, TIMER1, AND TIMER 2)**

The μPSD33XX Devices has three 16-bit Timer/Counter registers: Timer 0, Timer 1 and Timer 2. All of them can be configured to operate either as timers or event counters and are compatible with standard 8032 architecture (see Table 33).

In the “Timer” function, the register is incremented every 1/12 of the oscillator frequency ( $f_{OSC}$ ).

In the “Counter” function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T0 or T1. In this function, the external input is sampled by the counter. When the samples show a high in one machine cycle and a low in the another, the count is incremented. The new count value appears in the register during the cycle following the one in which the transition was detected. The maximum

count rate is 1/24 of the  $f_{OSC}$  as in standard 8032. There are no restrictions on the duty cycle of the external input signal, but to ensure that a given level is sampled at least once before it changes, it should be held for at least one full machine cycle. In addition to the “Timer” or “Counter” selection, Timer 0 and Timer 1 have four operating modes from which to select.

**Timer 0 and Timer 1**

The “Timer” or “Counter” function is selected by control bits C/T in the Special Function Register, TMOD (see Table 34). These Timer/Counters have four operating modes, which are selected by bit-pairs (M1, M0) in TMOD. Modes 0, 1, and 2 are the same for Timers/ Counters. Mode 3 is different.

**Table 33. TCON Register (88H, Reset Value 00H) - Timer 0, 1**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

**Table 34. TCON Register Details - Timer 0, 1**

Bit	Symbol	Function
7	TF1	Timer 1 Overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine
6	TR1	Timer 1 Run Control Bit. Set/cleared with software to turn Timer/Counter on or off
5	TF0	Timer 0 Overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine
4	TR0	Timer 0 Run Control Bit. Set/cleared with software to turn Timer/Counter on or off
3	IE1	Interrupt 1 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed
2	IT1	Interrupt 1 Type Control Bit. Set/cleared with software to specify falling-edge/low-level triggered external interrupt
1	IE0	Interrupt 0 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed
0	IT0	Interrupt 0 Type Control Bit. Set/cleared with software to specify falling-edge/low-level triggered external interrupt

**Mode 0.** Putting either Timer into Mode 0 makes it look like an 8048 Timer, which is an 8-bit Counter with a divide-by-32 prescaler. Figure 12 shows the Mode 0 operation as it applies to Timer 1.

In this mode, the Timer register is configured as a 13-bit register. As the count rolls over from all '1s' to all '0s,' it sets the Timer Interrupt flag TF1. The counted input is enabled to the Timer when TR1 = 1 and either GATE = 0 or /INT1 = 1. (Setting GATE = 1 allows the Timer to be controlled by external input /INT1 to facilitate pulse width measurements). TR1 is a control bit in the Special Function Register TCON (TCON Control Register). GATE is in TMOD (see Table 35 and Table 36, page 37).

The 13-bit register consists of all 8 bits of TH1 and the lower 5 bits of TL1. The upper 3 bits of TL1 are indeterminate and should be ignored. Setting the run flag does not clear the registers.

Mode 0 operation is the same for the Timer 0 as for Timer 1. Substitute TR0, TF0, and /INT0 for the corresponding Timer 1 signals in Figure 12. There are two different GATE Bits, one for Timer 1 and one for Timer 0.

**Mode 1.** Mode 1 is the same as Mode 0, except that the Timer register is being run with all 16 bits.

**Mode 2.** Mode 2 configures the Timer register as an 8-bit Counter (TL1) with automatic reload, as shown in Figure 13, page 36. Overflow from TL1 not only sets TF1, but also reloads TL1 with the contents of TH1, which is preset with software. The reload leaves TH1 unchanged. Mode 2 operation is the same for Timer/Counter 0.

**Mode 3.** Timer 1 in Mode 3 simply holds its count. The effect is the same as setting TR1 = 0.

Timer 0 in Mode 3 establishes TL0 and TH0 as two separate counters. The logic for Mode 3 on Timer 0 is shown in Figure 14. TL0 uses the Timer 0 control Bits: C/T, GATE, TR0, INT0, and TF0. TH0 is locked into a timer function (counting machine cycles) and takes over the use of TR1 and TF1 from Timer 1. Thus, TH0 now controls the "Timer 1" Interrupt.

Mode 3 is provided for applications requiring an extra 8-bit timer on the counter (see Figure 14, page 36). With Timer 0 in Mode 3, an μPSD33XX Devices can look like it has three Timer/Counters. When Timer 0 is in Mode 3, Timer 1 can be turned on and off by switching it out of and into its own Mode 3, or can still be used by the serial port as a baud rate generator, or in fact, in any application not requiring an interrupt.

**Figure 12. Timer/Counter Mode 0: 13-bit Counter**

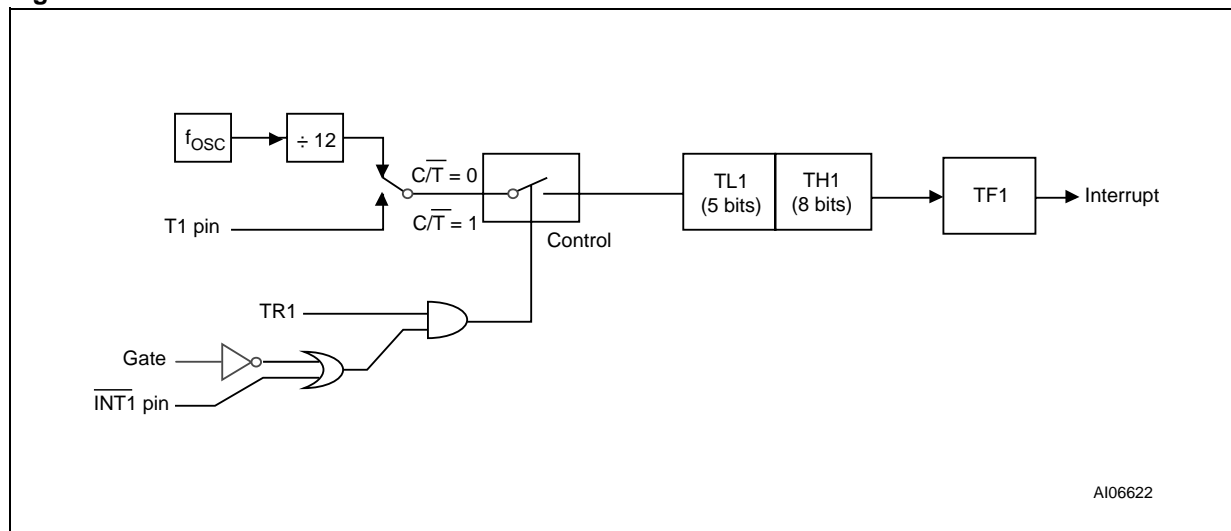


Figure 13. Timer/Counter Mode 2: 8-bit Auto-reload

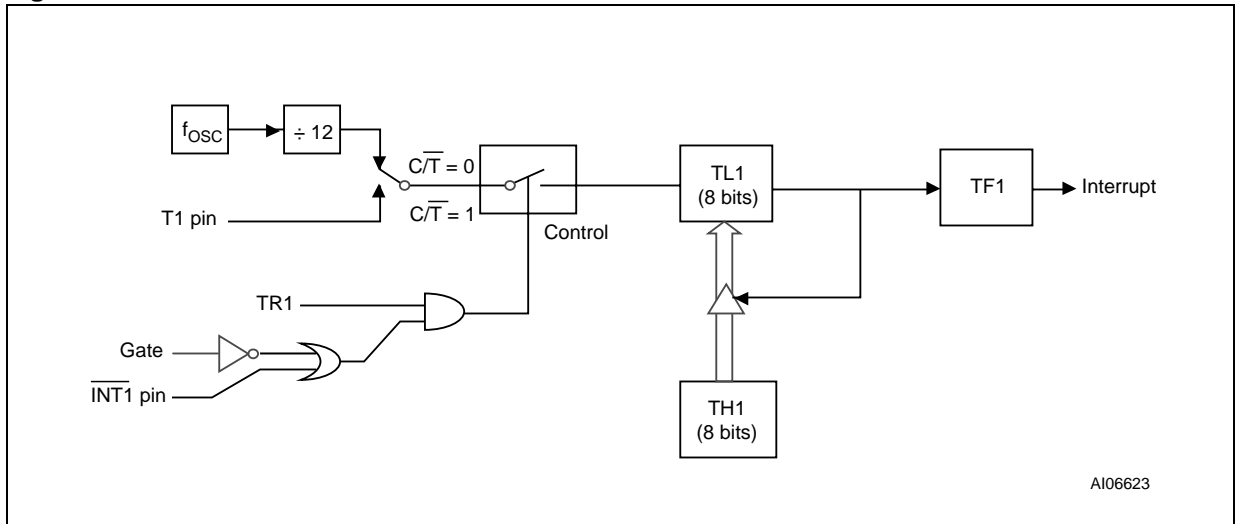
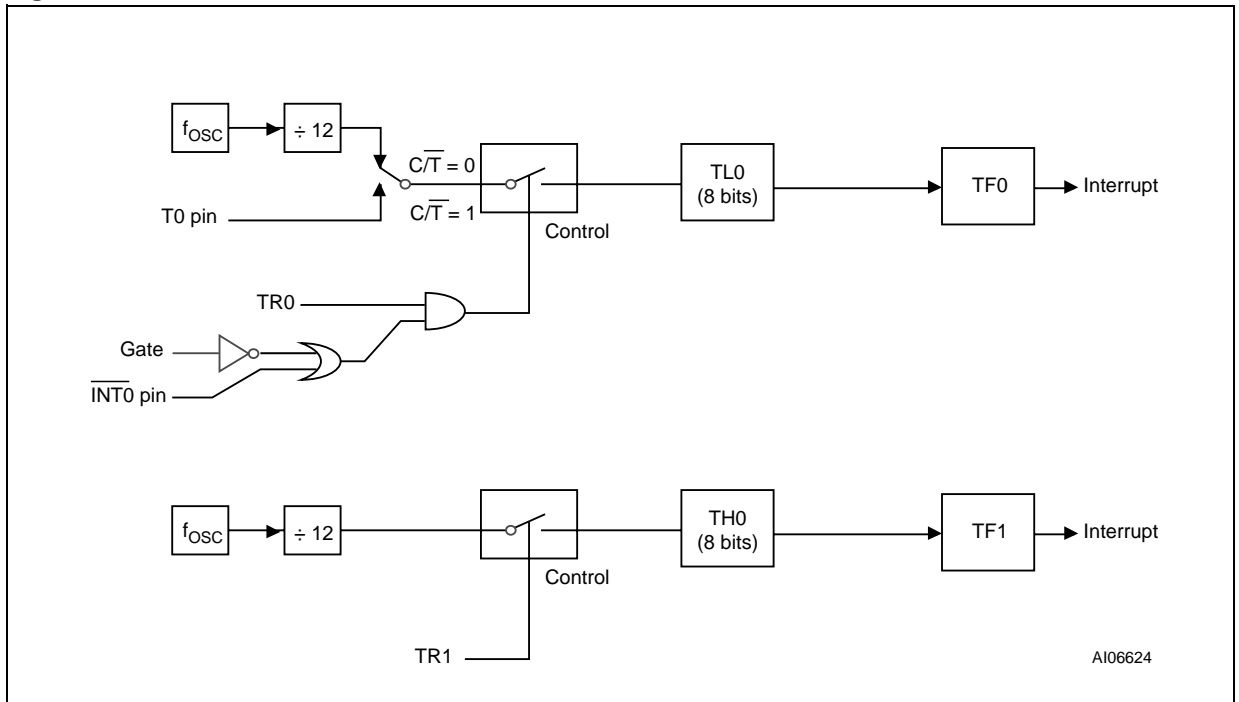


Figure 14. Timer/Counter Mode 3: Two 8-bit Counters



**Table 35. TMOD Register (TMOD)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
GATE	C/T	M1	M0	GATE	C/T	M1	M0

**Table 36. TMOD Register Details**

Bit	Symbol	Timer	Function
7	GATE	Timer 1	Gating control when set. Timer/Counter 1 is enabled only while INT1 pin is High and TR1 control pin is set. When cleared, Timer 1 is enabled whenever TR1 Control Bit is set
6	C/T		Timer or Counter selector, cleared for timer operation (input from internal system clock); set for counter operation (input from T1 input pin)
5	M1		(M1,M0)=(0,0): 13-bit Timer/Counter, TH1, with TL1 as 5-bit prescaler (M1,M0)=(0,1): 16-bit Timer/Counter. TH1 and TL1 are cascaded. There is no prescaler. (M1,M0)=(1,0): 8-bit auto-reload Timer/Counter. TH1 holds a value which is to be reloaded into TL1 each time it overflows (M1,M0)=(1,1): Timer/Counter 1 stopped
4	M0		
3	GATE	Timer 0	Gating control when set. Timer/Counter 0 is enabled only while INT0 pin is High and TR0 control pin is set. When cleared, Timer 0 is enabled whenever TR0 Control Bit is set
2	C/T		Timer or Counter selector, cleared for timer operation (input from internal system clock); set for counter operation (input from T0 input pin)
1	M1		(M1,M0)=(0,0): 13-bit Timer/Counter, TH0, with TL0 as 5-bit prescaler (M1,M0)=(0,1): 16-bit Timer/Counter. TH0 and TL0 are cascaded. There is no prescaler. (M1,M0)=(1,0): 8-bit auto-reload Timer/Counter. TH0 holds a value which is to be reloaded into TL0 each time it overflows (M1,M0)=(1,1): TL0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits. TH0 is an 8-bit timer only controlled by Timer 1 Control Bits
0	M0		

**Timer 2**

Like Timers 0 and 1, Timer 2 can operate as either an event timer or as an event counter. This is selected by Bit C/T2 in the special function register T2CON. It has three operating modes: capture, autoloading, and baud rate generator, which are selected by bits in the T2CON as shown in Table 37 and Table 38, page 38. In the Capture Mode there are two options which are selected by Bit EXEN2 in T2CON. If EXEN2 = 0, then Timer 2 is a 16-bit timer or counter which upon overflowing sets Bit TF2, the Timer 2 Overflow Bit, which can be used to generate an interrupt. If EXEN2 = 1, then Timer 2 still does the above, but with the added feature that a 1-to-0 transition at external input T2EX causes the current value in the Timer 2 registers, TL2 and TH2, to be captured into registers RCAP2L and RCAP2H, respectively. In addition, the transition at T2EX causes Bit EXF2 in T2CON to be set, and EXF2 like TF2 can generate an in-

terrupt. The Capture Mode is illustrated in Figure 15, page 39.

In the Auto-reload Mode, there are again two options, which are selected by bit EXEN2 in T2CON (see Table 39, page 39). If EXEN2 = 0, then when Timer 2 rolls over it not only sets TF2 but also causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2L and RCAP2H, which are preset with software. If EXEN2 = 1, then Timer 2 still does the above, but with the added feature that a 1-to-0 transition at external input T2EX will also trigger the 16-bit reload and set EXF2. The Auto-reload Mode is illustrated in Standard Serial Interface (UART) Figure 16, page 40. The Baud Rate Generation Mode is selected by (RCLK, RCLK1)=1 and/or (TCLK, TCLK1)=1. It will be described in conjunction with the serial port.

**Table 37. T2CON: Timer/Counter 2 Control Register (C8H, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T $\bar{2}$	CP/RL $\bar{2}$

**Table 38. T2CON Register Details**

Bit	Symbol	Function
7	TF2	Timer 2 Overflow flag. Set by a Timer 2 overflow, and must be cleared with software. TF2 will not be set when either (RCLK, RCLK1)=1 or (TCLK, TCLK)=1
6	EXF2	Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX and EXEN2=1. When Timer 2 Interrupt is enabled, EXF2=1 will cause the CPU to vector to the Timer 2 Interrupt routine. EXF2 must be cleared with software
5	RCLK <sup>(1)</sup>	Receive Clock flag (UART0). When set, causes the serial port to use Timer 2 overflow pulses for its receive clock in Modes 1 and 3. TCLK=0 causes Timer 1 overflow to be used for the receive clock
4	TCLK <sup>(1)</sup>	Transmit Clock flag (UART0). When set, causes the serial port to use Timer 2 overflow pulses for its transmit clock in Modes 1 and 3. TCLK=0 causes Timer 1 overflow to be used for the transmit clock
3	EXEN2	Timer 2 External Enable flag. When set, allows a capture or reload to occur as a result of a negative transition on T2EX if Timer 2 is not being used to clock the serial port. EXEN2=0 causes Time 2 to ignore events at T2EX
2	TR2	Start/Stop control for Timer 2. A logic 1 starts the timer
1	C/T $\bar{2}$	Timer or Counter Select for Timer 2. Cleared for timer operation (input from internal system clock, t <sub>CPU</sub> ); set for external event counter operation (negative edge triggered)
0	CP/RL $\bar{2}$	Capture/Reload flag. When set, capture will occur on negative transition of T2EX if EXEN2=1. When cleared, auto-reload will occur either with Timer 2 overflows, or negative transitions of T2EX when EXEN2=1. When either (RCLK, RCLK1)=1 or (TCLK, TCLK1)=1, this bit is ignored, and timer is forced to auto-reload on Timer 2 overflow

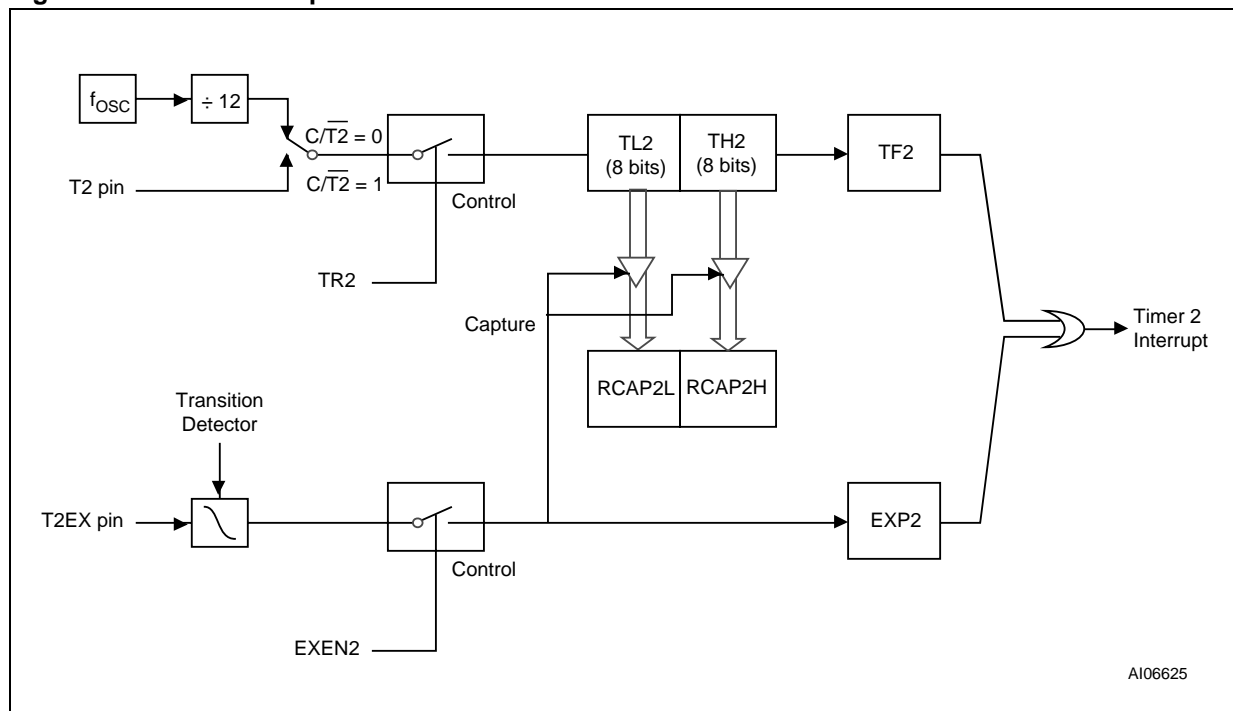
Note: 1. The RCLK1 and TCLK1 Bits in the PCON Register control UART1, and have the same function as RCLK and TCLK.

**Table 39. Timer/Counter 2 Operating Modes**

Mode	T2CON			T2CON EXEN	P1.1 T2EX	Remarks	Input Clock	
	RxCLK or TxCLK	CP/RL2	TR2				Internal	External (P1.0/T2)
16-bit Auto-reload	0	0	1	0	x	reload upon overflow	f <sub>osc</sub> /12	MAX f <sub>osc</sub> /24
	0	0	1	1	↓	reload trigger (falling edge)		
	0	0	1	x	0	Down counting		
	0	0	1	x	1	Up counting		
16-bit Capture	0	1	1	0	x	16-bit Timer/Counter (only up counting)	f <sub>osc</sub> /12	MAX f <sub>osc</sub> /24
	0	1	1	1	↓	Capture (TH2, TL2) → (RCAP2H, RCAP2L)		
Baud Rate Generator	1	x	1	0	x	No overflow interrupt request (TF2)	f <sub>osc</sub> /12	MAX f <sub>osc</sub> /24
	1	x	1	1	↓	Extra External Interrupt (Timer 2)		
Off	x	x	0	x	x	Timer 2 stops	—	—

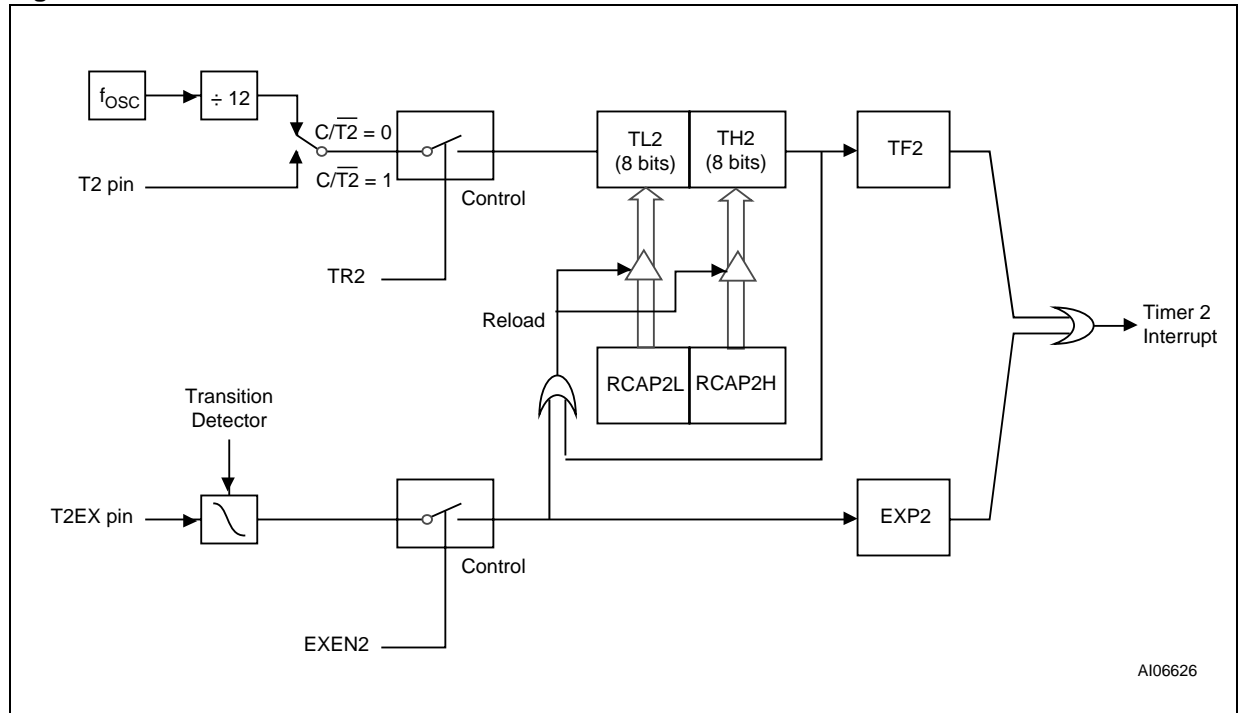
Note: ↓ = falling edge

**Figure 15. Timer 2 in Capture Mode**



AI06625

Figure 16. Timer 2 in Auto-Reload Mode



AI06626



## STANDARD SERIAL INTERFACE (UART)

The μPSD33XX Devices provide two standard 8032 UART serial ports. The first port (UART0) is connected to pin P3.0 (RxD0) and P3.1 (TxD0). The second port (UART1) is connected to pin P1.2 (RxD1) and P1.3(TxD1) or P4.2 and P4.3. The operation of the two serial ports are the same and are controlled by the SCON0 and SCON1 registers.

The serial port is full duplex, meaning it can transmit and receive simultaneously. It is also receive-buffered, meaning it can commence reception of a second byte before a previously received byte has been read from the register. (However, if the first byte still has not been read by the time reception of the second byte is complete, one of the bytes will be lost.) The serial port receive and transmit registers are both accessed at Special Function Register SBUF0 (or SBUF1 for the second serial port). Writing to SBUF loads the transmit register, and reading SBUF accesses a physically separate receive register.

The serial port can operate in 4 modes:

**Mode 0.** Serial data enters and exits through RxD. TxD outputs the shift clock. 8 bits are transmitted/received (LSB first). The baud rate is fixed at 1/12 the  $f_{osc}$ .

**Mode 1.** 10 bits are transmitted (through TxD) or received (through RxD): a Start Bit (0), 8 data bits (LSB first), and a Stop Bit (1). On receive, the Stop Bit goes into RB8 in Special Function Register SCON. The baud rate is variable.

**Mode 2.** 11 bits are transmitted (through TxD) or received (through RxD): Start Bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a Stop Bit (1). On Transmit, the 9th data bit (TB8 in SCON) can be assigned the value of '0' or '1.' Or, for example, the Parity Bit (P, in the PSW) could be moved into TB8. On receive, the 9th data bit goes into RB8 in Special Function Register SCON, while the Stop Bit is ignored. The baud rate is programmable to either 1/32 or 1/64 the oscillator frequency.

**Mode 3.** 11 bits are transmitted (through TxD) or received (through RxD): a Start Bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a Stop Bit (1). In fact, Mode 3 is the same as Mode 2 in all respects except baud rate. The baud rate in Mode 3 is variable.

In all four modes, transmission is initiated by any instruction that uses SBUF as a destination register. Reception is initiated in Mode 0 by the condition  $RI = 0$  and  $REN = 1$ . Reception is initiated in the other modes by the incoming Start Bit if  $REN = 1$ .

### Serial Port Control Register

The serial port control and status register is the Special Function Register SCON0 (SCON1 for the second port), shown in Table 40 and Table 41, page 42. This register contains not only the mode selection bits, but also the 9th data bit for transmit and receive (TB8 and RB8), and the Serial Port Interrupt Bits (TI and RI).

**Table 40. Serial Port Control Register (SCON0 and SCON1)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Note: 1. SCON0 (98H - UART0 Reset Value 00); SCON1 (D8H - UART1 Reset Value 00)

**Table 41. SCON0 and SCON1 Register Details**

Bit	Symbol	Function
7	SM0	See Table 42.
6	SM1	
5	SM2	Enables the multiprocessor communication features in Mode 2 and 3. In Mode 2 or 3, if SM2 is set to '1,' RI will not be activated if its received 8th data bit (RB8) is '0.' In Mode 1, if SM2=1, RI will not be activated if a valid Stop Bit was not received. In Mode 0, SM2 should be '0.'
4	REN	Enables serial reception. Set with software to enable reception. Clear with software to disable reception.
3	TB8	The 8th data bit that will be transmitted in Modes 2 and 3. Set or clear with software as desired
2	RB8	In Modes 2 and 3, this bit contains the 8th data bit that was received. In Mode 1, if SM2=0, RB8 is the Snap Bit that was received. In Mode 0, RB8 is not used.
1	TI	Transmit Interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the Stop Bit in the other modes, in any serial transmission. Must be cleared with software.
0	RI	Receive Interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the Stop Bit in the other modes, in any serial reception (except for SM2). Must be cleared with software.

**Table 42. UART Operating Table**

Mode	SCON		Baud Rate	Description
	SM0	SM1		
0	0	0	$f_{osc}/12$	Serial data enters and exits through RxD. TxD outputs the shift clock. 8-bit are transmitted/received (LSB first)
1	0	1	Timer 1/2 overflow rate	8-bit UART 10 bits are transmitted (through TxD) or received (RxD).
2	1	0	$f_{osc}/32$ or $f_{osc}/64$	9-bit UART 11 bits are transmitted (through TxD) or received (RxD)
3	1	1	Timer 1/2 overflow rate	9-bit UART Like Mode 2, except the variable baud rate.

**Baud Rates.** The baud rate in Mode 0 is fixed:

$$\text{Mode 0 Baud Rate} = f_{\text{OSC}} / 12$$

The baud rate in Mode 2 depends on the value of Bit SMOD = 0 (which is the value on reset), the baud rate is 1/64 the oscillator frequency. If SMOD = 1, the baud rate is 1/32 the oscillator frequency.

$$\text{Mode 2 Baud Rate} = (2^{\text{SMOD}} / 64) \times f_{\text{OSC}}$$

In the μPSD33XX Devices, the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate.

**Using Timer 1 to Generate Baud Rates.** When Timer 1 is used as the baud rate generator, the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate and the value of SMOD as follows:

$$\text{Mode 1,3 Baud Rate} = (2^{\text{SMOD}} / 32) \times (\text{Timer 1 overflow rate})$$

The Timer 1 Interrupt should be disabled in this application. The Timer itself can be configured for either “timer” or “counter” operation, and in any of its 3 running modes. In the most typical applications, it is configured for “timer” operation, in the Auto-reload Mode (high nibble of TMOD = 0010B). In that case the baud rate is given by the formula:

$$\text{Mode 1,3 Baud Rate} = (2^{\text{SMOD}} / 32) \times (f_{\text{OSC}} / (12 \times [256 - (\text{TH1})]))$$

One can achieve very low baud rates with Timer 1 by leaving the Timer 1 Interrupt enabled, and configuring the Timer to run as a 16-bit timer (high nibble of TMOD = 0001B), and using the Timer 1 Interrupt to do a 16-bit software reload. Figure 12, page 35 lists various commonly used baud rates and how they can be obtained from Timer 1.

**Using Timer/Counter 2 to Generate Baud Rates.** In the μPSD33XX Devices, Timer 2 selected as the baud rate generator by setting TCLK and/or RCLK (see Figure 12, page 35, Timer/Counter 2 Control Register).

**Note:** The baud rate for transmit and receive can be simultaneously different. Setting RCLK and/or TCLK puts Timer into its Baud Rate Generator Mode.

The RCLK and TCLK Bits in the T2CON register configure UART0. The RCLK1 and TCLK1 Bits in the PCON register configure UART1.

The Baud Rate Generator Mode is similar to the Auto-reload Mode, in that a roll over in TH2 causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2H and RCAP2L, which are preset with software.

Now, the baud rates in Modes 1 and 3 are determined at Timer 2’s overflow rate as follows:

$$\text{Mode 1,3 Baud Rate} = \text{Timer 2 Overflow Rate} / 16$$

The timer can be configured for either “timer” or “counter” operation. In the most typical applications, it is configured for “timer” operation (C/T2 = 0). “Timer” operation is a little different for Timer 2 when it’s being used as a baud rate generator. In this case, the baud rate is given by the formula:

$$\text{Mode 1,3 Baud Rate} = f_{\text{OSC}} / (32 \times [65536 - (\text{RCAP2H}, \text{RCAP2L})])$$

where (RCAP2H, RCAP2L) is the content of RC2H and RC2L taken as a 16-bit unsigned integer.

Timer 2 also may be used as the Baud Rate Generating Mode. This mode is valid only if RCLK + TCLK = 1 in T2CON or in PCON.

**Note:** A roll-over in TH2 does not set TF2, and will not generate an interrupt. Therefore, the Timer Interrupt does not have to be disabled when Timer 2 is in the Baud Rate Generator Mode.

**Note:** If EXEN2 is set, a 1-to-0 transition in T2EX will set EXF2 but will not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Thus when Timer 2 is in use as a baud rate generator, T2EX can be used as an extra external interrupt, if desired.

**Note:** When Timer 2 is running (TR2 = 1) in “timer” function in the Baud Rate Generator Mode, one should not try to READ or WRITE TH2 or TL2. Under these conditions the timer is being incremented every state time, and the results of a READ or WRITE may not be accurate.

The RC registers may be read, but should not be written to, because a WRITE might overlap a reload and cause WRITE and/or reload errors.

Turn the timer off (clear TR2) before accessing the Timer 2 or RC registers, in this case.

**IRDA INTERFACE TO INFRARED TRANSCIEVER**

The μPSD33XX provides an IrDA interface that meets the IrDA Specification. The IrDA Interface logic “pulse shaping” the UART1 (2nd UART) serial signal from a UART Frame to an IrDA standard IR Frame (and vice versa) that can be accepted by a standard IrDA Transceiver. When enabled and in transmitting mode, the IrDA Interface shortens the UART output signal to IrDA compatible electrical pulses. In receiving mode, the Interface stretches the IrDA transceiver signal to the proper bit rate to be received by the UART. The outputs of the IrDA Interface drive the IrDA Transceiver directly.

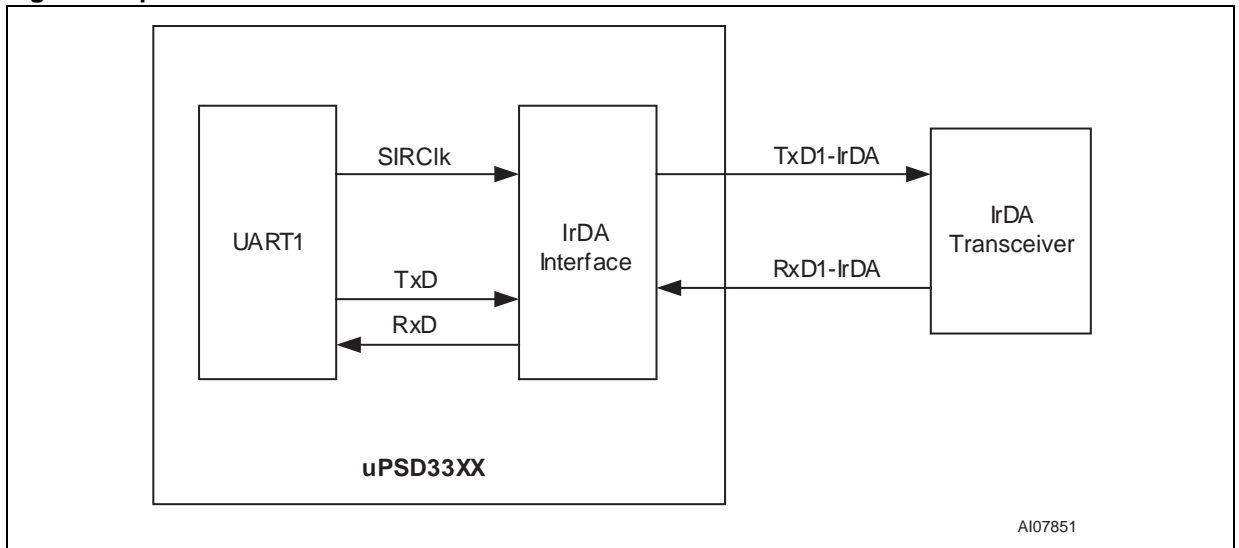
The UART1 can operate in 4 modes, Mode 0 to Mode 3. The IrDA Interface supports Mode 1 (10 bits transmit - Start Bit, 8 data bits and 1 Stop Bit) only so as to be compatible with the IrDA format. The IrDA Interface supports baud rate generated by Timer 1 or Timer 2, but the Tx and Rx must be of the same baud rate.

The features of the IrDA Interface are:

- Stretches the UART pulse to 1.627μs; it supports IrDA pulse from 1.41μs (Min) to 2,23μs (Max) pulse or 3/16 bit pulse duration.
- Support for baud rates from 1.2kHz to 115.2kHz
- Direct interface to SIR transceiver from UART1 I/O pins (RxD1, TxD1) on Ports 1 or 4
- IRDAICON Register bits select pulse duration and specify baud rate

The IrDA Interface is disabled on power-up and is enabled by the IRDAEN Bit in the IRDAICON Register (see Table 43 and Table 44, page 44). When it is disabled, the UART1's RxD and TxD bypass the IrDA Interface and are connected directly to the port pins. The IrDA Interface generates a 1.627μs or 3/16 bit pulse width output when the bit is a '0' on the TxD line and stays '0' when the bit is a '1.'

**Figure 17. μPSD33XX IrDA Interface**



**Table 43. IRDAICON Register Bit Definition (CEH, Reset Value 0FH)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	IRDAEN	PULSE	CDIV4	CDIV3	CDIV2	CDIV1	CDIV0

**Table 44. IRDAICON Register Details**

BIT	SYMBOL	RW	Definition
7	–		Reserved
6	IRDAEN	RW	IrDA Enable 0 = IrDA Interface is disabled 1 = IrDA is enabled, UART1 outputs are disconnected from Port 1 (or port 4)

BIT	SYMBOL	RW	Definition
5	PULSE	RW	IrDA Pulse Modulation Select 0 = 1.627μs 1 = 3/16 bit time pulses
4-0	CDIV[4:0]	RW	Specify Clock Divider (see Table 45)

**Baud Rate Select**

There will be two schemes for IrDA pulse modulation:

- In the event of 3/16 bit time pulse modulation:  
At maximum baud rate of 115.2kHz, the 3/16 bit time pulse is 1.627μs.
- In case of 1.627μs pulse modulation:  
To implement 1.627μs pulse modulation, a prescaler is needed to generate a subreference

clock (i.e, SIRClk) that is used to generate 1.627μs pulse modulation. Even though the pulse width is 1.627μs, the baud rate follows the configuration of UART1. Select a clock divider to generate a  $F_{SIRCLK}$  clock close to 1.8432MHz.

$$F_{SIRCLK} = f_{osc} / (CDIV[4:0])$$

where CDIV[4:0] must be 4 or larger.

**Table 45.  $f_{SIRCLK}$  Frequency**

$f_{osc}$	CDIV[4:0] (Clock divider)	$f_{SIRCLK}^{(1)}$
40.0MHz	22	1.8181MHz
33.0MHz	18	1.8333MHz
30.0MHz	16	1.8750MHz
24.0MHz	13	1.8461MHz
16.0MHz	9	1.7777MHz
12.0MHz	7	1.7142MHz

Note: 1.  $f_{SIRCLK}$  at 1.8342MHz is needed to generate the 1.627μs pulse.

**I<sup>2</sup>C INTERFACE**

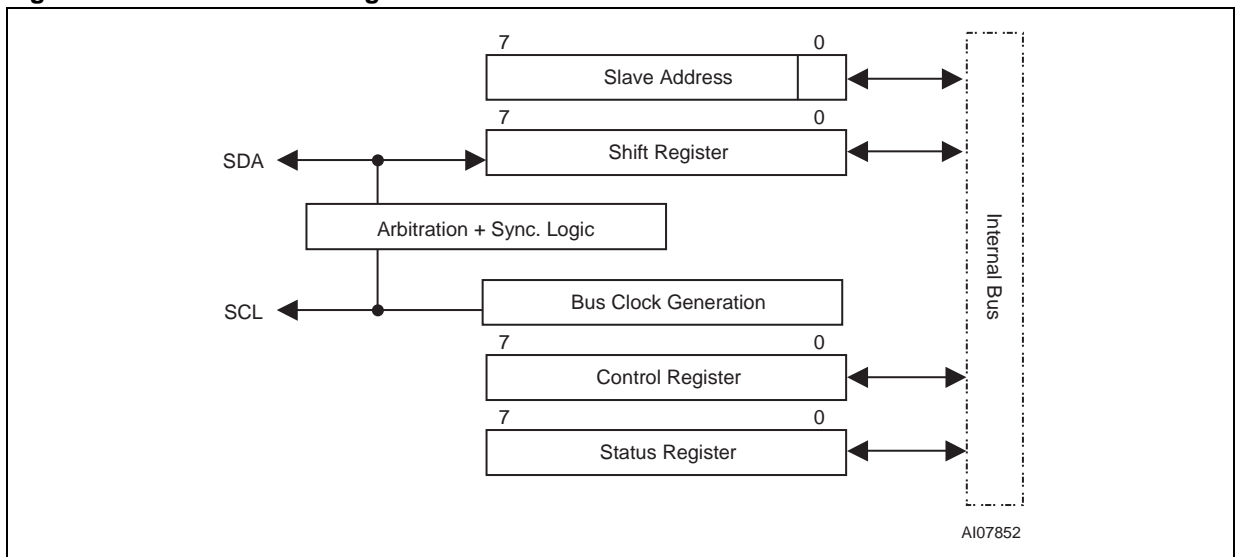
There is a serial I<sup>2</sup>C port implemented in the μPSD33XX Devices. The serial port supports the twin line I<sup>2</sup>C -bus and consists of a data line (SDA) and a clock line (SCL). Depending on the configuration, the SDA and SCL lines may require pull-up resistors.

The system is unique because data transport, clock generation, address recognition, and bus control arbitration are all controlled by hardware. The I<sup>2</sup>C serial I/O has complete autonomy in byte handling and operates in 4 modes:

- Master transmitter
- Master receiver
- Slave transmitter
- Slave receiver

These functions are controlled by the I<sup>2</sup>C SFRs:  
 S1CON: the Control register - control of byte handling and the operation of 4 mode  
 S1STA: the Status register - contents of its register may also be used as a vector to various service routines.  
 S1DAT: Data Shift register.  
 S1ADR: Slave Address register. Slave address recognition is performed by On-Chip Hardware.  
 Table 48, page 47 shows the divisor values and the I<sup>2</sup>C bit rate for some common f<sub>OSC</sub> frequencies.

**Figure 18. I<sup>2</sup>C Bus Block Diagram**



**I<sup>2</sup>C Registers Definition**

**Table 46. Serial Control Register S1CON (DCH, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CR2	ENI1	STA	STO	ADDR	AA	CR1	CR0

**Table 47. S1CON Register Details**

Bit	Symbol	Function
7	CR2	This bit, along with Bits CR1 and CR0 determines the serial clock frequency when SIO is in the Master Mode.
6	ENI1	Enable IIC. When ENI1 = 0, the IIC is disabled. SDA and SCL outputs are in the high impedance state.
5	STA	START flag. When this bit is set, the SIO H/W checks the status of the I <sup>2</sup> C bus and generates a START condition if the bus is free. If the bus is busy, the SIO will generate a repeated START condition when this bit is set. When a START condition is detected on the I <sup>2</sup> C Bus, the I <sup>2</sup> C hardware clears the STA flag. <b>Note:</b> If this bit is set during an interrupt service, the START condition occurs after the interrupt service.
4	STO	STOP flag. With this bit set while in Master Mode a STOP condition is generated. When a STOP condition is detected on the I <sup>2</sup> C bus, the I <sup>2</sup> C hardware clears the STO flag. <b>Note:</b> If this bit is set during an interrupt service, the STOP condition occurs after the interrupt service.
3	ADDR	This bit is set when address byte was received. Must be cleared with software.
2	AA	Acknowledge enable signal. If this bit is set, an acknowledge (low level to SDA is returned during the acknowledge clock pulse on the SCL line when: <ul style="list-style-type: none"> <li>■ Own slave address is received;</li> <li>■ A data byte is received while the device is programmed to be a Master Receiver;</li> <li>■ A data byte is received while the device is a selected Slave Receiver; and</li> <li>■ When this bit is reset, no acknowledge is returned.</li> </ul> SIO release SDA line as high during the acknowledge clock pulse.
1, 0	CR1, CR0	These two bits, along with the CR2 Bit determine the serial clock frequency when SIO is in the Master Mode.

**Table 48. Selection of the Serial Clock Frequency SCL in Master Mode**

CR2	CR1	CR0	fosc Divisor	Bit Rate (kHz) @ fosc			
				12MHz	24MHz	36MHz	40MHz
0	0	0	16	375	750	χ <sup>(1)</sup>	χ <sup>(1)</sup>
0	0	1	24	250	500	750	833
0	1	0	30	200	400	600	666
0	1	1	60	100	200	300	333
1	0	0	120	50	100	150	166
1	0	1	240	25	50	75	83
1	1	0	480	12.5	25	37.5	41
1	1	1	960	6.25	12.5	18.75	20

Note: 1. These values are beyond the supported bit rate.

**Serial Status Register (S1STA)**

S1STA is a “Read only” register (except Bit 5 INTR, see Table 49). The contents of this register may be used as a vector to a service routine. This optimized the response time of the software and consequently that of the I<sup>2</sup>C-bus.

The status codes for all possible modes of the I<sup>2</sup>C bus interface are given Table 50.

This flag is set, and an interrupt is generated after any of the following events occur.

1. Own slave address has been received during AA = 1: ack\_int
2. The general call address has been received while GC(S1ADR.0) = 1 and AA = 1

3. A data byte has been received or transmitted in Master Mode (even if arbitration is lost): ack\_int
4. A data byte has been received or transmitted as selected slave: ack\_int
5. A Stop condition is received as selected slave receiver or transmitter: stop\_int

**Data Shift Register (S1DAT)**

S1DAT contains the serial data to be transmitted or data which has just been received (see Table 51). The MSB (Bit 7) is transmitted or received first; that is, data shifted from right to left.

**Table 49. Serial Status Register S1STA (DDH, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
GC	STOP	INTR	TX_MODE	BBUSY	BLOST	$\overline{\text{ACK\_REP}}$	SLV

**Table 50. S1STA Status Register Details**

Bit	Symbol	Function
7	GC	General Call flag
6	STOP	STOP flag. This bit is set when a STOP condition is received.
5	INTR	Interrupt flag. This bit is set when a I <sup>2</sup> C interrupt is requested. Must be cleared with software.
4	TX_MODE	Transmission Mode flag. This bit is set when the I <sup>2</sup> C is a transmitter. Otherwise, this bit is reset.
3	BBUSY	Bus Busy State flag. This bit is set when the bus is being used by another master. Otherwise, this bit is reset.
2	BLOST	Bus Lost flag. This bit is set when the master loses the bus contention. Otherwise, this bit is reset.
1	$\overline{\text{ACK\_REP}}$	Acknowledge response flag. This bit is set when the receiver transmits the not acknowledge signal. This bit is reset when the receiver transmits the acknowledge signal. Even if this bit is set, the STOP condition does not occur in the bus. (MASTER MODE)
0	SLV	Slave Mode flag. This bit is set when the I <sup>2</sup> C plays role in the slave mode. Otherwise, this bit is reset.

**Table 51. Data Shift Register S1DAT (DEH, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
S1DAT7	S1DAT6	S1DAT5	S1DAT4	S1DAT3	S1DAT2	S1DAT1	S1DAT0



### Address Register (S1ADR)

This 8-bit register may be loaded with the 7-bit slave address to which the controller will respond when programmed as a slave receive/transmitter (see Table 52). The Start/Stop Hold Time Detection and System Clock registers (Table 53 and Ta-

ble 54) are included in the I<sup>2</sup>C unit to specify the start/stop detection time to work with the large range of MCU frequency values supported. Table 55 is an example with a system clock of 40MHz.

**Table 52. Address Register S1ADR (DFH, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SLA6	SLA5	SLA4	SLA3	SLA2	SLA1	SLA0	–

**Table 53. Start/Stop Hold Time Detection Register S1SETUP (DBH, Reset Value 00H)**

Bit 7	Bit 6 - Bit 0
Enable S1SETU	Bit 6 - 0 specify the number of sample clocks

**Table 54. System Clock of 40MHz**

S1SETUP Register Value	No. of Sample Clock (f <sub>osc</sub> = 25ns)	Required Start/Stop Hold Time	Note
00H	1EA	25ns	When Bit 7 (Enable bit) = 0, the number of sample clock is 1EA (ignore Bit 6 - Bit 0)
80h	1EA	25ns	
81h	2EA	50ns	
82h	3EA	75ns	
...	...		
8Bh	12EA	300ns	
97h	24EA	600ns	Fast Mode I <sup>2</sup> C Start/Stop Hold time specification
...	...		
FFh	128EA	3000ns	

**Table 55. System Clock Setup Examples**

System Clock	S1SETUP Register Value	No. of Sample Clock	Required Start/Stop Hold Time
40MHz (f <sub>osc</sub> -> 25ns)	97h	24EA	600ns
30MHz (f <sub>osc</sub> -> 33.3ns)	91h	18EA	599ns
20MHz (f <sub>osc</sub> -> 50ns)	8Bh	12EA	600ns
8MHz (f <sub>osc</sub> -> 125ns)	84h	5EA	625ns

**SPI (SYNCHRONOUS PERIPHERAL INTERFACE)**

The SPI is a master interface that enables synchronous, serial communication with external slave peripherals. The SPI features full-duplex, three-wire synchronous transfers and programmable clock polarity (optional 4 wires). The SPI performs parallel-to-serial conversion on data written to a 8-bit wide Transmit data register (SPITDR) and serial-to-parallel conversion on received data, buffering a 8-bit wide Receive data register (SPIRDR).

The SPI supports a subset of the SPI function, mainly the Master Mode with CPHA=1 Transfer Format. It will be able to interface a device that has a SPI Slave interface with the slave select being grounded or controlled by the SPI. The CPHA=1 Transfer Format requires that the first data bit is shifted out at the same time as the first SPICLK.

The SPI has the following features:

1. Support Master Mode, 8 bit data size
2. Programmable Clock Polarity

3. 8-bit wide, double-buffered transmit and receive operation
4. Full-duplex - Both transmit and receive operate simultaneously with two wires
5. 3, or 4 wires external pins (see Figure 19):

**SPITxD** – This pin is used to transmit data out of the SPI module.

**SPiRxD** – This pin is used to receive data from slave mode.

**SPiSEL** – This pin is used to output the select signal from the SPI module to another peripheral with which a data transfer is to take place.

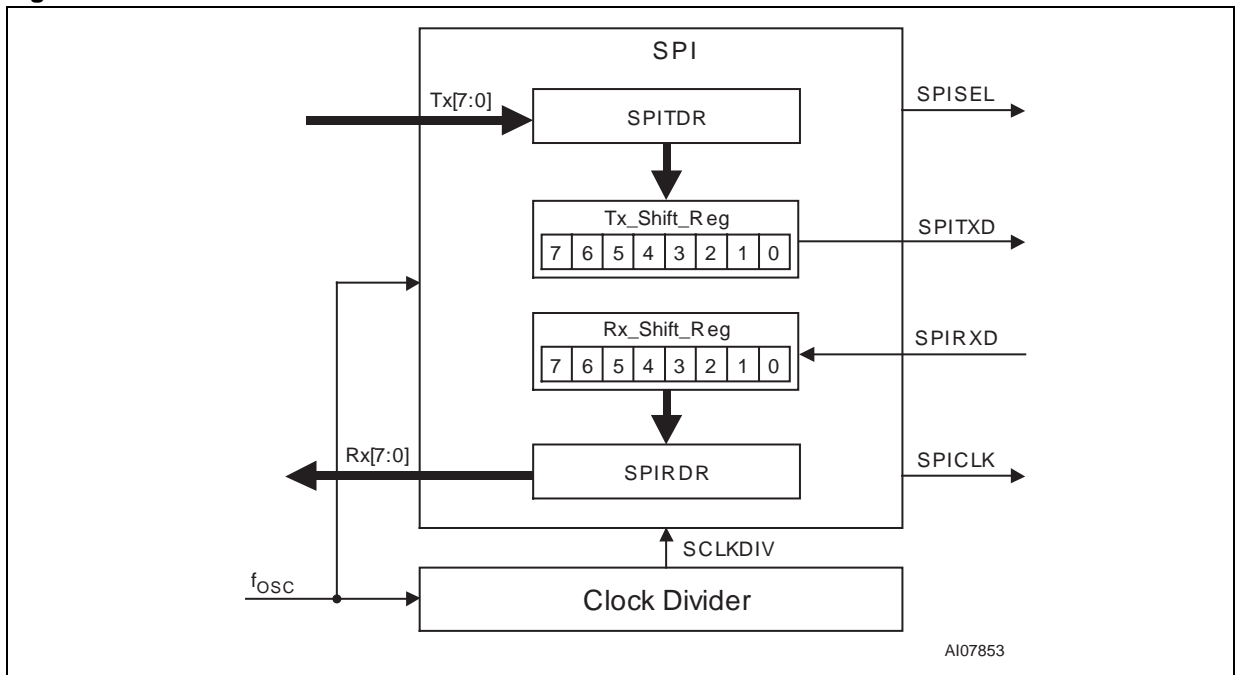
**SPiCLK** – This pin is used to output the SPiCLK clock

6. Programmable baud rate which can be modulated by SPiCLKD register

**SPI Registers**

The SPI has seven registers for data transmit, receive, and control (see Table 56, page 51 through Table 60, page 52).

**Figure 19. SPI Bus Interface**



AI07853

**Table 56. SPI Registers**

Register	SFR Offset	Dir.	Description	Reset Value
SPICON0	D6H	RW	Control Register 0	00
SPICON1	D7H	RW	Control Register 1	00
SPITDR	D4H	W	Transmit Data Register (data byte to be transmitted)	00
SPIRD	D5H	R	Receive Data Register (store received data byte)	00
SPICLKD	D2H	RW	Clock Divider Value	04
SPISTAT	D3H	R	Status Register	02

**Table 57. SPICON0 (Control Register 0) Details (D6H, Reset Value 00H)**

BIT	SYMBOL	RW	Definition
7	–		Reserved
6	TE	RW	Transmitter Enable 0 = Transmitter is disabled 1 = Transmitter is enabled
5	RE	RW	Receiver Enable 0 = Receiver is disabled 1 = Receiver is enabled
4	SPIEN	RW	SPI Enable 0 = SPI is disabled 1 = SPI is enabled
3	SSEL	RW	Slave Selection 0 = Slave Select output is disabled 1 = Slave Select output is enabled on Port pin P1.7 (or P4.7)
2	FLSB	RW	First LSB 0 = Transfer the most significant bit (MSB) first 1 = Transfer the least significant bit (LSB) first
1	SPO	RW	Sampling Polarity 0 = Sample transfer data at the falling edge of clock (SPICLK is '0' when idle) 1 = Sample transfer data at the rising edge of clock (SPICLK is '1' when idle)
0	–		Reserved

**Table 58. SPICON1 (Control Register 1) Details (D7H, Reset Value 00H)**

BIT	SYMBOL	RW	Definition
7-4	–		Reserved
3	TEIE	RW	Transmission End Interrupt Enable 0 = SPI Transmission end Interrupt Disable 1 = SPI Transmission end Interrupt Enable
2	RORIE	RW	Receive Overrun Interrupt Enable 0 = Receive Overrun Interrupt Disable 1 = Receive Overrun Interrupt Enable
1	TIE	RW	Transmission Interrupt Enable 0 = SPITDR empty interrupt Disable 1 = SPITDR Empty interrupt Enable
0	RIE	RW	Reception Interrupt Enable 0 = SPIRDR full interrupt Disable 1 = SPIRDR full interrupt Enable

**Table 59. SPICLKD (SPI Prescaler) Register (D2H, Reset Value 04H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DIV128	DIV64	DIV32	DIV16	DIV8	DIV4	–	–

**Table 60. SPICLKD (SPI Prescaler) Details**

BIT	SYMBOL	RW	Definition
7	DIV128	RW	0 = No division 1 = Divide $f_{OSC}$ clock by 128
6	DIV64	RW	0 = No division 1 = Divide $f_{OSC}$ clock by 64
5	DIV32	RW	0 = No division 1 = Divide $f_{OSC}$ clock by 32
4	DIV16	RW	0 = No division 1 = Divide $f_{OSC}$ clock by 16
3	DIV8	RW	0 = No division 1 = Divide $f_{OSC}$ clock by 8
2	DIV4	RW	0 = No division 1 = Divide $f_{OSC}$ clock by 4
1-0	Not Used		

The SPI serial clock frequency in Master Mode is the  $f_{OSC}$  clock divided by the SPICLKD divisors. The bits in the SPICLKD register can be set to provide divisor values of multiple of 4: 4, 8, 12, 16, 20... to 252.

**Operation**

The SPI transmitter and receiver share the same clock but are independent, so full-duplex communication is possible. The transmitter and receiver are also double-buffered, so continuous transmitting or receiving (back-to-back transfer) is possible by reading or writing data while transmitting or receiving is in progress.

**SPI Configuration**

The SPI is reset by the CPU Reset. Control register SPICON0 needs to be programmed to decide several operation parameters. The SPO Bit determines clock polarity. When SPO is set to '0,' the data bit is placed on the communication line from one rising edge of serial clock to the next and is

guaranteed valid at the fall of serial clock. When SPO is set to '1,' the data bit is placed on the communication line from one falling edge of serial clock to the next and is guaranteed valid at the rise of serial clock.

The FLFSB Bit determines the format of 8-bit serial data transfer. When FLFSB is '0,' the 8-bit data is transferred in order from MSB (first) to LSB (last). When FLFSB is '1,' the data is transferred in order from LSB (first) to MSB (last).

The bit rate requires the programming of the clock divider register SPICLKD. The value of SPICLKD divides the  $f_{OSC}$  clock to provide the serial transfer clock output -SPICLK.

The SPICON1 and SPICON0 have SPI Transmitter Enable (TE) and Receiver Enable (RE) and Interrupt Enable Bits (TEIE, RORIE, TIE, RIE). If TE is disabled, both transmitting and receiving are disabled because SPICLK is forced to LOW (SPO=0) or HIGH (SPO=1).

**Table 61. SPISTAT (Status) Register (D3H, Reset Value 02H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
–	–	–	BUSY	TEISF	RORISF	TISF	RISF

**Table 62. SPISTAT (Status) Register Details**

BIT	SYMBOL	RW	Definition
7-5	Reserved		
4	BUSY	R	SPI Busy 0 = Tx/Rx is completed 1 = Tx/Rx is on going
3	TEISF	R	Transmission End Interrupt Source flag 0 = Reset when users read this register 1 = Set when transmission end occurs
2	RORISF	R	Receive Overrun Interrupt Source flag 0 = Reset when user reads this register 1 = Set when Rx Overrun occurs
1	TISF	R	Transfer Interrupt Source flag 0 = Reset when SPITDR is full (when the SPITDR is written) 1 = Set when SPITDR is empty
0	RISF	R	Receive Interrupt Source flag 0 = Reset when SPIRDR is empty (when the SPITDR is read) 1 = Set when SPIRDR is full

### Slave Select Output

The SPI can be operated as a SPI bus in Master Mode. The Slave Select (SPISEL) line in the SPI bus is assigned to port pin P1.7 (or P4.7). When the SSEL Bit is set in the Control Register, the SPI drives the SPISEL line low to select the slave device before data transmission. The rising edge of SPISEL occurs after the last bit is shifted out.

**Transmit operation.** In transmitting serial data, the SPI operates as follows:

1. The initial sequence would be:
  - CPU writes the byte to SPITDR,
  - CPU sets SPIEN = 1, TIE = 1,
  - CPU sets TE = 1 to enable transmit,
  - SPI loads TSR with data from TDR, and
  - SPI sets TISF and interrupts the CPU to write the second byte.
2. In the ISR (Interrupt Service Routine) for SPI, the CPU writes new data on SPITDR. This update will automatically clear TISF.
3. If TISF is cleared (i.e., SPITDR has a valid data) and the TSR (Transmit Shift Register) is ready

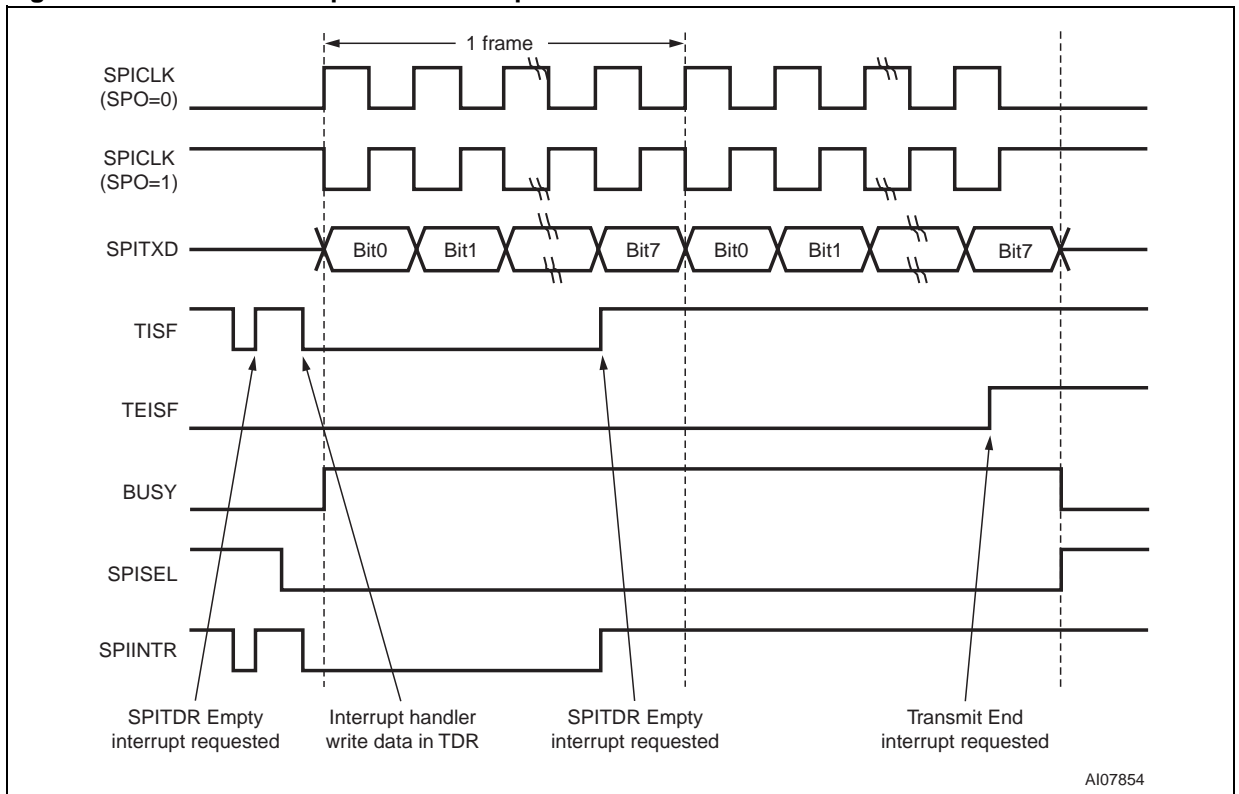
to load new data (e.g., the last bit (8th bit) of the TSR is being sent, or the TSR is empty), the SPI will load the TSR with data on SPITDR and set TISF to '1' (i.e., request CPU to fill SPITDR).

4. The SPI checks the TISF flag when it outputs the last bit (8th bit) of the eight-bit serial transmission data.
  - If the TISF flag is '0,' the SPI loads data from SPITDR into the TSR and begins serial transmission of the next 8-bit frame (continuous transfer).
  - If the TISF is '1,' the SPI sets the TEISF flag to '1' in SPISTAT, and if the TEIE Bit is set to '1' in SPICON1, a Transmit End Interrupt is requested at this time.

So, the TISF Bit must be '0' before the last bit is transmitted to perform continuous transfer. After transmitting the last bit, the SPI holds the SPITxD pin in the last bit state.

5. After the end of serial transmission, the SPICLK pin is held in a constant state.

**Figure 20. SPI Transmit Operation Example**



### Receive Operation

In receiving serial data, the SPI operates as follows:

1. The SPI generates serial clock and synchronizes internally.
2. Received data is stored in the RSR (Receive Shift Register) in order from MSB to LSB (FLSB = 0) or from LSB to MSB (FLSB = 1). After receiving the data, the SPI checks to see if the RIS flag is '0' or not.

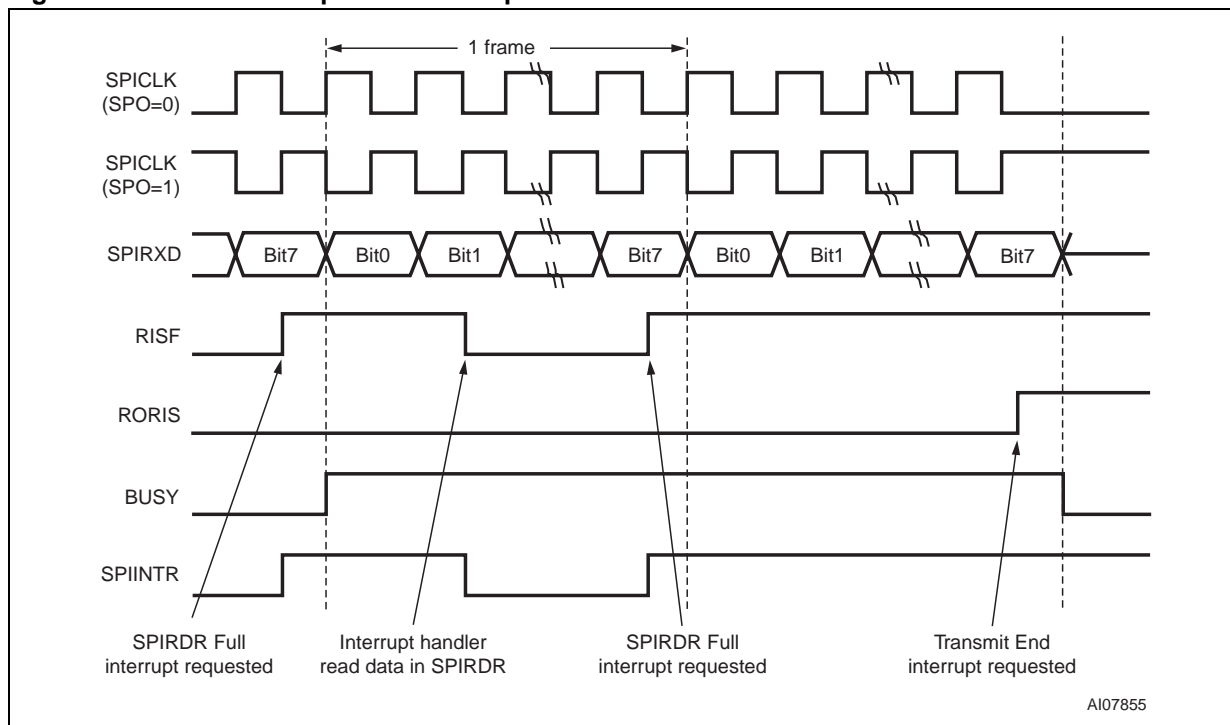
If this check passes, the received data in the RSR is stored in SPIRDR and the RIS flag is set to 1.

When the check fails (i.e., the RIS flag is '1' or the last received data in SPIRDR is not read until the 8th bit of currently received data is received in the RSR), the RORIS flag is set to '1' and received data in the RSR is lost. When the RORIS flag is set to '1' and the RORIE Bit is set to '1' at SPICON1, the subsequent transmit and receive operations are disabled.

3. If the RIE Bit in SPICON1 is set to '1' and the RIS flag is set to '1,' the SPIRDR Full Interrupt is requested.

If the RORIE Bit in SPICON1 is set to '1' and the RORIS flag is set to '1,' the Receive Overrun Interrupt is requested.

**Figure 21. SPI Receive Operation Example**



**ANALOG-TO-DIGITAL CONVERTOR (ADC)**

The ADC unit in the μPSD33XX is a SAR type ADC with a SAR register, an auto-zero comparator and three internal DACs. The unit has 8 input channels with 10-bit resolution. The A/D converter has its own V<sub>REF</sub> input (80-pin package only), which specifies the voltage reference for the A/D operations. The analog to digital converter (A/D) allows conversion of an analog input to a corresponding 10-bit digital value. The A/D module has eight analog inputs (P1.0 through P1.7) to an 8x1 multiplexor. One ADC channel is selected by the bits in the configuration register. The converter generates a 10-bits result via successive approximation. The analog supply voltage is connected to the V<sub>REF</sub> input, which powers the resistance ladder in the A/D module.

The A/D module has 3 registers, the control register ACON, the A/D result register ADAT0, and the second A/D result register ADAT1. The ADAT0 register stores Bits 0.. 7 of the converter output, Bits 8.. 9 are stored in Bits 0..1 of the ADAT1 register. The ACON register controls the operation of the A/D converter module. Three of the bits in the ACON register select the analog channel inputs, and the remaining bits control the converter operation.

ADC channel pin input is enabled by setting the corresponding bit in the P1SFS0 and P1SFS1 registers to '1' and the channel select bits in the ACON register.

The ADC reference clock (ADCCLK) is generated from f<sub>OSC</sub> divided by the divider in the ADCPS reg-

ister. The ADC operates within a range of 2 to 16MHz, with typical ADCCLK frequency at 8MHz.

The conversion time is 4μs typical at 8MHz.

The processing of conversion starts when the Start Bit ADST is set to '1.' After one cycle, it is cleared by hardware. The ADC is monotonic with no missing codes. Measurement is by continuous conversion of the analog input. The ADAT register contains the results of the A/D conversion. When conversion is complete, the result is loaded into the ADAT. The A/D Conversion Status Bit ADSF is set to '1.' The block diagram of the A/D module is shown in Figure 22. The A/D status bit ADSF is set automatically when A/D conversion is completed and cleared when A/D conversion is in process.

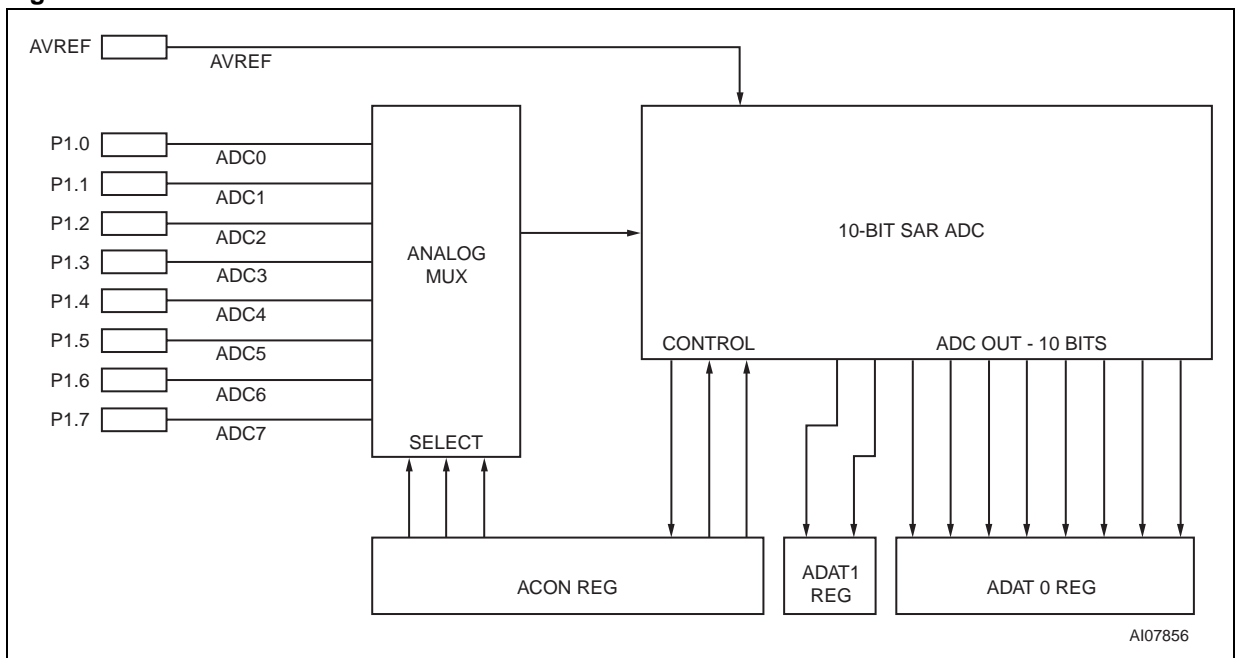
In addition, the ADC unit sets the interrupt flag in the ACON register after a conversion is complete (if AINTEN is set to '1'). The ADC interrupts the CPU when the enable bit AINTEN is set.

**Port 1 ADC Channel Selects**

The P1SFS0 and P1SFS1 Registers control the selection of the Port 1 pin functions. When the P1SFS0 Bit is '0,' the pin functions as a GPIO. When bits are set to '1,' the pins are configured as alternate functions. A new P1SFS1 Register selects which of the alternate functions is enabled. The ADC channel is enabled when the bit in P1SFS1 is set to '1.'

**Note:** In the 52-pin package, there is no individual V<sub>REF</sub> pin.

**Figure 22. 10-Bit ADC**





**Table 63. ACON Register (97H, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AINTF	AINTEN	ADEN	ADS2	ADS1	ADS0	ADST	ADSF

**Table 64. ACON Register Details**

Bit	Symbol	Function
7	AINTF	ADC Interrupt flag. This bit must be cleared with software. 0 = No interrupt request 1 = The AINTF flag is set when ADSF goes from '0' to '1.' Interrupts CPU when both AINTF and AINTEN are set to '1.'
6	AINTEN	ADC Interrupt Enable 0 = ADC interrupt is disabled 1 = ADC interrupt is enabled
5	ADEN	ADC Enable Bit 0 = ADC shut off and consumes no operating current 1 = Enable ADC. ADC must be enabled before setting the ADST Bit.
4.. 2	ADS2.. 0	Analog channel Select 000 Select channel 0 (P1.0) 001 Select channel 0 (P1.1) 010 Select channel 0 (P1.2) 011 Select channel 0 (P1.3) 101 Select channel 0 (P1.5) 110 Select channel 0 (P1.6) 111 Select channel 0 (P1.7)
1	ADST	ADC Start Bit 0 = Force to zero 1 = Start and ADC, then after one cycle, the bit is cleared to '0.'
0	ADSF	ADC Status Bit 0 = ADC conversion is not completed 1 = ADC conversion is completed. The bit can also be cleared with software.

**Table 65. ADCPS Register Details (94H, Reset Value 00H)**

Bit	Symbol	Function
7:4	–	Reserved
3	ADCCE	ADC Conversion Reference Clock Enable 0 = ADC reference clock is disabled (default) 1 = ADC reference clock is enabled
2:0	ADCPS[2:0]	ADC Reference Clock PreScaler $f_{ACLK} = f_{OSC} / 2^{ADCPS[2:0]}$ Example: for $f_{OSC} = 40\text{MHz}$ and $ADCPS[2:0] = 2$ , the $f_{ACLK} = 40 / 2^2 = 10\text{MHz}$ . $f_{ACLK}$ frequency range must be 2–16MHz.

**Table 66. ADAT0 Register (95H, Reset Value 00H)**

Bit	Symbol	Function
7:0	–	Store ADC output, Bit 7 - 0

**Table 67. ADAT1 Register (96H, Reset Value 00H)**

Bit	Symbol	Function
7:2	–	Reserved
1.. 0	–	Store ADC output, Bit 9, 8

**PROGRAMMABLE COUNTER ARRAY (PCA) WITH PWM**

There are two Programmable Counter Array blocks (PCA0 and PCA1) in the μPSD33XX. A PCA block consists of a 16-bit up-counter, which is shared by three TCM (Timer Counter Module). A TCM can be programmed to perform one of the following four functions:

1. Capture Mode: capture counter values by external input signals
2. Timer Mode
3. Toggle Output Mode
4. PWM Mode: fixed frequency (8-bit or 16-bit), programmable frequency (8-bit only)

**PCA Block**

The 16-bit Up-Counter in the PCA block is a free-running counter (except in PWM Mode with programmable frequency). The Counter has a choice

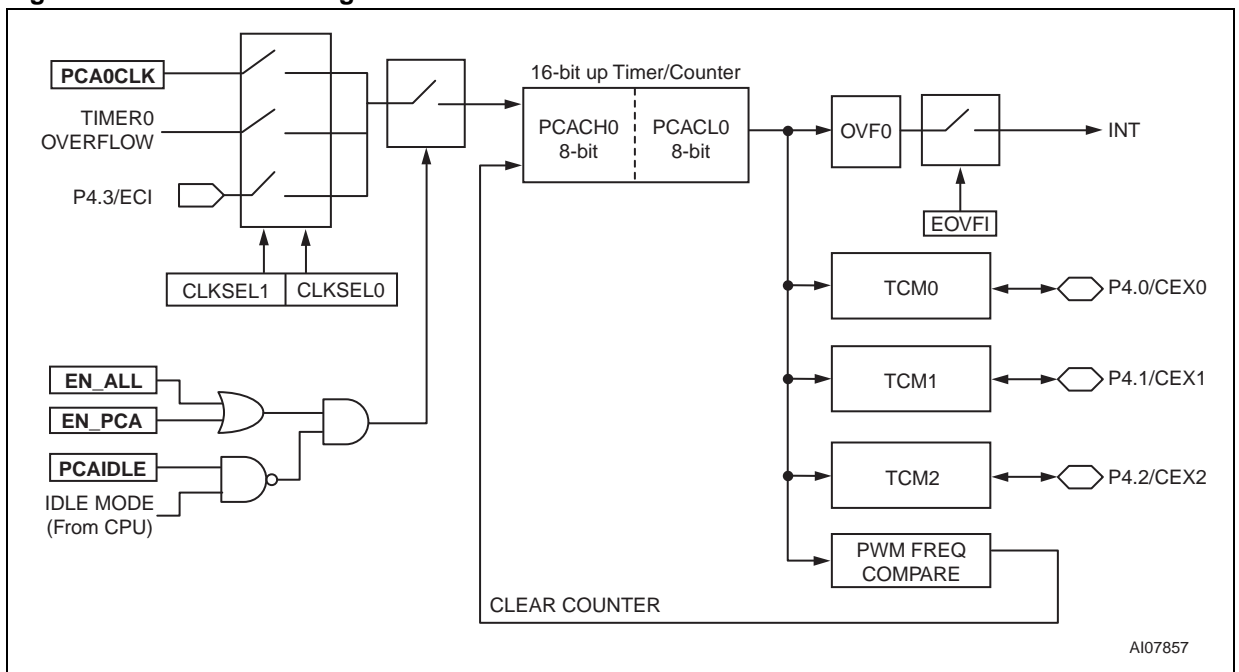
of clock input: from an external pin, Timer 0 Overflow, or PCA Clock.

A PCA block has 3 Timer Counter Modules (TCM) which share the 16-bit Counter output. The TCM can be configured to capture or compare counter value, generate a toggling output, or PWM functions. Except for the PWM function, the other TCM functions can generate an interrupt when an event occurs.

Every TCM is connected to a port pin in Port 4; the TCM pin can be configured as an event input, a PWMs, a Toggle Output, or as External Clock Input. The pins are general I/O pins when not assigned to the TCM.

The TCM operation is configured by Control registers and Capture/Compare registers. Table 68, page 59 lists the SFR registers in the PCA blocks.

**Figure 23. PCA0 Block Diagram**



**Table 68. PCA0 and PCA1 Registers**

SFR Address		Register Name		RW	Register Function
PCA0	PCA1	PCA0	PCA1		
A2		PCACL0	PCACL1	RW	The low 8 bits of PCA 16-bit counter.
A3		PCACH0	PCACH1	RW	The high 8 bits of PCA 16-bit counter.
A4		PCACON0	PCACON1	RW	Control Register <ul style="list-style-type: none"> <li>■ Enable PCA, Timer Overflow flag, PCA Idle Mode, and Select clock source.</li> </ul>
A5		PCASTA	N/A	RW	Status Register, Interrupt Status flags <ul style="list-style-type: none"> <li>■ Common for both PCA Block 0 and 1.</li> </ul>
A9, AA, AB	BD, BE, BF	TCMMODE0 TCMMODE1 TCMMODE2	TCMMODE3 TCMMODE4 TCMMODE5	RW	TCM Mode <ul style="list-style-type: none"> <li>■ Capture, Compare, and Toggle Enable Interrupts</li> <li>■ PWM Mode Select.</li> </ul>
AC AD	C1 C2	CAPCOML0 CAPCOMH0	CAPCOML3 CAPCOMH3	RW	Capture/Compare registers of TCM0
AF B1	C3 C4	CAPCOML1 CAPCOMH1	CAPCOML4 CAPCOMH4	RW	Capture/Compare registers of TCM1
B2 B3	C5 C6	CAPCOML2 CAPCOMH2	CAPCOML5 CAPCOMH5	RW	Capture/Compare registers of TCM2
B4	C7	PWMF0	PWMF1	RW	The 8-bit register to program the PWM frequency. This register is used for programmable, 8-bit PWM Mode only.

**Operation of TCM Modes**

Each of the TCM in a PCA block supports four modes of operation. However, an exception is when the TCM is configured in PWM Mode with programmable frequency. In this mode, all TCM in a PCA block must be configured in the same mode or left to be not used.

**Capture Mode**

The CAPCOM registers in the TCM are loaded with the counter values when an external pin input changes state. The user can configure the counter value to be loaded by positive edge, negative edge or any transition of the input signal. At loading, the TCM can generate an interrupt if it is enabled.

**Timer Mode**

The TCM modules can be configured as software timers by enable the comparator. The user writes a value to the CAPCOM registers, which is then compared with the 16-bit counter. If there is a match, an interrupt can be generated to CPU.

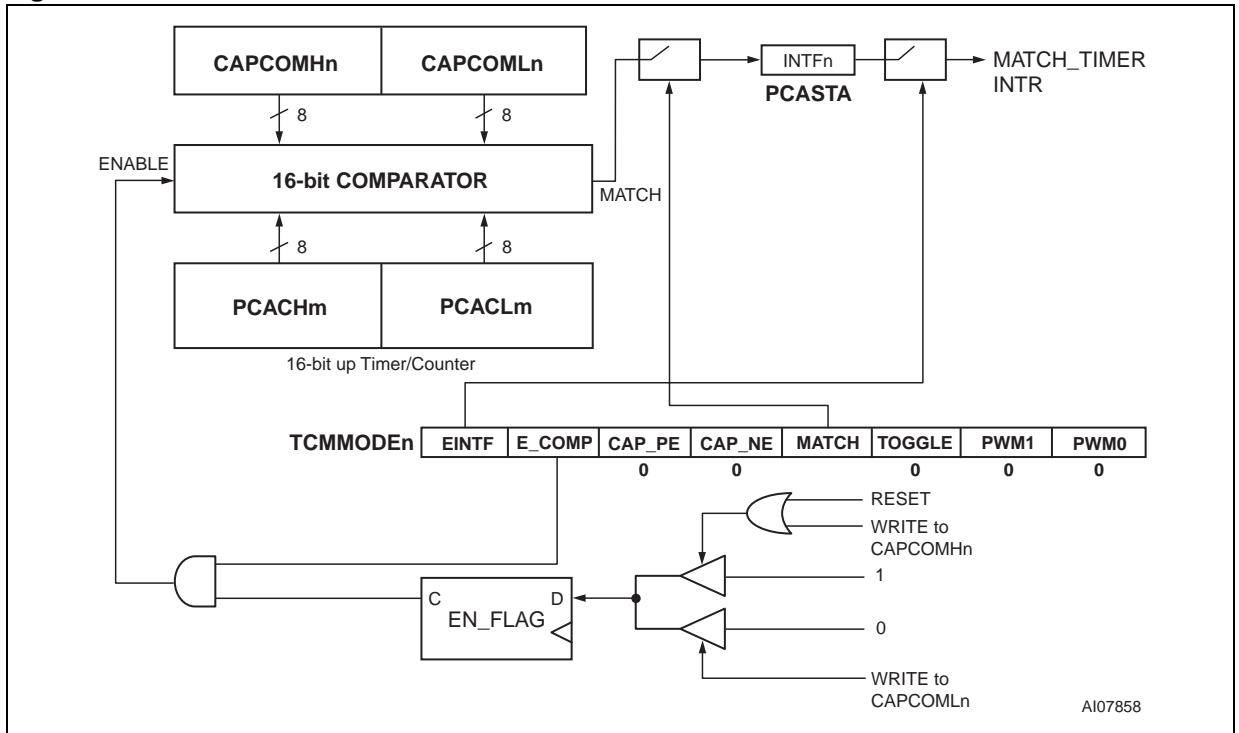
**Toggle Mode**

In this mode, the user writes a value to the TCM's CAPCOM registers and enables the comparator. When there is a match with the Counter output, the output of the TCM pin toggles. This mode is a simple extension of the Timer Mode.

**PWM Mode - (X8), Fixed Frequency**

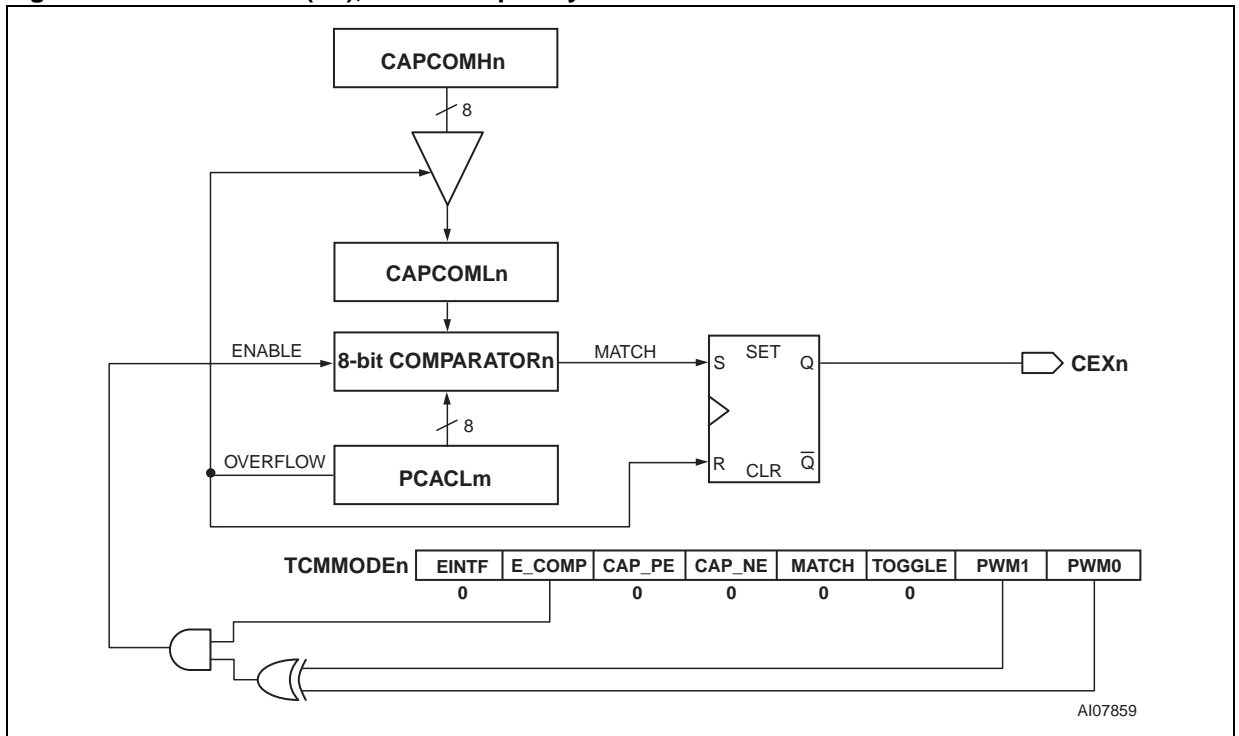
In this mode, one or all the TCM's can be configured to have a fixed frequency PWM output on the port pins. The PWM frequency depends on when the low byte of the Counter overflows (module 256). The duty cycle of each TCM module can be specified in the CAPCOMHn register. When the PCA\_Counter\_L value is equal to or greater than the value in CAPCOMHn, the PWM output is switched to a high state. When the PCA\_Counter\_L Register overflows, the content in CAPCOMHn is loaded to CAPCOMLn and a new PWM pulse starts.

Figure 24. Timer Mode



Note: m = 0: n = 0, 1, or 2  
 m = 1: n = 3, 4, or 5

Figure 25. PWM Mode - (X8), Fixed Frequency



Note: m = 0: n = 0, 1, or 2  
 m = 1: n = 3, 4, or 5

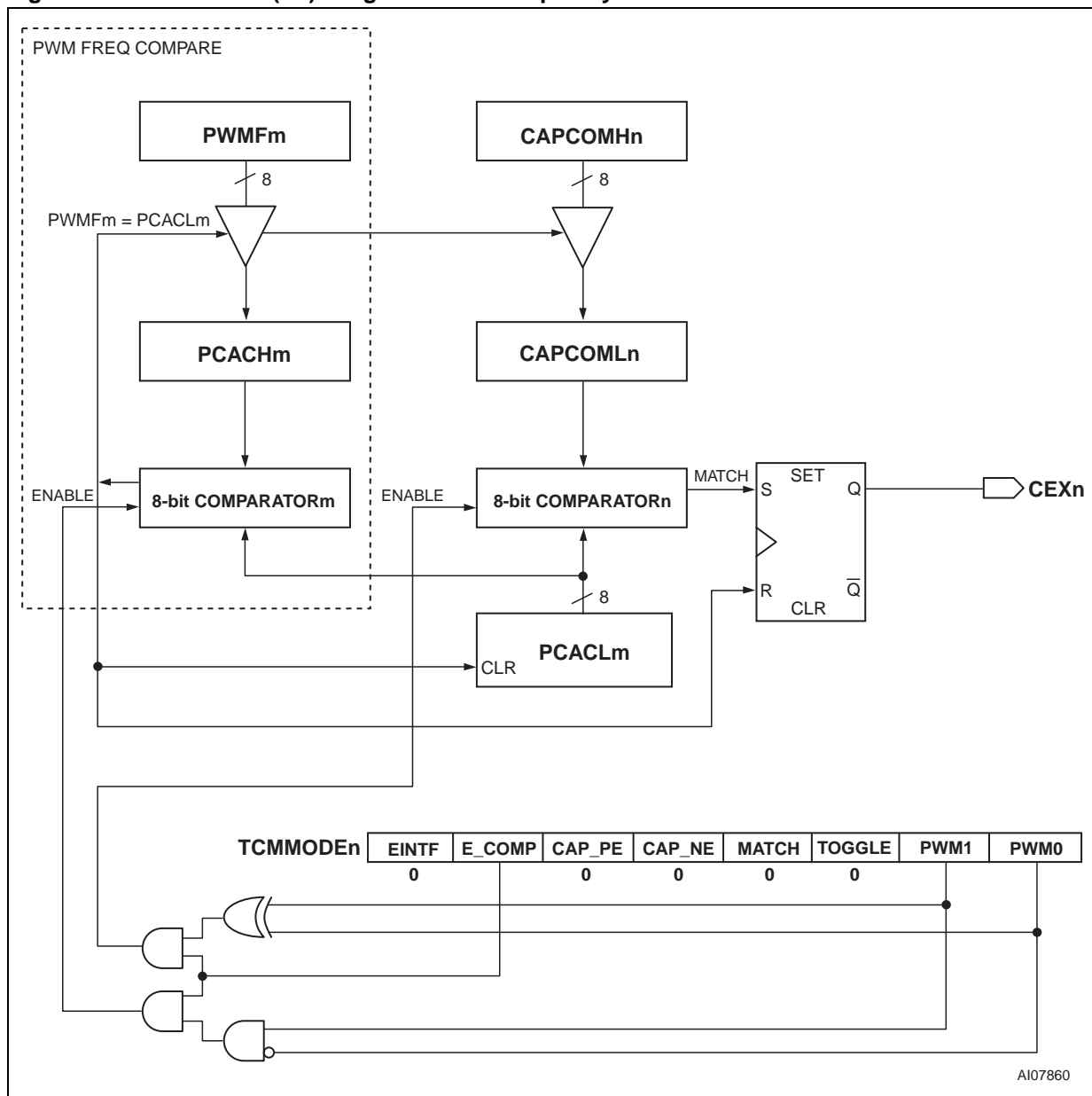
### PWM Mode - (X8), Programmable Frequency

In this mode, the PWM frequency is not determined by the overflow of the low byte of the Counter. Instead, the frequency is determined by the PWMFm register. The user can load a value in the PWMFm register, which is then compared to the low byte of the Counter. If there is a match, the Counter is cleared and the Load registers (PWMFm, CAPCOMHn) are re-loaded for the next PWM pulse. There is only one PWMFm Register which serves all 3 TCM in a PCA block.

If one of the TCM modules is operating in this mode, the other modules in the PCA must be configured to the same mode or left not to be used. The duty cycle of the PWM can be specified in the CAPCOMHn register as in the PWM with fixed frequency mode. Different TCM modules can have their own duty cycle.

**Note:** The value in the Frequency register (PWMFm) must be larger than the duty cycle register (CAPCOM).

Figure 26. PWM Mode - (X8) Programmable Frequency



Note: m = 0: n = 0, 1, or 2  
m = 1: n = 3, 4, or 5

**PWM Mode - Fixed Frequency, 16-bit**

The operation of the 16-bit PWM is the same as the 8-bit PWM with fixed frequency. In this mode, one or all the TCM can be configured to have a fixed frequency PWM output on the port pins. The PWM frequency is depending on the clock input frequency to the 16-bit Counter. The duty cycle of each TCM module can be specified in the CAPCOMHn and CAPCOMLn registers. When the 16 bit PCA\_Counter is equal or greater than the values in registers CAPCOMHn and CAPCOMLn, the PWM output is switched to a high state. When the PCA\_Counter overflows, CEXn is asserted low.

**Writing to Capture/Compare Registers**

When writing a 16-bit value to the PCA0 Capture/Compare registers, the low byte should always be written first. Writing to CAPCOMLn clears the

E\_COMP bit to '0'; writing to CAPCOMHn sets E\_COMP to '1' the largest duty cycle is 100% (CAPCOMHn CAPCOMLn = 0x0000), and the smallest duty cycle is 0.0015% (CAPCOMHn CAPCOMLn = 0xFFFF). A 0% duty cycle may be generated by clearing the E\_COMP bit to '0'.

**Control Register Bit Definition**

Each PCA has its own PCA\_CONFIGn, and each module within the PCA block has its own TCM\_Mode Register which defines the operation of that module (see Table 69 through Table 71). There is one PCA\_STATUS Register that covers both PCA0 and PCA1 (see Table 72, page 62 and Table 73, page 63).

**Table 69. PCA0 Control Register PCACON0 (0A4H, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EN-ALL	EN_PCA	EOVFI	PCAIDLE	–	–	CLK_SEL[1:0]	

**Table 70. PCA1 Control Register PCACON0 (0BCH, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EN-ALL	EN_PCA	EOVFI	PCAIDLE	–	–	CLK_SEL[1:0]	

**Table 71. PCA0, PCA1 Register Details**

Bit	Symbol	Function
7	EN-ALL	0 = No impact on TCM modules 1 = Enable both PCA counters simultaneously (override the EN_PCA Bits) This bit is to synchronize the two 16-bit counters in the PCA. For customers who want 5 PWM, for example, this bit can synchronize all of the PWM outputs.
6	EN_PCA	0 = PCA counter is disabled 1 = PCA counter is enabled EN_PCA Counter Run Control Bit. Set with software to turn the PCA counter on. Must be cleared with software to turn the PCA counter off.
5	EOVFI	1 = Enable Counter Overflow Interrupt if overflow flag (OVF) is set
4	PCAIDLE	0 = PCA operates when CPU is in Idle Mode 1 = PCA stops running when CPU is in Idle Mode
3-2	–	Reserved
1-0	CLK_SEL [1:0]	00 Select Prescaler clock as Counter clock 01 Select Timer 0 Overflow 10 Select External Clock pin (P4.3 for PCA0, P4.7 for PCA1) (MAX clock rate = f <sub>OSC</sub> /4)

**Table 72. PCA Status Register PCASTA (0A5H, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OVF1	INTF5	INTF4	INTF3	OVF0	INTF2	INTF1	INTF0

**Table 73. PCA Status Register PCASTA Details**

Bit	Symbol	Function
7	OVF1	PCA1 Counter OverFlow flag. Set by hardware when the counter rolls over. OVF1 flags an interrupt if Bit EOVI in PCACON1 is set. OVF1 may be set with either hardware or software but can only be cleared with software.
6	INTF5	TCM5 Interrupt flag. Set by hardware when a match or capture event occurs. Must be clear with software.
5	INTF4	TCM4 Interrupt flag. Set by hardware when a match or capture event occurs. Must be clear with software.
4	INTF3	TCM3 Interrupt flag. Set by hardware when a match or capture event occurs. Must be clear with software.
3	OVF0	PCA0 Counter OverFlow flag. Set by hardware when the counter rolls over. OVF0 flags an interrupt if Bit EOVI in PCACON0 is set. OVF0 may be set with either hardware or software but can only be cleared with software.
2	INTF2	TCM2 Interrupt flag. Set by hardware when a match or capture event occurs. Must be clear with software.
1	INTF1	TCM1 Interrupt flag. Set by hardware when a match or capture event occurs. Must be clear with software.
0	INTF0	TCM0 Interrupt flag. Set by hardware when a match or capture event occurs. Must be clear with software.

**TCM Interrupts**

There are 8 TCM interrupts: 6 match or capture interrupts and two counter overflow interrupts. The 8 interrupts are “ORed” as one PCA interrupt to the CPU. By the nature of PCA application, it is unlikely that many of the interrupts occur simultaneous-

ly. If they do, the CPU has to read the interrupt flags and determine which one to serve. The software has to clear the interrupt flag in the Status Register after serving the interrupt.

**Table 74. TCMMODE0 - TCMMODE5 (6 Registers, Reset Value 00H)**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EINTF	E_COMP	CAP_PE	CAP_NE	MATCH	TOGGLE	PWM[1:0]	

**Table 75. TCMMODE0 - TCMMODE5 Register Details**

e	Symbol	Function
7	EINTF	1 - Enable the interrupt flags (INTF) in the Status Register to generate an interrupt.
6	E_COMP	1 - Enable the comparator when set
5	CAP_PE	1 - Enable Capture Mode, a positive edge on the CEXn pin.
4	CAP_NE	1 - Enable Capture Mode, a negative edge on the CEXn pin.
3	MATCH	1 - A match from the comparator sets the INTF bits in the Status Register.
2	TOGGLE	1 - A match on the comparator results in a toggling output on CEXn pin.
1-0	PWM[1:0]	01 Enable PWM Mode (x8), fixed frequency. Enable the CEXn pin as a PWM output. 10 Enable PWM Mode (x8) with programmable frequency. Enable the CEXn pin as a PWM output. 11 Enable PWM Mode (x16), fixed frequency. Enable the CEXn pin as a PWM output.

**Table 76. TCMMODE Register Configurations**

EINTF	E_COMP	CAP_PE	CAP_NE	MATCH	TOGGLE	PWM1	PWM0	TCM FUNCTION
0	0	0	0	0	0	0	0	No operation (reset value)
0	1	0	0	0	0	0	1	8-bit PWM, fixed frequency
0	1	0	0	0	0	1	0	8-bit PWM, programmable frequency
0	1	0	0	0	0	1	1	16-bit PWM, fixed frequency
X	1	0	0	1	1	0	0	16-bit toggle
X	1	0	0	1	0	0	0	16-bit Software Timer
X	X	0	1	0	0	0	0	16-bit capture, negative trigger
X	X	1	0	0	0	0	0	16-bit capture, positive trigger
X	X	1	1	0	0	0	0	16-bit capture, transition trigger



## PSD MODULE

- The PSD Module provides configurable Program and Data memories to the Turbo 8032 MCU Module. In addition, it has its own set of I/O ports and a PLD with 16 macrocells for general logic implementation.
- Ports A,B,C, and D are general purpose programmable I/O ports that have a port architecture which is different from the I/O ports in the MCU Module.
- The PSD Module communicates with the MCU Module through the internal address, data bus (A0-A15, D0-D7) and control signals (RD, WR, PSEN, ALE, RESET). The user defines the Decoding PLD in the PSDsoft Development Tool and can map the resources in the PSD Module to any program or data address space. Figure 27, page 66 shows the functional blocks in the PSD Module.

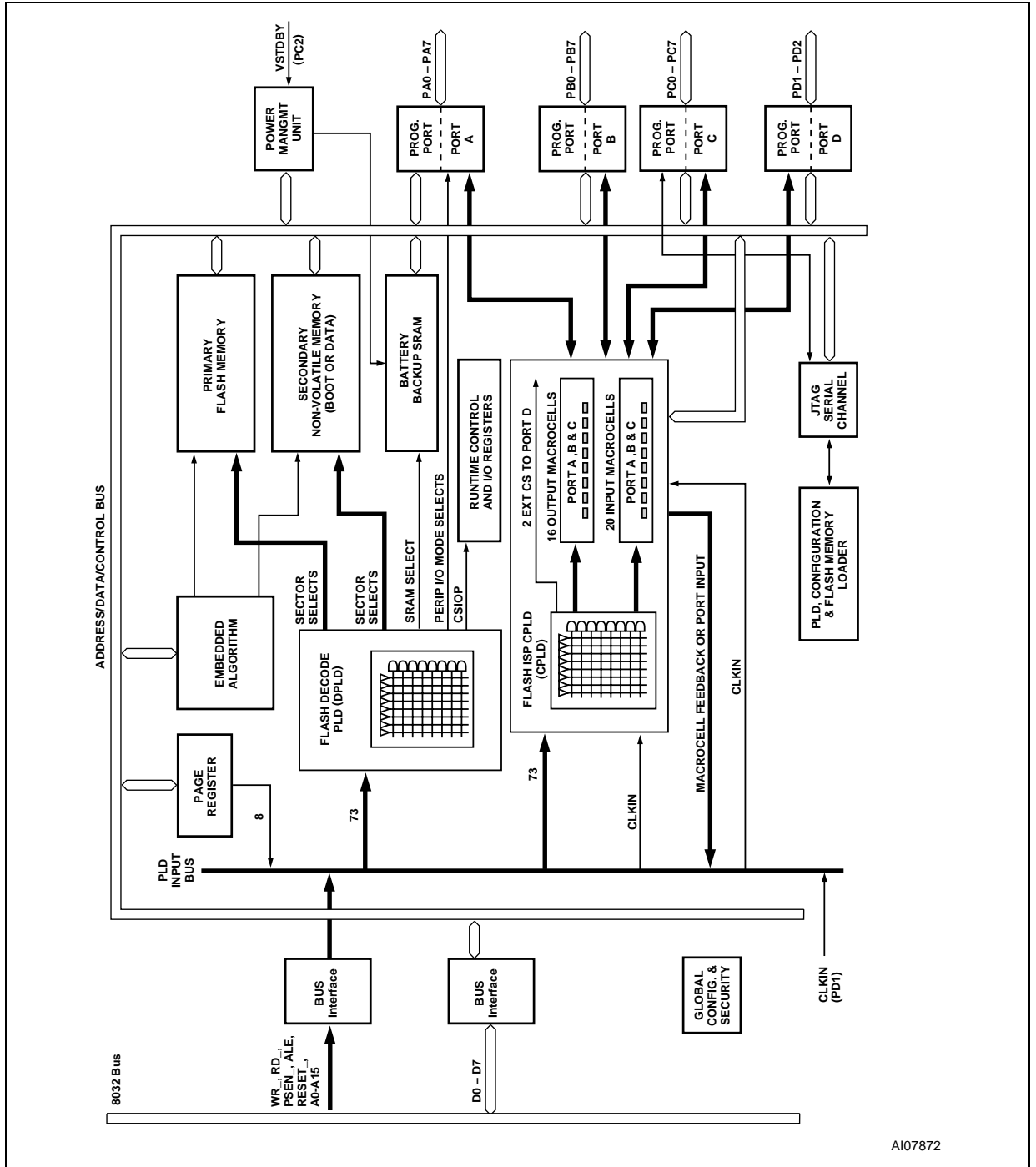
### Functional Overview

- 64K, 128K, or 256K bytes Flash memory. This is the main Flash memory. It is divided into equally-sized blocks that can be accessed with user-specified addresses.
- Secondary 16K or 32K bytes Flash boot memory. It is divided into equally-sized blocks that can be accessed with user-specified addresses. This secondary memory brings the ability to execute code and update the main Flash *concurrently*.
- 2K, 8K, or 32K bytes SRAM. The SRAM's contents can be protected from a power failure by connecting an external battery.
- CPLD with 16 Output Micro Cells (OMCs) and 20 Input Micro Cells (IMCs). The CPLD may be used to efficiently implement a variety of logic functions for internal and external control.

Examples include state machines, loadable shift registers, and loadable counters.

- Decode PLD (DPLD) that decodes address for selection of memory blocks in the PSD Module.
- Configurable I/O ports (Port A,B,C and D) that can be used for the following functions:
  - MCU I/Os
  - PLD I/Os
  - Latched MCU address output
  - Special function I/Os.
  - I/O ports may be configured as open-drain outputs.
- Built-in JTAG compliant serial port allows full-chip In-System Programmability (ISP). With it, you can program a blank device or reprogram a device in the factory or the field.
- Internal page register that can be used to expand the 8032 MCU Module address space by a factor of 256.
- Internal programmable Power Management Unit (PMU) that supports a low-power mode called Power-down Mode. The PMU can automatically detect a lack of the 8032 CPU core activity and put the PSD Module into Power-down Mode.
- Erase/WRITE cycles:
  - Flash memory - 100,000 minimum
  - PLD - 1,000 minimum
  - Data Retention: 15 year minimum (for Main Flash memory, Boot, PLD and Configuration bits)

Figure 27. PSD Module Block Diagram



AI07872

**PSD MODULE REGISTER DESCRIPTION AND ADDRESS OFFSET**

Table 77 shows the offset addresses to the PSD Module registers relative to the CSIOP base address. The CSIOP space is the 256 bytes of address that is allocated by the user to the internal

PSD Module registers. Table 79, page 69 provides brief descriptions of the registers in CSIOP space. The following section gives a more detailed description.

**Table 77. Register Address Offset**

Register Name	Port A	Port B	Port C	Port D	Other <sup>(1)</sup>	Description
Data In	00	01	10	11		Reads Port pin as input, MCU I/O Input Mode
Control	02	03				Selects mode between MCU I/O or Address Out
Data Out	04	05	12	13		Stores data for output to Port pins, MCU I/O Output Mode
Direction	06	07	14	15		Configures Port pin as input or output
Drive Select	08	09	16	17		Configures Port pins as either CMOS or Open Drain on some pins, while selecting high slew rate on other pins.
Input Macrocell	0A	0B	18			Reads Input Macrocells
Enable Out	0C	0D	1A	1B		Reads the status of the output enable to the I/O Port driver
Output Macrocells AB	20	20				READ – reads output of macrocells AB WRITE – loads macrocell flip-flops
Output Macrocells BC		21	21			READ – reads output of macrocells BC WRITE – loads macrocell flip-flops
Mask Macrocells AB	22	22				Blocks writing to the Output Macrocells AB
Mask Macrocells BC		23	23			Blocks writing to the Output Macrocells BC
Primary Flash Protection					C0	Read-only – Primary Flash Sector Protection
Secondary Flash memory Protection					C2	Read-only – PSD Module Security and Secondary Flash memory Sector Protection
PMMR0					B0	Power Management Register 0
PMMR2					B4	Power Management Register 2
Page					E0	Page Register
VM					E2	Places PSD Module memory areas in Program and/or Data space on an individual basis.

Note: 1. Other registers that are not part of the I/O ports.

**PSD MODULE DETAILED OPERATION**

As shown in Figure 27, page 66, the PSD Module consists of five major types of functional blocks:

- Memory Block
- PLD Blocks
- I/O Ports
- Power Management Unit (PMU)
- JTAG Interface

The functions of each block are described in the following sections. Many of the blocks perform multiple functions, and are user configurable.

**MEMORY BLOCKS**

The PSD Module has the following memory blocks (see Table 78):

- Primary Flash memory
- Secondary Flash memory
- SRAM

The Memory Select signals for these blocks originate from the Decode PLD (DPLD) and are user-defined in PSDsoft Express.

**Primary Flash Memory and Secondary Flash memory Description**

The primary Flash memory is divided evenly into eight equal sectors. The secondary Flash memory is divided into four equal sectors. Each sector of either memory block can be separately protected from Program and Erase cycles.

Flash memory may be erased on a sector-by-sector basis. Flash sector erasure may be suspended while data is read from other sectors of the block and then resumed after reading.

During a Program or Erase cycle in Flash memory, the status can be output on Ready/Busy (PC3). This pin is set up using PSDsoft Express Configuration.

**Memory Block Select Signals**

The DPLD generates the Select signals for all the internal memory blocks (see the section entitled “PLDs,” page 78). Each of the sectors of the primary Flash memory has a Select signal (FS0-FS7) which can contain up to three product terms. Each of the four sectors of the secondary Flash memory has a Select signal (CSBOOT0-CSBOOT3) which can contain up to three product terms. Having three product terms for each Select signal allows a given sector to be mapped in Program or Data space.

**Flash Memory Instructions**

The Flash memory instructions are detailed in Table 79. For efficient decoding of the instructions, the first two bytes of an instruction are the coded cycles and are followed by an instruction byte or confirmation byte. The coded cycles consist of writing the data AAh to address X555h during the first cycle and data 55h to address XAAAh during the second cycle. Address signals A15-A12 are Don't Care during the instruction WRITE cycles. However, the appropriate Sector Select (FS0-FS7 or CSBOOT0-CSBOOT3) must be selected.

The primary and secondary Flash memories have the same instruction set (except for Read Primary Flash Identifier). The Sector Select signals determine which Flash memory is to receive and execute the instruction. The primary Flash memory is selected if any one of Sector Select (FS0-FS7) is High, and the secondary Flash memory is selected if any one of Sector Select (CSBOOT0-CSBOOT3) is High.

**Table 78. μPSD33XX Memory Configuration**

Device	Main Flash			Secondary Flash			SRAM
	Flash Size	Sector Size	# of Sectors (Sector Select Signal)	Flash Size	Sector Size	# of Sectors (Sector Select Signal)	Size
μPSD3312	64KB	16KB	4 (FS0-3)	16KB	8KB	2 (CSBOOT0-1)	2KB
μPSD3333	128KB	16KB	8 (FS0-7)	32KB	8KB	4 (CSBOOT0-3)	8KB
μPSD3334	256KB	32KB	8 (FS0-7)	32KB	8KB	4 (CSBOOT0-3)	32KB
μPSD3354	256KB	32KB	8 (FS0-7)	32KB	8KB	4 (CSBOOT0-3)	32KB

**Table 79. Instructions**

Instruction	FS0-FS7 or CSBOOT0-CSBOOT3	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5	Cycle 6	Cycle 7
READ <sup>(5)</sup>	1	“Read” RD @ RA						
READ Sector Protection <sup>(6,8,13)</sup>	1	AAh@ X555h	55h@ XAAAh	90h@ X555h	Read status @ XX02h			
Program a Flash Byte <sup>(13)</sup>	1	AAh@ X555h	55h@ XAAAh	A0h@ X555h	PD@ PA			
Flash Sector Erase <sup>(7,13)</sup>	1	AAh@ X555h	55h@ XAAAh	80h@ X555h	AAh@ X555h	55h@ XAAAh	30h@ SA	30h <sup>7</sup> @ next SA
Flash Bulk Erase <sup>(13)</sup>	1	AAh@ X555h	55h@ XAAAh	80h@ X555h	AAh@ X555h	55h@ XAAAh	10h@ X555h	
Suspend Sector Erase <sup>(11)</sup>	1	B0h@ XXXXh						
Resume Sector Erase <sup>(12)</sup>	1	30h@ XXXXh						
RESET <sup>(6)</sup>	1	F0h@ XXXXh						
Unlock Bypass	1	AAh@ X555h	55h@ XAAAh	20h@ X555h				
Unlock Bypass Program <sup>(9)</sup>	1	A0h@ XXXXh	PD@ PA					
Unlock Bypass Reset <sup>(10)</sup>	1	90h@ XXXXh	00h@ XXXXh					

- Note: 1. All bus cycles are WRITE bus cycles, except the ones with the “Read” label
2. All values are in hexadecimal:  
 X = Don't care. Addresses of the form XXXXh, in this table, must be even addresses  
 RA = Address of the memory location to be read  
 RD = Data READ from location RA during the READ cycle  
 PA = Address of the memory location to be programmed. Addresses are latched on the falling edge of WRITE Strobe ( $\overline{WR}$ , CNTL0).  
 PA is an even address for PSD in Word Programming Mode.  
 PD = Data word to be programmed at location PA. Data is latched on the rising edge of WRITE Strobe ( $\overline{WR}$ , CNTL0)  
 SA = Address of the sector to be erased or verified. The Sector Select (FS0-FS7 or CSBOOT0-CSBOOT3) of the sector to be erased, or verified, must be Active (High).
3. Sector Select (FS0-FS7 or CSBOOT0-CSBOOT3) signals are active High, and are defined in PSDsoft Express.
4. Only address Bits A11-A0 are used in instruction decoding.
5. No Unlock or instruction cycles are required when the device is in the READ Mode
6. The RESET instruction is required to return to the READ Mode after reading the Sector Protection Status, or if the Error Flag Bit (DQ5/DQ13) goes High.
7. Additional sectors to be erased must be written at the end of the Sector Erase instruction within 80µs.
8. The data is 00h for an unprotected sector, and 01h for a protected sector. In the fourth cycle, the Sector Select is active, and (A1,A0)=(1,0)
9. The Unlock Bypass instruction is required prior to the Unlock Bypass Program instruction.
10. The Unlock Bypass Reset Flash instruction is required to return to reading memory data when the device is in the Unlock Bypass Mode.
11. The system may perform READ and Program cycles in non-erasing sectors, read the Flash ID or read the Sector Protection Status when in the Suspend Sector Erase Mode. The Suspend Sector Erase instruction is valid only during a Sector Erase cycle.
12. The Resume Sector Erase instruction is valid only during the Suspend Sector Erase Mode.
13. The MCU cannot invoke these instructions while executing code from the same Flash memory as that for which the instruction is intended. The MCU must retrieve, for example, the code from the secondary Flash memory when reading the Sector Protection Status of the primary Flash memory.

## READ Flash Memory

After power-up, the MCU may read the primary Flash memory or the secondary Flash memory using READ operations just as it would a ROM or RAM device. Alternately, the MCU may use READ operations to obtain status information about a Program or Erase cycle that is currently in progress. Lastly, the MCU may use instructions to read sector protection and the Erase/Program status bits.

**Reading the Erase/Program Status Bits.** The Flash memory provides several status bits to be used by the MCU to confirm the completion of an Erase or Program cycle of Flash memory. These status bits minimize the time that the MCU spends performing these tasks and are defined in Table 80, page 71. The status bits can be read as many times as needed.

For Flash memory, the MCU can perform a READ operation to obtain these status bits while an Erase or Program instruction is being executed by the embedded algorithm. See the section entitled “Programming Flash Memory, page 71,” for details.

**Data Polling Flag (DQ7).** When erasing or programming in Flash memory, the Data Polling Flag Bit (DQ7) outputs the complement of the bit being entered for programming/writing on the DQ7 Bit. Once the Program instruction or the WRITE operation is completed, the true logic value is read on the Data Polling Flag Bit (DQ7) (in a READ operation).

- Data Polling is effective after the fourth WRITE pulse (for a Program instruction) or after the sixth WRITE pulse (for an Erase instruction). It must be performed at the address being programmed or at an address within the Flash memory sector being erased.
- During an Erase cycle, the Data Polling Flag Bit (DQ7) outputs a '0.' After completion of the cycle, the Data Polling Flag Bit (DQ7) outputs the last bit programmed (it is a '1' after erasing).
- If the byte to be programmed is in a protected Flash memory sector, the instruction is ignored.
- If all the Flash memory sectors to be erased are protected, the Data Polling Flag Bit (DQ7) is reset to '0' for about 100μs, and then returns to the previous addressed byte. No erasure is performed.

**Toggle Flag (DQ6).** The Flash memory offers another way for determining when the Program cycle is completed. During the internal WRITE operation and when either the FS0-FS7 or CSBOOT0-CSBOOT3 is true, the Toggle Flag Bit (DQ6) toggles from '0' to '1' and '1' to '0' on subsequent attempts to read any byte of the memory.

When the internal cycle is complete, the toggling stops and the data READ on the Data Bus D0-D7 is the addressed memory byte. The device is now accessible for a new READ or WRITE operation. The cycle is finished when two successive Reads yield the same output data.

- The Toggle Flag Bit (DQ6) is effective after the fourth WRITE pulse (for a Program instruction) or after the sixth WRITE pulse (for an Erase instruction).
- If the byte to be programmed belongs to a protected Flash memory sector, the instruction is ignored.
- If all the Flash memory sectors selected for erasure are protected, the Toggle Flag Bit (DQ6) toggles to '0' for about 100μs and then returns to the previous addressed byte.

**Error Flag (DQ5).** During a normal Program or Erase cycle, the Error Flag Bit (DQ5) is to '0.' This bit is set to '1' when there is a failure during Flash memory Byte Program, Sector Erase, or Bulk Erase cycle.

In the case of Flash memory programming, the Error Flag Bit (DQ5) indicates the attempt to program a Flash memory bit from the programmed state, '0', to the erased state, '1,' which is not valid. The Error Flag Bit (DQ5) may also indicate a Time-out condition while attempting to program a byte.

In case of an error in a Flash memory Sector Erase or Byte Program cycle, the Flash memory sector in which the error occurred or to which the programmed byte belongs must no longer be used. Other Flash memory sectors may still be used. The Error Flag Bit (DQ5) is reset after a Reset Flash instruction.

**Erase Time-out Flag (DQ3).** The Erase Time-out Flag Bit (DQ3) reflects the time-out period allowed between two consecutive Sector Erase instructions. The Erase Time-out Flag Bit (DQ3) is reset to '0' after a Sector Erase cycle for a time period of 100μs + 20% unless an additional Sector Erase instruction is decoded. After this time period, or when the additional Sector Erase instruction is decoded, the Erase Time-out Flag Bit (DQ3) is set to '1.'

**Table 80. Status Bit**

Functional Block	FS0-FS7/ CSBOOT0- CSBOOT3	DQ7 <sup>(2)</sup>	DQ6 <sup>(2)</sup>	DQ5 <sup>(2)</sup>	DQ4 <sup>(1,2)</sup>	DQ3 <sup>(2)</sup>	DQ2 <sup>(1,2)</sup>	DQ1 <sup>(1,2)</sup>	DQ0 <sup>(1,2)</sup>
Flash Memory	V <sub>IH</sub> <sup>(3)</sup>	Data Polling	Toggle Flag	Error Flag	X	Erase Time-out	X	X	X

Note: 1. X = Not guaranteed value, can be read either '1' or '0.'  
 2. DQ7-DQ0 represent the Data Bus bits, D7-D0.  
 3. FS0-FS7 and CSBOOT0-CSBOOT3 are active High.

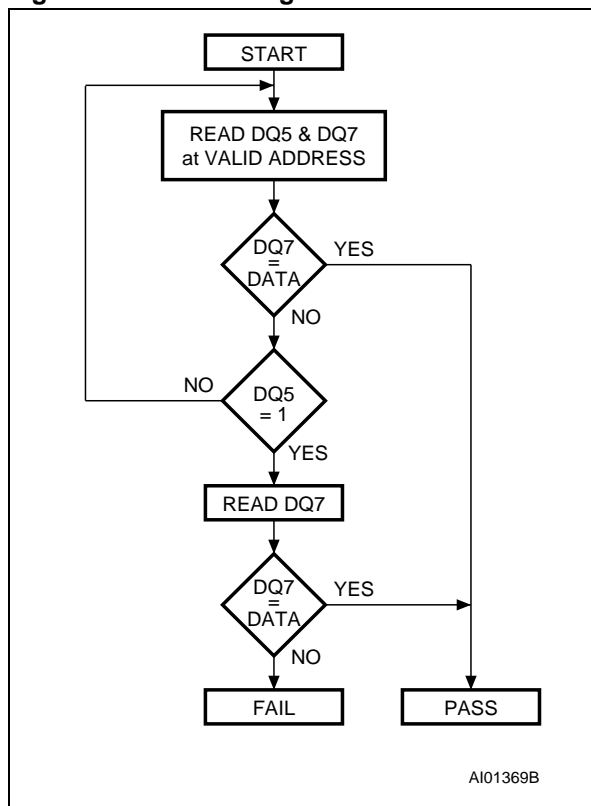
**Programming Flash Memory**

Flash memory must be erased prior to being programmed. A byte of Flash memory is erased to all '1s' (FFh), and is programmed by setting selected bits to '0.' The MCU may erase Flash memory all at once or by-sector, but not byte-by-byte. However, the MCU may program Flash memory byte-by-byte.

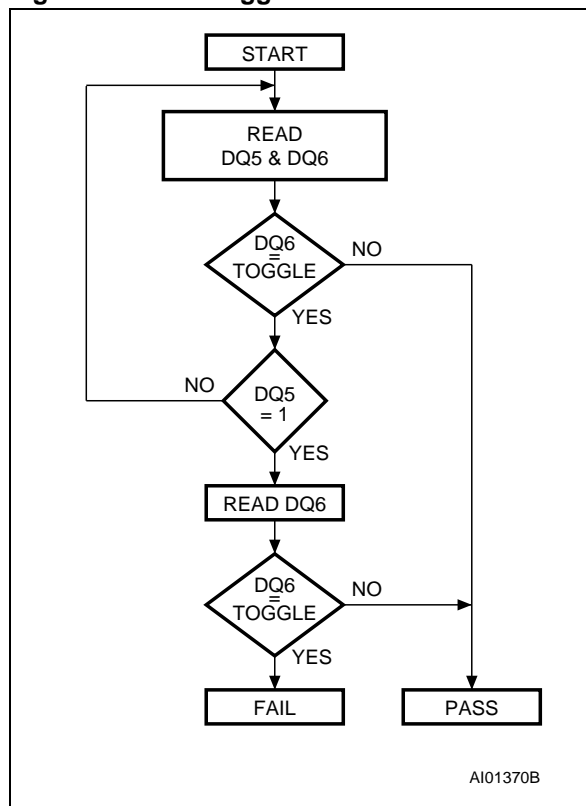
The primary and secondary Flash memories require the MCU to send an instruction to program a byte or to erase sectors (see Table 79, page 69).

Once the MCU issues a Flash memory Program or Erase instruction, it must check for the status bits for completion. The embedded algorithms that are invoked support several means to provide status to the MCU. Status may be checked using any of three methods: Data Polling, Data Toggle, or Ready/Busy (PC3).

**Figure 28. Data Polling Flowchart**



**Figure 29. Data Toggle Flowchart**



**Unlock Bypass.** The Unlock Bypass instructions allow the system to program bytes to the Flash memories faster than using the standard Program instruction. The Unlock Bypass Mode is entered by first initiating two Unlock cycles. This is followed by a third WRITE cycle containing the Unlock Bypass code, 20h (as shown in Table 79).

The Flash memory then enters the Unlock Bypass Mode. A two-cycle Unlock Bypass Program instruction is all that is required to program in this mode. The first cycle in this instruction contains the Unlock Bypass Program code, A0h. The second cycle contains the program address and data. Additional data is programmed in the same manner. These instructions dispense with the initial two Unlock cycles required in the standard Program instruction, resulting in faster total Flash memory programming.

During the Unlock Bypass Mode, only the Unlock Bypass Program and Unlock Bypass Reset Flash instructions are valid.

To exit the Unlock Bypass Mode, the system must issue the two-cycle Unlock Bypass Reset Flash instruction. The first cycle must contain the data 90h; the second cycle the data 00h. Addresses are Don't Care for both cycles. The Flash memory then returns to READ Mode.

### Erasing Flash Memory

**Flash Bulk Erase.** The Flash Bulk Erase instruction uses six WRITE operations followed by a READ operation of the status register, as described in Table 79. If any byte of the Bulk Erase instruction is wrong, the Bulk Erase instruction aborts and the device is reset to the READ Flash memory status.

During a Bulk Erase, the memory status may be checked by reading the Error Flag Bit (DQ5), the Toggle Flag Bit (DQ6), and the Data Polling Flag Bit (DQ7), as detailed in Programming Flash Memory, page 71. The Error Flag Bit (DQ5) returns a '1' if there has been an Erase Failure (maximum number of Erase cycles have been executed).

It is not necessary to program the memory with 00h because the PSD Module automatically does this before erasing to 0FFh.

During execution of the Bulk Erase instruction, the Flash memory does not accept any instructions.

**Flash Sector Erase.** The Sector Erase instruction uses six WRITE operations, as described in Table 79. Additional Flash Sector Erase codes and Flash memory sector addresses can be written subsequently to erase other Flash memory sectors in parallel, without further coded cycles, if the additional bytes are transmitted in a shorter time than the time-out period of about 100μs. The input of a new Sector Erase code restarts the time-out period.

The status of the internal timer can be monitored through the level of the Erase Time-out Flag Bit (DQ3). If the Erase Time-out Flag Bit (DQ3) is '0,' the Sector Erase instruction has been received and the time-out period is counting. If the Erase Time-out Flag Bit (DQ3) is '1,' the time-out period has expired and the embedded algorithm is busy erasing the Flash memory sector(s). Before and during Erase time-out, any instruction other than Suspend Sector Erase and Resume Sector Erase instructions abort the cycle that is currently in progress, and reset the device to READ Mode.

During a Sector Erase, the memory status may be checked by reading the Error Flag Bit (DQ5), the Toggle Flag Bit (DQ6), and the Data Polling Flag Bit (DQ7), as detailed in Programming Flash Memory, page 71.

During execution of the Erase cycle, the Flash memory accepts only RESET and Suspend Sector Erase instructions. Erasure of one Flash memory sector may be suspended, in order to read data from another Flash memory sector, and then resumed.

**Suspend Sector Erase.** When a Sector Erase cycle is in progress, the Suspend Sector Erase instruction can be used to suspend the cycle by writing 0B0h to any address when an appropriate Sector Select (FS0-FS7 or CSBOOT0-CSBOOT3) is High. (See Table 79). This allows reading of data from another Flash memory sector after the Erase cycle has been suspended. Suspend Sector Erase is accepted only during an Erase cycle and defaults to READ Mode. A Suspend Sector Erase instruction executed during an Erase time-out period, in addition to suspending the Erase cycle, terminates the time out period.

The Toggle Flag Bit (DQ6) stops toggling when the internal logic is suspended. The status of this bit must be monitored at an address within the Flash memory sector being erased. The Toggle Flag Bit (DQ6) stops toggling between 0.1μs and 15μs after the Suspend Sector Erase instruction has been executed. The Flash memory is then automatically set to READ Mode.

If an Suspend Sector Erase instruction was executed, the following rules apply:

- Attempting to read from a Flash memory sector that was being erased outputs invalid data.
- Reading from a Flash sector that was *not* being erased is valid.
- The Flash memory *cannot* be programmed, and only responds to Resume Sector Erase and Reset Flash instructions (READ is an operation and is allowed).
- If a Reset Flash instruction is received, data in the Flash memory sector that was being erased is invalid.



**Resume Sector Erase.** If a Suspend Sector Erase instruction was previously executed, the erase cycle may be resumed with this instruction. The Resume Sector Erase instruction consists of writing 030h to any address while an appropriate Sector Select (FS0-FS7 or CSBOOT0-CSBOOT3) is High. (See Table 79.)

**Specific Features**

**Flash Memory Sector Protect.** Each primary and secondary Flash memory sector can be separately protected against Program and Erase cycles. Sector Protection provides additional data security because it disables all Program or Erase cycles. This mode can be activated through the JTAG Port or a Device Programmer.

Sector protection can be selected for each sector using the PSDsoft Express Configuration pro-

gram. This automatically protects selected sectors when the device is programmed through the JTAG Port or a Device Programmer. Flash memory sectors can be unprotected to allow updating of their contents using the JTAG Port or a Device Programmer. The MCU can read (but cannot change) the sector protection bits.

Any attempt to program or erase a protected Flash memory sector is ignored by the device. The Verify operation results in a READ of the protected data. This allows a guarantee of the retention of the Protection status.

The sector protection status can be read by the MCU through the Flash memory protection registers (in the CSIOP block). See Table 81 and Table 82.

**Table 81. Sector Protection/Security Bit Definition – Flash Protection Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Sec7_Prot	Sec6_Prot	Sec5_Prot	Sec4_Prot	Sec3_Prot	Sec2_Prot	Sec1_Prot	Sec0_Prot

Note: Bit Definitions:  
 Sec<i>\_Prot 1 = Primary Flash memory or secondary Flash memory Sector <i> is write-protected.  
 Sec<i>\_Prot 0 = Primary Flash memory or secondary Flash memory Sector <i> is not write-protected.

**Table 82. Sector Protection/Security Bit Definition – Secondary Flash Protection Register**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Security_Bit	not used	not used	not used	Sec3_Prot	Sec2_Prot	Sec1_Prot	Sec0_Prot

Note: Bit Definitions:  
 Sec<i>\_Prot 1 = Secondary Flash memory Sector <i> is write-protected.  
 Sec<i>\_Prot 0 = Secondary Flash memory Sector <i> is not write-protected.  
 Security\_Bit 0 = Security Bit in device has not been set; 1 = Security Bit in device has been set.

**Reset Flash.** The Reset Flash instruction consists of one WRITE cycle (see Table 79). It can also be optionally preceded by the standard two WRITE decoding cycles (writing AAh to 55h and 55h to AAh). It must be executed after:

- Reading the Flash Protection Status
- An Error condition has occurred (and the device has set the Error Flag Bit (DQ5) to '1' during a Flash memory Program or Erase cycle.

The Reset Flash instruction puts the Flash memory back into normal READ Mode. If an Error condition has occurred (and the device has set the Error Flag Bit (DQ5) to '1' the Flash memory is put back into normal READ Mode within 25μs of the Reset Flash instruction having been issued. The Reset Flash instruction is ignored when it is issued during a Program or Bulk Erase cycle of the Flash memory. The Reset Flash instruction aborts any on-going Sector Erase cycle, and returns the Flash memory to the normal READ Mode within 25μs.

**Reset (RESET) Signal.** A pulse on Reset (RESET) aborts any cycle that is in progress, and resets the Flash memory to the READ Mode. When the reset occurs during a Program or Erase cycle, the Flash memory takes up to 25μs to return to the READ Mode. It is recommended that the Reset (RESET) pulse (except for Power-on RESET, as described on page 94) be at least 25μs so that the Flash memory is always ready for the MCU to retrieve the bootstrap instructions after the reset cycle is complete.

**SRAM**

The SRAM is enabled when SRAM Select (RS0) from the DPLD is High. SRAM Select (RS0) can contain up to two product terms, allowing flexible memory mapping.

The SRAM can be backed up using an external battery. The external battery should be connected to Voltage Stand-by (V<sub>STBY</sub>, PC2). If you have an external battery connected to the μPSD3200, the

contents of the SRAM are retained in the event of a power loss. The contents of the SRAM are retained so long as the battery voltage remains at 2V or greater. If the supply voltage falls below the battery voltage, an internal power switch-over to the battery occurs.

PC4 can be configured as an output that indicates when power is being drawn from the external battery. Battery-on Indicator (V<sub>BATON</sub>, PC4) is High with the supply voltage falls below the battery voltage and the battery on Voltage Stand-by (V<sub>STBY</sub>, PC2) is supplying power to the internal SRAM.

SRAM Select (RS0), Voltage Stand-by (V<sub>STBY</sub>, PC2) and Battery-on Indicator (V<sub>BATON</sub>, PC4) are all configured using PSDsoft Express Configuration.

**Sector Select and SRAM Select**

Sector Select (FS0-FS7, CSBOOT0-CSBOOT3) and SRAM Select (RS0) are all outputs of the DPLD. They are setup by writing equations for them in PSDsoft Express. The following rules apply to the equations for these signals:

1. Primary Flash memory and secondary Flash memory Sector Select signals must *not* be larger than the physical sector size.
2. Any primary Flash memory sector must *not* be mapped in the same memory space as another Flash memory sector.
3. A secondary Flash memory sector must *not* be mapped in the same memory space as another secondary Flash memory sector.
4. SRAM, I/O, and Peripheral I/O spaces must *not* overlap.
5. A secondary Flash memory sector *may* overlap a primary Flash memory sector. In case of overlap, priority is given to the secondary Flash memory sector.
6. SRAM, I/O, and Peripheral I/O spaces *may* overlap any other memory sector. Priority is given to the SRAM, I/O, or Peripheral I/O.



**Example.** FS0 is valid when the address is in the range of 8000h to BFFFh, CSBOOT0 is valid from 8000h to 9FFFh, and RS0 is valid from 8000h to 87FFh. Any address in the range of RS0 always accesses the SRAM. Any address in the range of CSBOOT0 greater than 87FFh (and less than 9FFFh) automatically addresses secondary Flash memory segment 0. Any address greater than 9FFFh accesses the primary Flash memory segment 0. You can see that half of the primary Flash memory segment 0 and one-fourth of secondary Flash memory segment 0 cannot be accessed in this example.

**Note:** An equation that defined FS1 to anywhere in the range of 8000h to BFFFh would *not* be valid.

Figure 30 shows the priority levels for all memory components. Any component on a higher level can overlap and has priority over any component on a lower level. Components on the same level must *not* overlap. Level one has the highest priority and level 3 has the lowest.

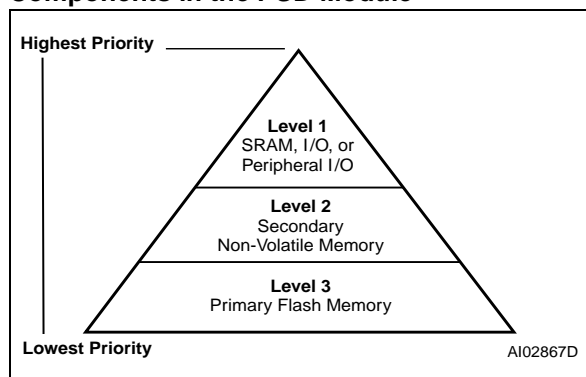
**Memory Select Configuration in Program and Data Spaces.** The MCU Core has separate address spaces for Program memory and Data memory. Any of the memories within the PSD Module can reside in either space or both spaces. This is controlled through manipulation of the VM Register that resides in the CSIOP space.

The VM Register is set using PSDsoft Express to have an initial value. It can subsequently be

changed by the MCU so that memory mapping can be changed on-the-fly.

For example, you may wish to have SRAM and primary Flash memory in the Data space at Boot-up, and secondary Flash memory in the Program space at Boot-up, and later swap the primary and secondary Flash memories. This is easily done with the VM Register by using PSDsoft Express Configuration to configure it for Boot-up and having the MCU change it when desired. Table 83 describes the VM Register.

**Figure 30. Priority Level of Memory and I/O Components in the PSD Module**



**Table 83. VM Register**

Bit 7 PIO_EN	Bit 6	Bit 5	Bit 4 Primary FL_Data	Bit 3 Secondary Data	Bit 2 Primary FL_Code	Bit 1 Secondary Code	Bit 0 SRAM_Code
0 = disable PIO Mode	not used	not used	0 = $\overline{RD}$ can't access Flash memory	0 = $\overline{RD}$ can't access Secondary Flash memory	0 = $\overline{PSEN}$ can't access Flash memory	0 = $\overline{PSEN}$ can't access Secondary Flash memory	0 = $\overline{PSEN}$ can't access SRAM
1 = enable PIO Mode	not used	not used	1 = $\overline{RD}$ access Flash memory	1 = $\overline{RD}$ access Secondary Flash memory	1 = $\overline{PSEN}$ access Flash memory	1 = $\overline{PSEN}$ access Secondary Flash memory	1 = $\overline{PSEN}$ access SRAM

**Separate Space Mode.** Program space is separated from Data space. For example, Program Select Enable ( $\overline{\text{PSEN}}$ ) is used to access the program code from the primary Flash memory, while READ Strobe ( $\overline{\text{RD}}$ ) is used to access data from the secondary Flash memory, SRAM and I/O Port blocks. This configuration requires the VM Register to be set to 0Ch (see Figure 31).

**Combined Space Modes.** The Program and Data spaces are combined into one memory space that allows the primary Flash memory, secondary Flash memory, and SRAM to be accessed by either Program Select Enable ( $\overline{\text{PSEN}}$ ) or READ Strobe ( $\overline{\text{RD}}$ ). For example, to configure the primary Flash memory in Combined space, Bits b2 and b4 of the VM Register are set to '1' (see Figure 32).

Figure 31. Separate Space Mode

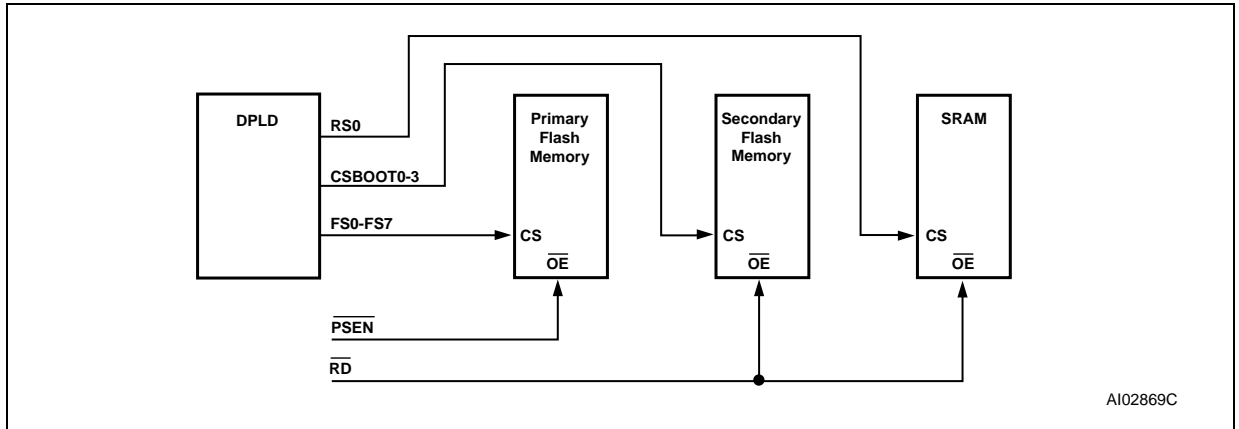
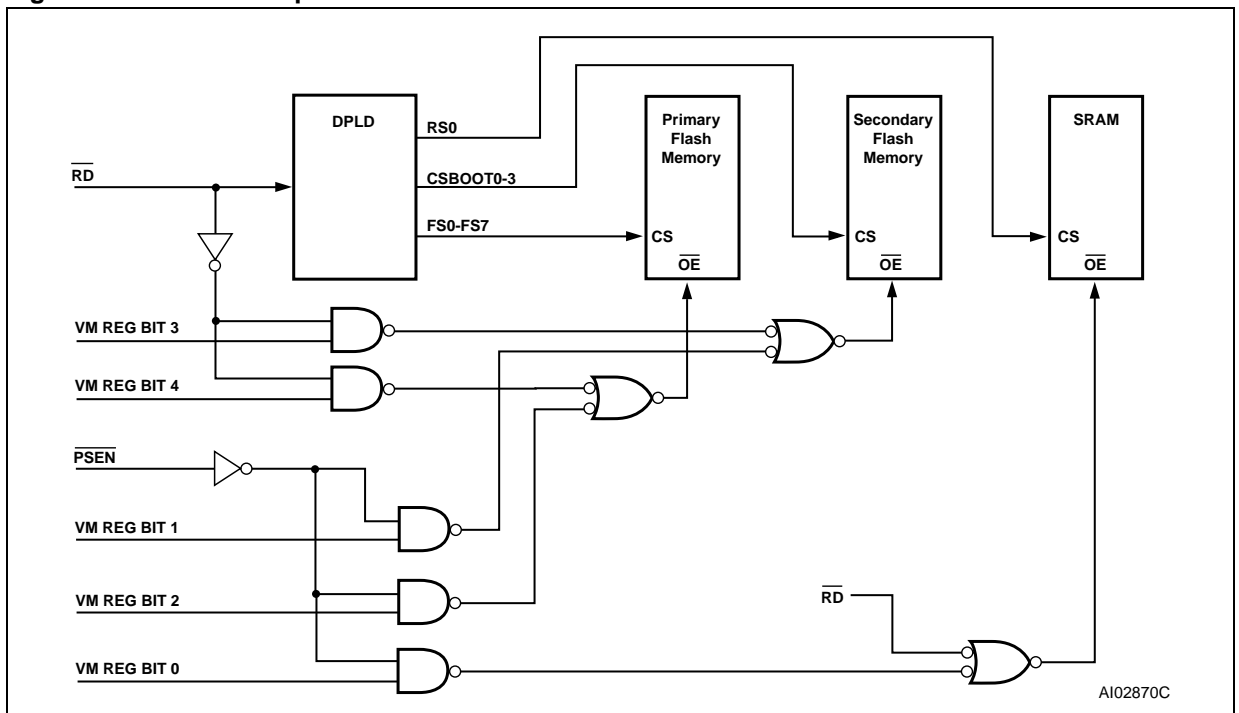


Figure 32. Combined Space Mode



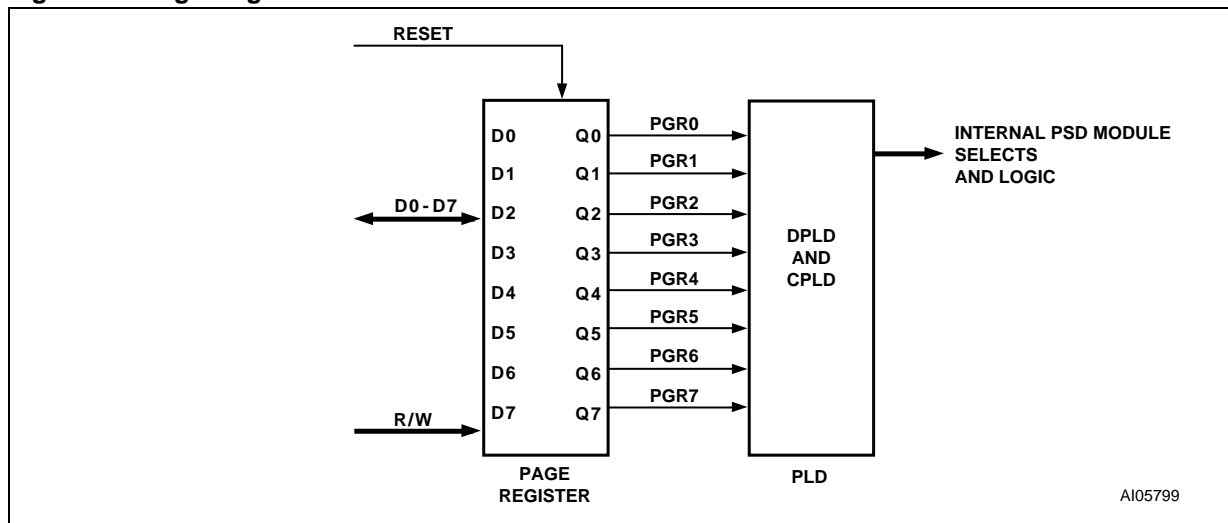
### Page Register

The 8-bit Page Register increases the addressing capability of the MCU Core by a factor of up to 256. The contents of the register can also be read by the MCU. The outputs of the Page Register (PGR0-PGR7) are inputs to the DPLD decoder and can be included in the Sector Select (FS0-FS7, CSBOOT0-CSBOOT3), and SRAM Select (RS0) equations.

If memory paging is not needed, or if not all 8 page register bits are needed for memory paging, then these bits may be used in the CPLD for general logic.

Figure 33 shows the Page Register. The eight flip-flops in the register are connected to the internal data bus D0-D7. The MCU can write to or read from the Page Register. The Page Register can be accessed at address location CSIOP + E0h.

**Figure 33. Page Register**



**PLDS**

The PLDs bring programmable logic functionality to the μPSD. After specifying the logic for the PLDs in PSDsoft Express, the logic is programmed into the device and available upon Power-up.

The PSD Module contains two PLDs: the Decode PLD (DPLD), and the Complex PLD (CPLD).

The DPLD performs address decoding for Select signals for PSD Module components, such as memory, registers, and I/O ports.

The CPLD can be used for logic functions, such as loadable counters and shift registers, state machines, and encoding and decoding logic.

**The Turbo Bit in PSD Module**

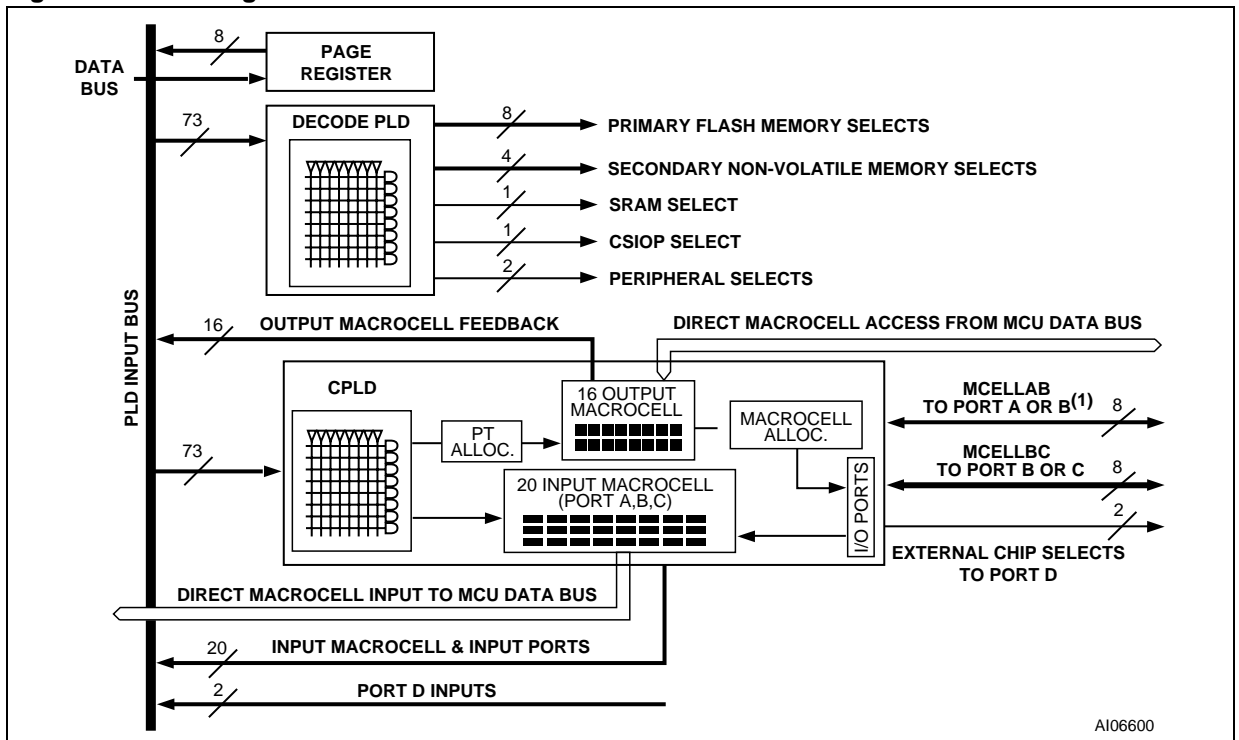
The PLDs can minimize power consumption by switching off when inputs remain unchanged for an extended time of about 70ns. Resetting the Turbo Bit to '0' (Bit 3 of PMMR0) automatically places the PLDs into standby if no inputs are changing. Turning the Turbo Mode off increases propagation delays while reducing power consumption. See POWER MANAGEMENT, page 90 for details on setting the Turbo Bit.

**Table 84. DPLD and CPLD Inputs**

Input Source	Input Name	No. of Signals
MCU Address Bus	A15-A0	16
MCU Control Signals	$\overline{\text{PSEN}}$ , $\overline{\text{RD}}$ , $\overline{\text{WR}}$ , ALE	4
RESET	$\overline{\text{RST}}$	1
Power-down	PDN	1
Port A Input Macrocells <sup>(1)</sup>	PA7-PA0	8
Port B Input Macrocells	PB7-PB0	8
Port C Input Macrocells	PC2-4, PC7	4
Port D Inputs	PD2-PD1	2
Page Register	PGR7-PGR0	8
Macrocell AB Feedback	MCELLAB.FB7-FB0	8
Macrocell BC Feedback	MCELLBC.FB7-FB0	8
Flash memory Program Status Bit	Ready/ $\overline{\text{Busy}}$	1

Note: 1. These inputs are not available in the 52-pin package.

**Figure 34. PLD Diagram**



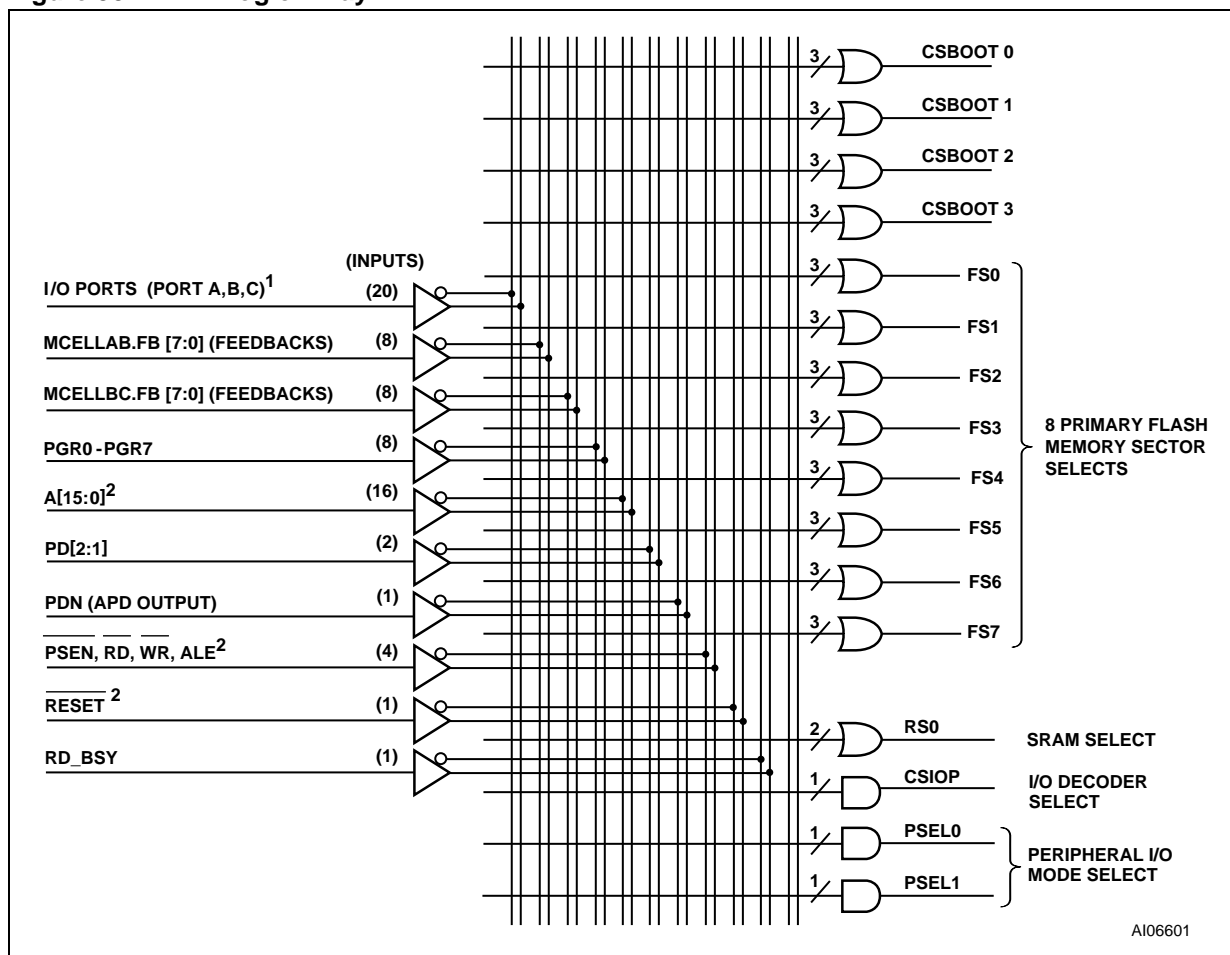
Note: 1. Ports A is not available in the 52-pin package

**Decode PLD (DPLD)**

The DPLD, shown in Figure 35, is used for decoding the address for PSD Module and external components. The DPLD can be used to generate the following decode signals:

- Up to 8 Sector Select (FS0-FS7) signals for the primary Flash memory (three product terms each)
- Up to 4 Sector Select (CSBOOT0-CSBOOT3) signals for the secondary Flash memory (three product terms each)
- 1 internal SRAM Select (RS0) signal (two product terms)
- 1 internal CSIOP Select signal (selects the PSD Module registers)
- 2 internal Peripheral Select signals (Peripheral I/O Mode).

**Figure 35. DPLD Logic Array**



Note: 1. Port A inputs are not available in the 52-pin package  
 2. Inputs from the MCU module

**Complex PLD (CPLD)**

The CPLD can be used to implement system logic functions, such as loadable counters and shift registers, system mailboxes, handshaking protocols, state machines, and random logic. The CPLD can also be used to generate External Chip Select (ECS1-ECS2), routed to Port D.

As shown in Figure 34, the CPLD has the following blocks:

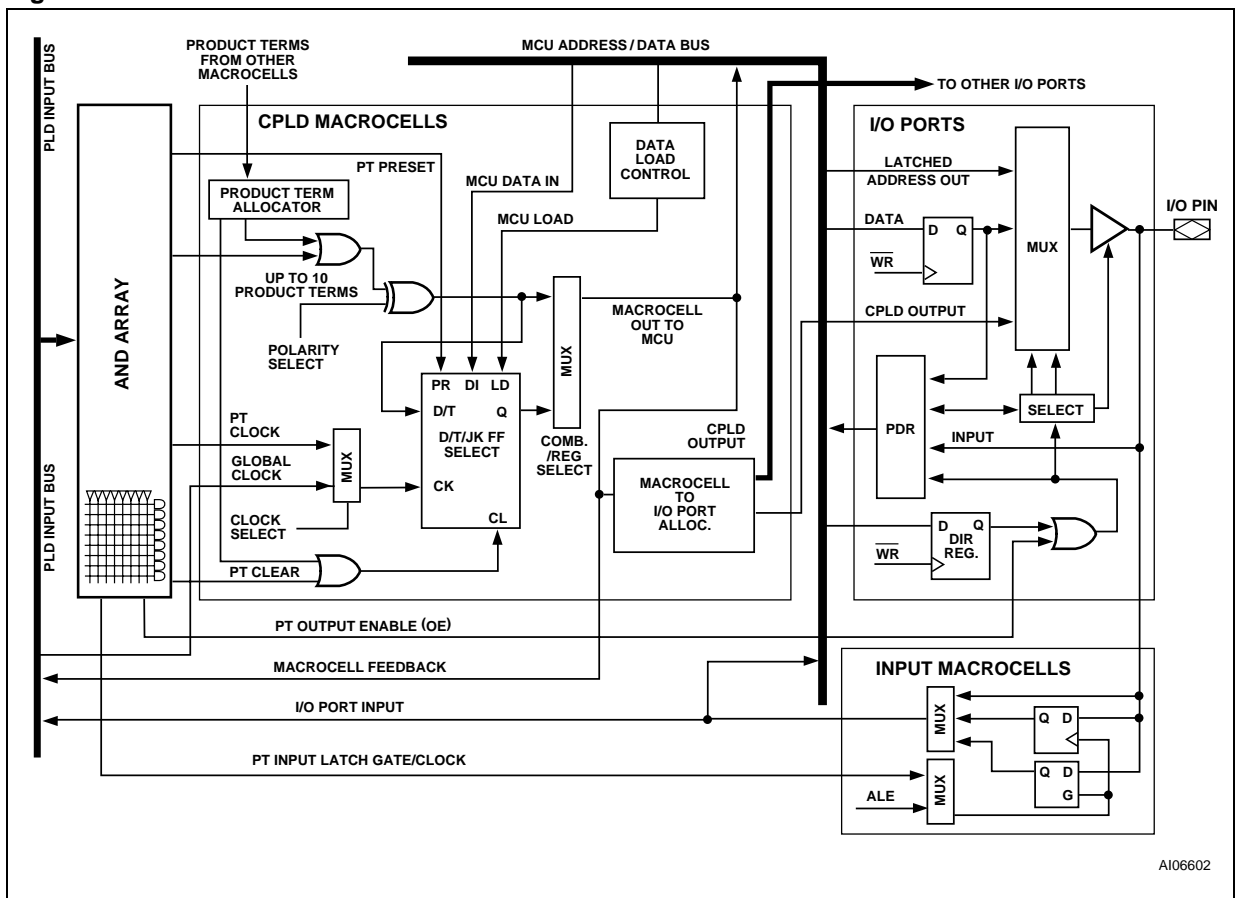
- 20 Input Macrocells (IMC)
- 16 Output Macrocells (OMC)
- Macrocell Allocator
- Product Term Allocator
- AND Array capable of generating up to 137 product terms
- Four I/O Ports.

Each of the blocks are described in the sections that follow.

The Input Macrocells (IMC) and Output Macrocells (OMC) are connected to the PSD Module internal data bus and can be directly accessed by the MCU. This enables the MCU software to load data into the Output Macrocells (OMC) or read data from both the Input and Output Macrocells (IMC and OMC).

This feature allows efficient implementation of system logic and eliminates the need to connect the data bus to the AND Array as required in most standard PLD macrocell architectures.

**Figure 36. Macrocell and I/O Port**





### Output Macrocell (OMC)

Eight of the Output Macrocells (OMC) are connected to Ports A and B pins and are named as McellAB0-McellAB7. The other eight macrocells are connected to Ports B and C pins and are named as McellBC0-McellBC7. If an McellAB output is not assigned to a specific pin in PSDsoft, the Macrocell Allocator block assigns it to either Port A or B. The same is true for a McellBC output on Port B or C. Table 85 shows the macrocells and port assignment.

The Output Macrocell (OMC) architecture is shown in Figure 37.

The flip-flop in the Output Macrocell (OMC) block can be configured as a D, T, JK, or SR type in PS-Dsoft. The flip-flop's clock, preset, and clear inputs may be driven from a product term of the AND Array. Alternatively, CLKIN (PD1) can be used for the clock input to the flip-flop. The flip-flop is clocked on the rising edge of CLKIN (PD1). The preset and clear are active High inputs. Each clear input can use up to two product terms.

**Table 85. Output Macrocell Port and Data Bit Assignments**

Output Macrocell	Port Assignment <sup>(1,2)</sup>	Native Product Terms	Maximum Borrowed Product Terms	Data Bit for Loading or Reading
McellAB0	Port A0, B0	3	6	D0
McellAB1	Port A1, B1	3	6	D1
McellAB2	Port A2, B2	3	6	D2
McellAB3	Port A3, B3	3	6	D3
McellAB4	Port A4, B4	3	6	D4
McellAB5	Port A5, B5	3	6	D5
McellAB6	Port A6, B6	3	6	D6
McellAB7	Port A7, B7	3	6	D7
McellBC0	Port B0	4	5	D0
McellBC1	Port B1	4	5	D1
McellBC2	Port B2, C2	4	5	D2
McellBC3	Port B3, C3	4	5	D3
McellBC4	Port B4, C4	4	6	D4
McellBC5	Port B5	4	6	D5
McellBC6	Port B6	4	6	D6
McellBC7	Port B7, C7	4	6	D7

Note: 1. McellAB0-McellAB7 can only be assigned to Port B in the 52-pin package.

2. Port PC0, PC1, PC5, and PC6 are assigned to JTAG pins and are not available as Macrocell outputs.



**The OMC Mask Register.** There is one Mask Register for each of the two groups of eight Output Macrocells (OMC). The Mask Registers can be used to block the loading of data to individual Output Macrocells (OMC). The Mask Registers can be used to block the loading of data to individual Output Macrocells (OMC). The default value for the Mask Registers is 00h, which allows loading of the Output Macrocells (OMC). When a given bit in a Mask Register is set to a '1,' the MCU is blocked from writing to the associated Output Macrocells (OMC). For example, suppose McellAB0-McellAB3 are being used for a state machine. You would not want a MCU write to McellAB to overwrite the state machine registers. Therefore, you would want to load the Mask Register for McellAB (Mask Macrocell AB) with the value 0Fh.

**The Output Enable of the OMC.** The Output Macrocells (OMC) block can be connected to an I/O port pin as a PLD output. The output enable of each port pin driver is controlled by a single product term from the AND Array, ORed with the Direction Register output. The pin is enabled upon Power-up if no output enable equation is defined and if the pin is declared as a PLD output in PSDsoft Express.

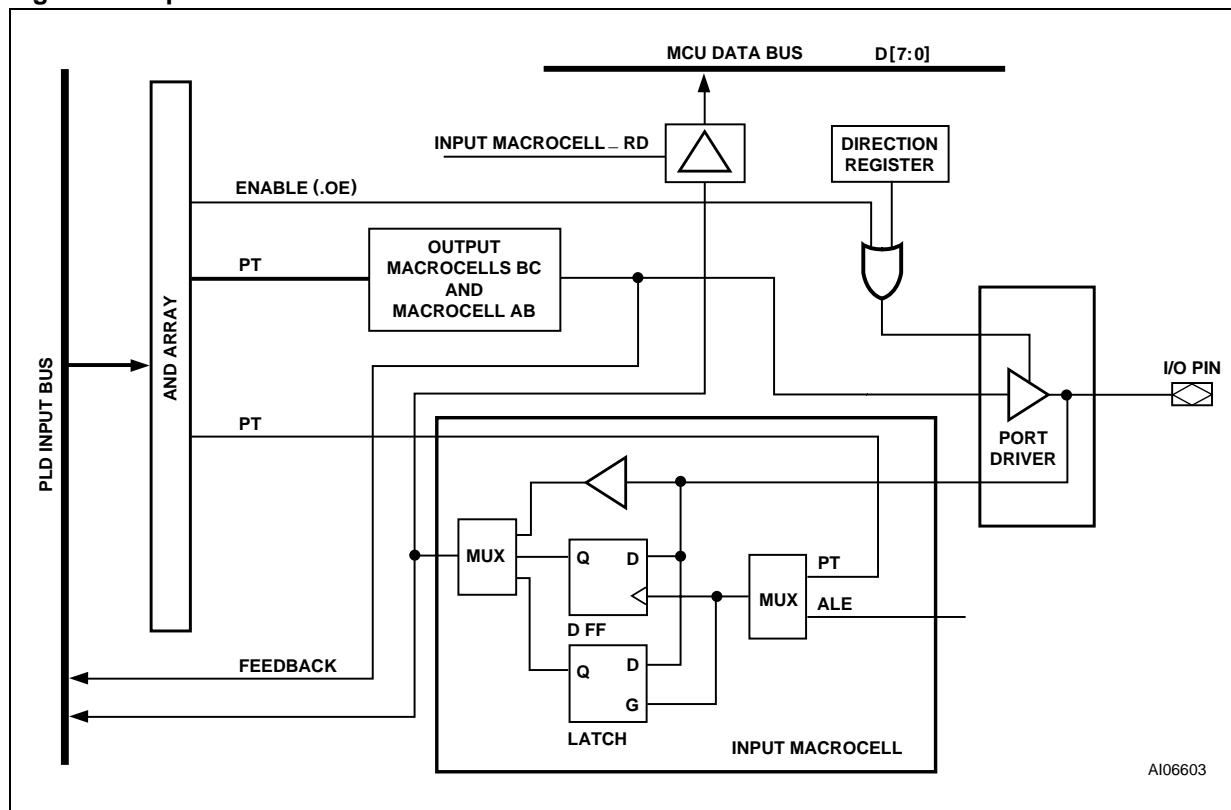
If the Output Macrocell (OMC) output is declared as an internal node and not as a port pin output in the PSDlabel file, the port pin can be used for other I/O functions. The internal node feedback can be routed as an input to the AND Array.

**Input Macrocells (IMC)**

The CPLD has 20 Input Macrocells (IMC), one for each pin on Ports A and B, and 4 on Port C. The architecture of the Input Macrocells (IMC) is shown in Figure 38. The Input Macrocells (IMC) are individually configurable, and can be used as a latch, register, or to pass incoming Port signals prior to driving them onto the PLD input bus. The outputs of the Input Macrocells (IMC) can be read by the MCU through the internal data bus.

The enable for the latch and clock for the register are driven by a multiplexer whose inputs are a product term from the CPLD AND Array or the MCU Address Strobe (ALE). Each product term output is used to latch or clock four Input Macrocells (IMC). Port inputs 3-0 can be controlled by one product term and 7-4 by another.

Figure 38. Input Macrocell



A106603

**I/O PORTS (PSD MODULE)**

There are four programmable I/O ports: Ports A, B, C, and D in the PSD Module. Each of the ports is eight bits except Port D. Each port pin is individually user configurable, thus allowing multiple functions per port. The ports are configured using PSDsoft Express Configuration or by the MCU writing to on-chip registers in the CSIOP space. Port A is not available in the 52-pin package.

**General Port Architecture**

The general architecture of the I/O Port block is shown in Figure 39. In general, once the purpose for a port pin has been defined, that pin is no longer available for other purposes. Exceptions are noted.

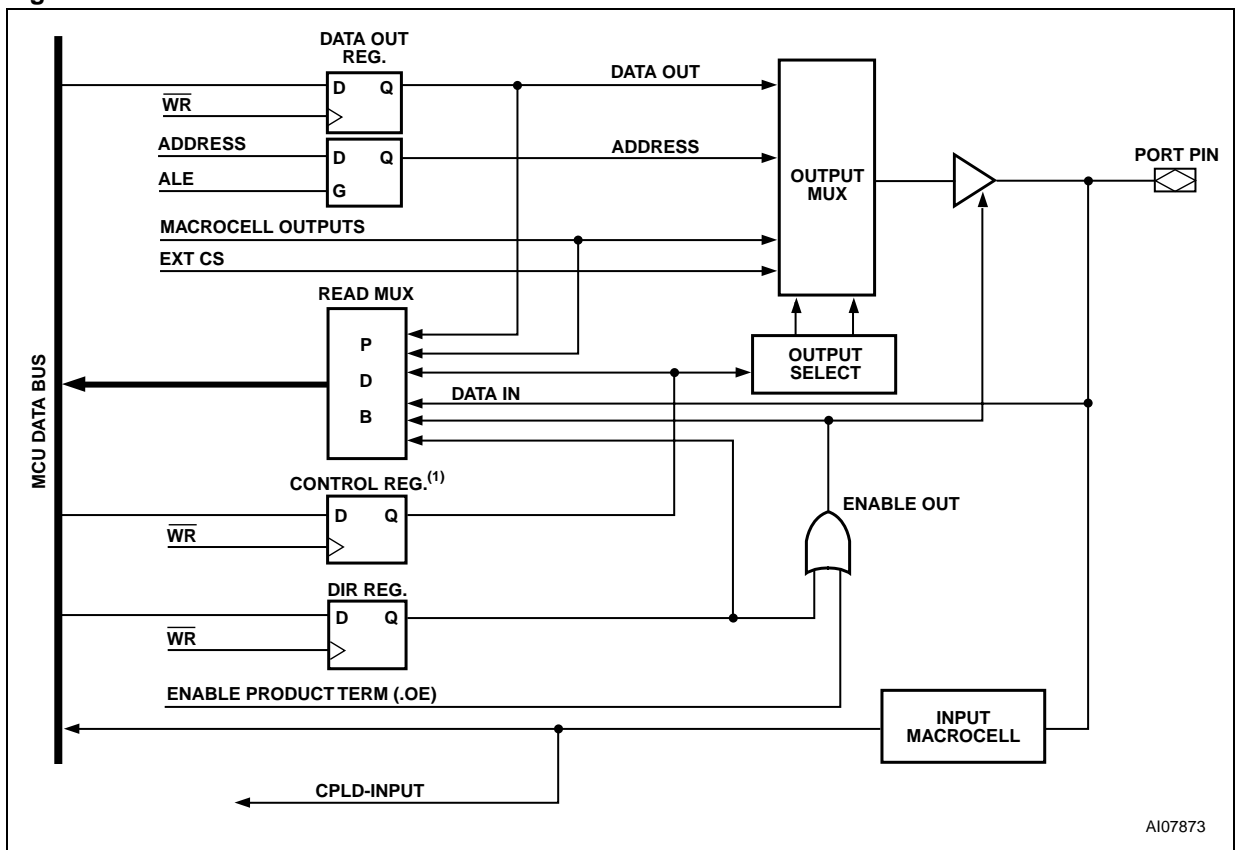
As shown in Figure 39, the ports contain an output multiplexer whose select signals are driven by the configuration bits in the Control Registers (Ports A

and B only) and PSDsoft Express Configuration. Inputs to the multiplexer include the following:

- Output data from the Data Out register
- Latched address outputs
- CPLD macrocell output
- External Chip Select (ECS1-ECS2) from the CPLD.

The Port Data Buffer (PDB) is a tri-state buffer that allows only one source at a time to be read. The Port Data Buffer (PDB) is connected to the Internal Data Bus for feedback and can be read by the MCU. The Data Out and macrocell outputs, Direction and Control Registers, and port pin input are all connected to the Port Data Buffer (PDB).

**Figure 39. General I/O Port Architecture**



Note: 1. Control Register is not available in Ports C and D.

The Port pin's tri-state output driver enable is controlled by a two input OR gate whose inputs come from the CPLD AND Array enable product term and the Direction Register. If the enable product term of any of the Array outputs are not defined and that port pin is not defined as a CPLD output in the PSDsoft, then the Direction Register has sole control of the buffer that drives the port pin.

The contents of these registers can be altered by the MCU. The Port Data Buffer (PDB) feedback path allows the MCU to check the contents of the registers.

Ports A, B, and C have embedded Input Macrocells (IMC). The Input Macrocells (IMC) can be configured as latches, registers, or direct inputs to the PLDs. The latches and registers are clocked by Address Strobe (ALE) or a product term from the PLD AND Array. The outputs from the Input Macrocells (IMC) drive the PLD input bus and can be read by the MCU. See Input Macrocell, page 83.

**Port Operating Modes**

The I/O Ports have several modes of operation. Some modes can be defined using PSDsoft, some by the MCU writing to the Control Registers in CSIOP space, and some by both. The modes that can only be defined using PSDsoft must be programmed into the device and cannot be changed unless the device is reprogrammed. The modes that can be changed by the MCU can be done so dynamically at run-time. The PLD I/O, Data Port, Address Input, and Peripheral I/O Modes are the only modes that must be defined before programming the device. All other modes can be changed by the MCU at run-time. See Application Note AN1171 for more detail.

Table 86 summarizes which modes are available on each port. Table 89 shows how and where the different modes are configured. Each of the port operating modes are described in the following sections.

**MCU I/O Mode**

In the MCU I/O Mode, the MCU uses the I/O Ports block to expand its own I/O ports. By setting up the CSIOP space, the ports on the PSD Module are mapped into the MCU address space. The addresses of the ports are listed in Table 77, page 67.

A port pin can be put into MCU I/O Mode by writing a '0' to the corresponding bit in the Control Register. The MCU I/O direction may be changed by writing to the corresponding bit in the Direction Register, or by the output enable product term (see Peripheral I/O Mode, page 85). When the pin

is configured as an output, the content of the Data Out Register drives the pin. When configured as an input, the MCU can read the port input through the Data In buffer. See Figure 39, page 84.

Ports C and D do not have Control Registers, and are in MCU I/O Mode by default. They can be used for PLD I/O if equations are written for them in PSDsoft.

**PLD I/O Mode**

The PLD I/O Mode uses a port as an input to the CPLD's Input Macrocells (IMC), and/or as an output from the CPLD's Output Macrocells (OMC). The output can be tri-stated with a control signal. This output enable control signal can be defined by a product term from the PLD, or by resetting the corresponding bit in the Direction Register to '0.' The corresponding bit in the Direction Register must not be set to '1' if the pin is defined for a PLD input signal in PSDsoft. The PLD I/O Mode is specified in PSDsoft by declaring the port pins, and then writing an equation assigning the PLD I/O to a port.

**Address Out Mode**

Address Out Mode can be used to drive latched MCU addresses on to the port pins. These port pins can, in turn, drive external devices. Either the output enable or the corresponding bits of both the Direction Register and Control Register must be set to a '1' for pins to use Address Out Mode. This must be done by the MCU at run-time. See Table 88 for the address output pin assignments on Ports A and B for various MCUs.

**Peripheral I/O Mode**

Peripheral I/O Mode can be used to interface with external peripherals. In this mode, all of Port A serves as a tri-state, bi-directional data buffer for the MCU. Peripheral I/O Mode is enabled by setting Bit 7 of the VM Register to a '1.' Figure 40 shows how Port A acts as a bi-directional buffer for the MCU data bus if Peripheral I/O Mode is enabled. An equation for PSEL0 and/or PSEL1 must be written in PSDsoft. The buffer is tri-stated when PSEL0 or PSEL1 is low (not active). The PSEN signal should be "ANDed" in the PSEL equations to disable the buffer when PSEL resides in the data space.

**JTAG In-System Programming (ISP)**

Port C is JTAG compliant, and can be used for In-System Programming (ISP). For more information on the JTAG Port, see PROGRAMMING IN-CIRCUIT USING THE JTAG SERIAL INTERFACE, page 95.

Figure 40. Peripheral I/O Mode

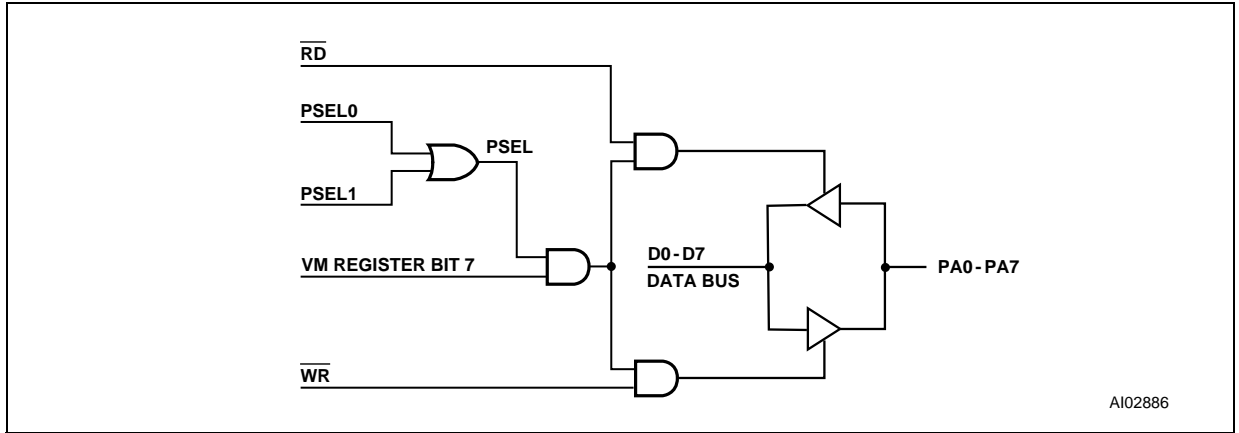


Table 86. Port Operating Modes

Port Mode	Port A <sup>(1)</sup>	Port B	Port C	Port D
MCU I/O	Yes	Yes	Yes	Yes
PLD I/O				
McellAB Outputs	Yes	Yes	No	No
McellBC Outputs	No	Yes	Yes <sup>(2)</sup>	No
Additional Ext. CS Outputs	No	No	No	Yes
PLD Inputs	Yes	Yes	Yes	Yes
Address Out	Yes (A7 – 0)	Yes (A7 – 0)	No	No
Peripheral I/O	Yes	No	No	No
JTAG ISP	No	No	Yes <sup>(3)</sup>	No

Note: 1. Port A is not available in the 52-pin package.  
 2. On pins PC2, PC3, PC4, and PC7 only.  
 3. JTAG pins (TMS, TCK, TDI, TDO) are dedicated pins.

Table 87. Port Operating Mode Settings

Mode	Defined in PSDsoft	Control Register Setting <sup>(1)</sup>	Direction Register Setting <sup>(1)</sup>	VM Register Setting <sup>(1)</sup>
MCU I/O	Declare pins only	0	1 = output, 0 = input (Note 2)	N/A
PLD I/O	Logic equations	N/A	(Note 2)	N/A
Address Out (Port A,B)	Declare pins only	1	1 (Note 2)	N/A
Peripheral I/O (Port A)	Logic equations (PSEL0 & 1)	N/A	N/A	PIO Bit = 1

Note: 1. N/A = Not Applicable  
 2. The direction of the Port A,B,C, and D pins are controlled by the Direction Register ORed with the individual output enable product term (.oe) from the CPLD AND Array.

Table 88. I/O Port Latched Address Output Assignments

Port A (PA3-PA0)	Port A (PA7-PA4)	Port B (PB3-PB0)	Port B (PB7-PB4)
Address a3-a0	Address a7-a4	Address a3-a0	Address a7-a4

### Port Configuration Registers (PCR)

Each Port has a set of Port Configuration Registers (PCR) used for configuration. The contents of the registers can be accessed by the MCU through normal READ/WRITE bus cycles at the addresses given in Table 77, page 67. The addresses in Table 79, page 69 are the offsets in hexadecimal from the base of the CSIOP register.

The pins of a port are individually configurable and each bit in the register controls its respective pin. For example, Bit 0 in a register refers to Bit 0 of its port. The three Port Configuration Registers (PCR), shown in Table 89, are used for setting the Port configurations. The default Power-up state for each register in Table 89 is 00h.

**Control Register.** Any bit reset to '0' in the Control Register sets the corresponding port pin to MCU I/O Mode, and a '1' sets it to Address Out Mode. The default mode is MCU I/O. Only Ports A and B have an associated Control Register.

**Direction Register.** The Direction Register, in conjunction with the output enable (except for Port D), controls the direction of data flow in the I/O Ports. Any bit set to '1' in the Direction Register causes the corresponding pin to be an output, and any bit set to '0' causes it to be an input. The default mode for all port pins is input.

Figure 39, page 84 shows the Port Architecture diagrams for Ports A/B and C, respectively. The direction of data flow for Ports A, B, and C are controlled not only by the direction register, but also by the output enable product term from the PLD AND Array. If the output enable product term is not active, the Direction Register has sole control of a given pin's direction.

An example of a configuration for a Port with the three least significant bits set to output and the remainder set to input is shown in Table 92. Since Port D only contains two pins, the Direction Register for Port D has only two bits active.

**Drive Select Register.** The Drive Select Register configures the pin driver as Open Drain or CMOS for some port pins, and controls the slew rate for the other port pins. An external pull-up resistor should be used for pins configured as Open Drain.

A pin can be configured as Open Drain if its corresponding bit in the Drive Select Register is set to a '1.' The default pin drive is CMOS.

**Note:** The slew rate is a measurement of the rise and fall times of an output. A higher slew rate means a faster output response and may create more electrical noise. A pin operates in a high slew rate when the corresponding bit in the Drive Register is set to '1.' The default rate is slow slew.

Table 93, page 88 shows the Drive Register for Ports A, B, C, and D. It summarizes which pins can be configured as Open Drain outputs and which pins the slew rate can be set for.

**Table 89. Port Configuration Registers (PCR)**

Register Name	Port	MCU Access
Control	A,B	WRITE/READ
Direction	A,B,C,D	WRITE/READ
Drive Select <sup>(1)</sup>	A,B,C,D	WRITE/READ

Note: 1. See Table 93 for Drive Register Bit definition.

**Table 90. Port Pin Direction Control, Output Enable P.T. Not Defined**

Direction Register Bit	Port Pin Mode
0	Input
1	Output

**Table 91. Port Pin Direction Control, Output Enable P.T. Defined**

Direction Register Bit	Output Enable P.T.	Port Pin Mode
0	0	Input
0	1	Output
1	0	Output
1	1	Output

**Table 92. Port Direction Assignment Example**

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	1	1	1

**Port Data Registers**

The Port Data Registers, shown in Table 94, are used by the MCU to write data to or read data from the ports. Table 94 shows the register name, the ports having each register type, and MCU access for each register type. The registers are described below.

**Data In.** Port pins are connected directly to the Data In buffer. In MCU I/O Input Mode, the pin input is read through the Data In buffer.

**Data Out Register.** Stores output data written by the MCU in the MCU I/O Output Mode. The contents of the Register are driven out to the pins if the Direction Register or the output enable product term is set to '1.' The contents of the register can also be read back by the MCU.

**Output Macrocells (OMC).** The CPLD Output Macrocells (OMC) occupy a location in the MCU's address space. The MCU can read the output of the Output Macrocells (OMC). If the OMC Mask

Register Bits are not set, writing to the macrocell loads data to the macrocell flip-flops. See PLDs, page 78.

**OMC Mask Register.** Each OMC Mask Register Bit corresponds to an Output Macrocell (OMC) flip-flop. When the OMC Mask Register Bit is set to a '1,' loading data into the Output Macrocell (OMC) flip-flop is blocked. The default value is '0' or unblocked.

**Input Macrocells (IMC).** The Input Macrocells (IMC) can be used to latch or store external inputs. The outputs of the Input Macrocells (IMC) are routed to the PLD input bus, and can be read by the MCU. See PLDs, page 78.

**Enable Out.** The Enable Out register can be read by the MCU. It contains the output enable values for a given port. A '1' indicates the driver is in output mode. A '0' indicates the driver is in tri-state and the pin is in input mode.

**Table 93. Drive Register Pin Assignment**

Drive Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Port A	Open Drain	Open Drain	Open Drain	Open Drain	Slew Rate	Slew Rate	Slew Rate	Slew Rate
Port B	Open Drain	Open Drain	Open Drain	Open Drain	Slew Rate	Slew Rate	Slew Rate	Slew Rate
Port C	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain
Port D	NA <sup>(1)</sup>	NA <sup>(1)</sup>	NA <sup>(1)</sup>	NA <sup>(1)</sup>	NA <sup>(1)</sup>	Slew Rate	Slew Rate	NA <sup>(1)</sup>

Note: 1. NA = Not Applicable.

**Table 94. Port Data Registers**

Register Name	Port	MCU Access
Data In	A,B,C,D	READ – input on pin
Data Out	A,B,C,D	WRITE/READ
Output Macrocell	A,B,C	READ – outputs of macrocells WRITE – loading macrocells flip-flop
Mask Macrocell	A,B,C	WRITE/READ – prevents loading into a given macrocell
Input Macrocell	A,B,C	READ – outputs of the Input Macrocells
Enable Out	A,B,C	READ – the output enable control of the port driver



### Ports A and B – Functionality and Structure

Ports A and B have similar functionality and structure, as shown in Figure 39, page 84. The two ports can be configured to perform one or more of the following functions:

- MCU I/O Mode
- CPLD Output – Macrocells McellAB7-McellAB0 can be connected to Port A or Port B. McellBC7-McellBC0 can be connected to Port B or Port C.
- CPLD Input – Via the Input Macrocells (IMC).
- Latched Address output – Provide latched address output as per Table 88.
- Open Drain/Slew Rate – pins PA3-PA0 and PB3-PB0 can be configured to fast slew rate, pins PA7-PA4 and PB7-PB4 can be configured to Open Drain Mode.
- Peripheral Mode – Port A only (80-pin package)

### Port C – Functionality and Structure

Port C can be configured to perform one or more of the following functions (see Figure 39, page 84):

- MCU I/O Mode
- CPLD Output – McellBC[2, 3, 4, 7] outputs can be connected to Port C.
- CPLD Input – via the Input Macrocells (IMC) on pins PC2, PC3, PC4, and PC7.
- In-System Programming (ISP) – Port pins PC0, PC1, PC5, and PC6 are dedicated for JTAG ISP programming (TMS, TCK, TDI, TDO, see PROGRAMMING IN-CIRCUIT USING THE JTAG SERIAL INTERFACE, page 95, for more information on JTAG programming.)

- Open Drain – Port C pins can be configured in Open Drain Mode
- Battery Backup features – PC2 can be configured for a battery input supply, Voltage Stand-by ( $V_{STBY}$ ).  
PC4 can be configured as a Battery-on Indicator ( $V_{BATON}$ ), indicating when  $V_{CC}$  is less than  $V_{BAT}$ .

### Port D – Functionality and Structure

Port D has two I/O pins (only one pin, PD1, in the 52-pin package). This port does not support Address Out Mode, and therefore no Control Register is required. Of the eight bits in the Port D registers, only Bits 2 and 1 are used to configure pins PD2 and PD1.

Port D can be configured to perform one or more of the following functions:

- MCU I/O Mode;
  - CPLD Output – External Chip Select (ECS1-ECS2), each ECS consists of one product term that can be configured active high or low;
  - CPLD Input – direct input to the CPLD, no Input Macrocells (IMC); and
  - Slew rate – pins can be set up for fast slew rate
- Port D pins can be configured in PSDsoft Express as input pins for other dedicated functions:
- CLKIN (PD1) as input to the macrocells flip-flops and APD counter, and
  - PSD Chip Select Input ( $\overline{CSI}$ , PD2). Driving this signal High disables the Flash memory, SRAM and CSIOP.

**POWER MANAGEMENT**

All PSD Module offers configurable power saving options. These options may be used individually or in combinations, as follows:

- The primary and secondary Flash memory, and SRAM blocks are built with power management technology. In addition to using special silicon design methodology, power management technology puts the memories into Standby Mode when address/data inputs are not changing (zero DC current). As soon as a transition occurs on an input, the affected memory “wakes up,” changes and latches its outputs, then goes back to standby. The designer does *not* have to do anything special to achieve Memory Standby Mode when no inputs are changing—it happens automatically.

The PLD sections can also achieve Standby Mode when its inputs are not changing, as described in the sections on the Power Management Mode Registers (PMMR).

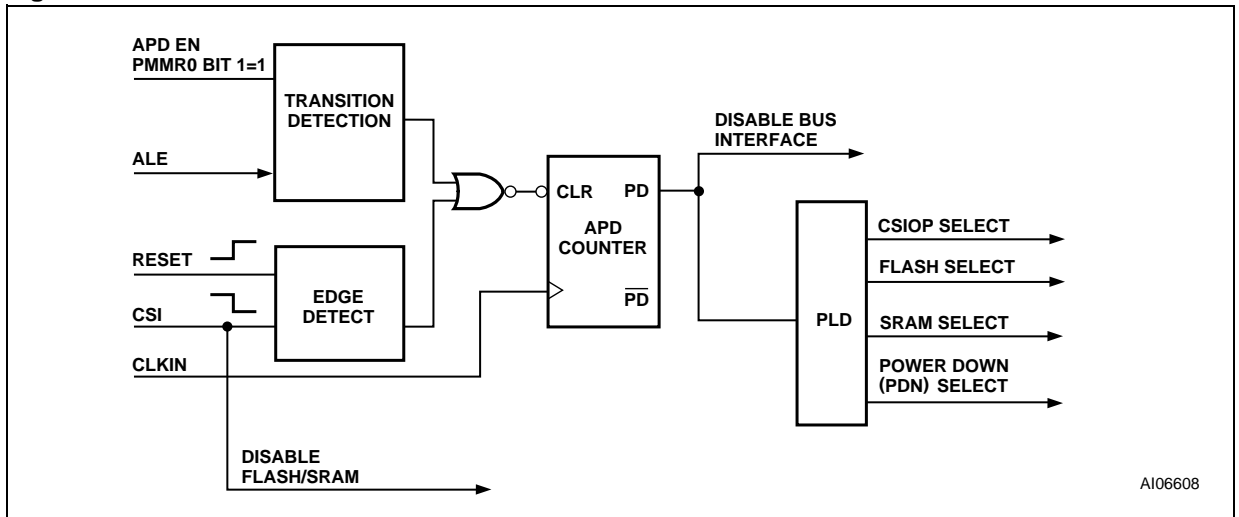
- As with the Power Management Mode, the Automatic Power Down (APD) block allows the PSD Module to reduce to stand-by current automatically. The APD Unit can also block MCU address/data signals from reaching the memories and PLDs.

Built in logic monitors the Address Strobe of the MCU for activity. If there is no activity for a certain time period (MCU is asleep), the APD Unit initiates Power-down Mode (if enabled).

Once in Power-down Mode, all address/data signals are blocked from reaching memory and PLDs, and the memories are deselected internally. This allows the memory and PLDs to remain in Standby Mode even if the address/data signals are changing state externally (noise, other devices on the MCU bus, etc.). Keep in mind that any unblocked PLD input signals that are changing states keeps the PLD out of Stand-by Mode, but not the memories.

- PSD Chip Select Input ( $\overline{CSI}$ , PD2) can be used to disable the internal memories, placing them in Standby Mode even if inputs are changing. This feature does not block any internal signals or disable the PLDs. This is a good alternative to using the APD Unit. There is a slight penalty in memory access time when PSD Chip Select Input ( $\overline{CSI}$ , PD2) makes its initial transition from deselected to selected.
- The PMMRs can be written by the MCU at runtime to manage power. The PSD Module supports “blocking bits” in these registers that are set to block designated signals from reaching both PLDs. Current consumption of the PLDs is directly related to the composite frequency of the changes on their inputs (see Figure 44 and Figure 45). Significant power savings can be achieved by blocking signals that are not used in DPLD or CPLD logic equations.

**Figure 41. APD Unit**



The PSD Module has a Turbo Bit in PMMR0. This bit can be set to turn the Turbo Mode off (the default is with Turbo Mode turned on). While Turbo Mode is off, the PLDs can achieve standby current when no PLD inputs are changing (zero DC current). Even when inputs do change, significant power can be saved at lower frequencies (AC current), compared to when Turbo Mode is on. When the Turbo Mode is on, there is a significant DC current component and the AC component is higher.

**Automatic Power-down (APD) Unit and Power-down Mode.** The APD Unit, shown in Figure 41, puts the PSD Module into Power-down Mode by monitoring the activity of Address Strobe (ALE). If the APD Unit is enabled, as soon as activity on Address Strobe (ALE) stops, a four-bit counter starts counting. If Address Strobe (ALE/AS, PD0) remains inactive for fifteen clock periods of CLKIN (PD1), Power-down (PDN) goes High, and the PSD Module enters Power-down Mode, as discussed next.

**Power-down Mode.** By default, if you enable the APD Unit, Power-down Mode is automatically enabled. The device enters Power-down Mode if Address Strobe (ALE) remains inactive for fifteen periods of CLKIN (PD1).

The following should be kept in mind when the PSD Module is in Power-down Mode:

- If Address Strobe (ALE) starts pulsing again, the PSD Module returns to normal Operating mode. The PSD Module also returns to normal Operating mode if either PSD Chip Select Input (CSI, PD2) is Low or the RESET input is High.
- The MCU address/data bus is blocked from all memory and PLDs.
- Various signals can be blocked (prior to Power-down Mode) from entering the PLDs by setting the appropriate bits in the PMMR registers. The blocked signals include MCU control signals and the common CLKIN (PD1).

**Note:** Blocking CLKIN (PD1) from the PLDs does not block CLKIN (PD1) from the APD Unit.

- All memories enter Standby Mode and are drawing standby current. However, the PLD and I/O ports blocks do *not* go into Standby Mode because you don't want to have to wait for the logic and I/O to "wake-up" before their outputs can change. See Table 95 for Power-down Mode effects on PSD Module ports.

- Typical standby current is of the order of microamperes. These standby current values assume that there are no transitions on any PLD input.

**Other Power Saving Options.** The PSD Module offers other reduced power saving options that are independent of the Power-down Mode. Except for the SRAM Stand-by and PSD Chip Select Input (CSI, PD2) features, they are enabled by setting bits in PMMR0 and PMMR2.

Figure 42. Enable Power-down Flow Chart

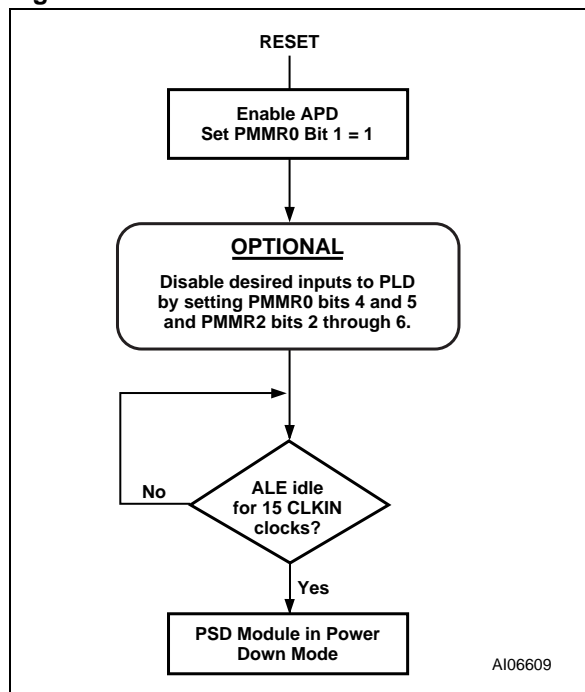


Table 95. Power-down Mode's Effect on Ports

Port Function	Pin Level
MCU I/O	No Change
PLD Out	No Change
Address Out	Undefined
Peripheral I/O	Tri-State

**PLD Power Management**

The power and speed of the PLDs are controlled by the Turbo Bit (Bit 3) in PMMR0. By setting the bit to '1,' the Turbo Mode is off and the PLDs consume the specified stand-by current when the inputs are not switching for an extended time of 70ns. The propagation delay time is increased by 10ns (for a 5V device) after the Turbo Bit is set to '1' (turned off) when the inputs change at a composite frequency of less than 15MHz. When the Turbo Bit is reset to '0' (turned on), the PLDs run at full power and speed. The Turbo Bit affects the PLD's DC power, AC power, and propagation delay. When the Turbo Mode is off, the μPSD3200 input clock frequency is reduced by 5MHz from the maximum rated clock frequency.

Blocking MCU control signals with the bits of PMMR2 can further reduce PLD AC power consumption.

**SRAM Standby Mode (Battery Backup).** The SRAM in the PSD Module supports a battery backup mode in which the contents are retained in the event of a power loss. The SRAM has Voltage Stand-by (V<sub>STBY</sub>, PC2) that can be connected to an external battery. When V<sub>CC</sub> becomes lower than V<sub>STBY</sub> then the SRAM automatically connects to Voltage Stand-by (V<sub>STBY</sub>, PC2) as a power source. The SRAM Standby Current (I<sub>STBY</sub>) is typically 0.5 μA. The SRAM data retention voltage is 2V minimum. The Battery-on Indicator (VBATON) can be routed to PC4. This signal indicates when the V<sub>CC</sub> has dropped below V<sub>STBY</sub>.

**PSD Chip Select Input (CSI, PD2)**

PD2 of Port D can be configured in PSDsoft Express as PSD Chip Select Input (CSI). When Low, the signal selects and enables the PSD Module Flash memory, SRAM, and I/O blocks for READ or WRITE operations. A High on PSD Chip Select Input (CSI, PD2) disables the Flash memory, and SRAM, and reduces power consumption. However, the PLD and I/O signals remain operational when PSD Chip Select Input (CSI, PD2) is High.

**Input Clock**

CLKIN (PD1) can be turned off, to the PLD to save AC power consumption. CLKIN (PD1) is an input to the PLD AND Array and the Output Macrocells (OMC).

During Power-down Mode, or, if CLKIN (PD1) is not being used as part of the PLD logic equation, the clock should be disabled to save AC power. CLKIN (PD1) is disconnected from the PLD AND Array or the Macrocells block by setting Bits 4 or 5 to a '1' in PMMR0.

**Input Control Signals**

The PSD Module provides the option to turn off the MCU signals (WR, RD, PSEN, and Address Strobe (ALE)) to the PLD to save AC power consumption. These control signals are inputs to the PLD AND Array. During Power-down Mode, or, if any of them are not being used as part of the PLD logic equation, these control signals should be disabled to save AC power. They are disconnected from the PLD AND Array by setting Bits 2, 3, 4, 5, and 6 to a '1' in PMMR2.

**Table 96. Power Management Mode Registers PMMR0**

Bit 0	X	0	Not used, and should be set to zero.
Bit 1	APD Enable	0 = off	Automatic Power-down (APD) is disabled.
		1 = on	Automatic Power-down (APD) is enabled.
Bit 2	X	0	Not used, and should be set to zero.
Bit 3	PLD Turbo	0 = on	PLD Turbo Mode is on
		1 = off	PLD Turbo Mode is off, saving power. μPSD3200 operates at 5MHz below the maximum rated clock frequency
Bit 4	PLD Array clk	0 = on	CLKIN (PD1) input to the PLD AND Array is connected. Every change of CLKIN (PD1) Powers-up the PLD when Turbo Bit is '0.'
		1 = off	CLKIN (PD1) input to PLD AND Array is disconnected, saving power.
Bit 5	PLD MCell clk	0 = on	CLKIN (PD1) input to the PLD macrocells is connected.
		1 = off	CLKIN (PD1) input to PLD macrocells is disconnected, saving power.
Bit 6	X	0	Not used, and should be set to zero.
Bit 7	X	0	Not used, and should be set to zero.

**Table 97. Power Management Mode Registers PMMR2**

Bit 0	X	0	Not used, and should be set to zero.
Bit 1	X	0	Not used, and should be set to zero.
Bit 2	PLD Array WR	0 = on	$\overline{WR}$ input to the PLD AND Array is connected.
		1 = off	$\overline{WR}$ input to PLD AND Array is disconnected, saving power.
Bit 3	PLD Array RD	0 = on	$\overline{RD}$ input to the PLD AND Array is connected.
		1 = off	$\overline{RD}$ input to PLD AND Array is disconnected, saving power.
Bit 4	PLD Array PSEN	0 = on	$\overline{PSEN}$ input to the PLD AND Array is connected.
		1 = off	$\overline{PSEN}$ input to PLD AND Array is disconnected, saving power.
Bit 5	PLD Array ALE	0 = on	ALE input to the PLD AND Array is connected.
		1 = off	ALE input to PLD AND Array is disconnected, saving power.
Bit 6	X	0	Not used, and should be set to zero.
Bit 7	X	0	Not used, and should be set to zero.

Note: The bits of this register are cleared to zero following Power-up. Subsequent RESET pulses do not clear the registers.

**Table 98. APD Counter Operation**

APD Enable Bit	ALE Level	APD Counter
0	X	Not Counting
1	Pulsing	Not Counting
1	0 or 1	Counting (Generates PDN after 15 Clocks)

**RESET TIMING AND DEVICE STATUS AT RESET**

Upon Power-up, the PSD Module requires a Reset (RESET) pulse of duration  $t_{NLNH-PO}$  after  $V_{CC}$  is steady. During this period, the device loads internal configurations, clears some of the registers and sets the Flash memory into operating mode. The Flash memory is reset to the READ Mode upon Power-up. Sector Select (FS0-FS7 and CSBOOT0-CSBOOT3) must all be Low. Any Flash memory WRITE cycle initiation is prevented automatically when  $V_{DD}$  is below  $V_{LKO}$ .

**Warm RESET**

Once the device is up and running, the PSD Module can be reset with a pulse of a much shorter duration,  $t_{NLNH}$ .

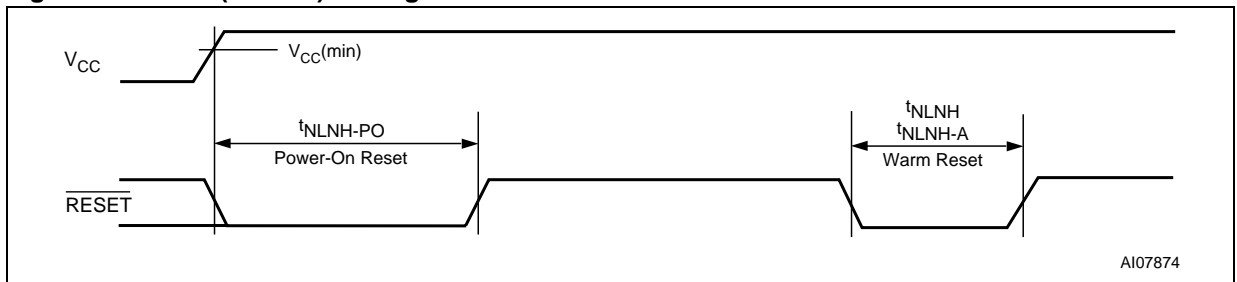
**I/O Pin, Register and PLD Status at RESET**

Table 99 shows the I/O pin, register and PLD status during Power-on RESET, Warm RESET, and Power-down Mode. PLD outputs are always valid during Warm RESET, and they are valid in Power-on RESET once the internal Configuration bits are loaded. This loading is completed typically long before the  $V_{CC}$  ramps up to operating level. Once the PLD is active, the state of the outputs are determined by the PLD equations.

**Reset of Flash Memory Erase and Program Cycles**

A Reset (RESET) also resets the internal Flash memory state machine. During a Flash memory Program or Erase cycle, Reset (RESET) terminates the cycle and returns the Flash memory to the READ Mode within a period of  $t_{NLNH-A}$ .

**Figure 43. Reset (RESET) Timing**



**Table 99. Status During Power-on RESET, Warm RESET and Power-down Mode**

Port Configuration	Power-On RESET	Warm RESET	Power-down Mode
MCU I/O	Input mode	Input mode	Unchanged
PLD Output	Valid after internal PSD configuration bits are loaded	Valid	Depends on inputs to PLD (addresses are blocked in PD Mode)
Address Out	Tri-stated	Tri-stated	Not defined
Peripheral I/O	Tri-stated	Tri-stated	Tri-stated

Register	Power-On RESET	Warm RESET	Power-down Mode
PMMR0 and PMMR2	Cleared to '0'	Unchanged	Unchanged
Macrocells flip-flop status	Cleared to '0' by internal Power-on RESET	Depends on .re and .pr equations	Depends on .re and .pr equations
VM Register <sup>(1)</sup>	Initialized, based on the selection in PSDsoft Configuration menu	Initialized, based on the selection in PSDsoft Configuration menu	Unchanged
All other registers	Cleared to '0'	Cleared to '0'	Unchanged

Note: 1. The SR\_cod and PeriphMode Bits in the VM Register are always cleared to '0' on Power-on RESET or Warm RESET.

## PROGRAMMING IN-CIRCUIT USING THE JTAG SERIAL INTERFACE

The JTAG Serial Interface pins (TMS, TCK, TDI, TDO) are dedicated pins on Port C (see Table 100). All memory blocks (primary and secondary Flash memory), PLD logic, and PSD Module Configuration Register Bits may be programmed through the JTAG Serial Interface block. A blank device can be mounted on a printed circuit board and programmed using JTAG.

The standard JTAG signals (IEEE 1149.1) are TMS, TCK, TDI, and TDO. Two additional signals, TSTAT and TERR, are optional JTAG extensions used to speed up Program and Erase cycles.

*By default, on a blank device (as shipped from the factory or after erasure), four pins on Port C are the basic JTAG signals TMS, TCK, TDI, and TDO.*

### Standard JTAG Signals

At power-up, the standard JTAG pins are inputs, waiting for a JTAG serial command from an external JTAG controller device (such as FlashLINK or Automated Test Equipment). When the enabling command is received, TDO becomes an output and the JTAG channel is fully functional. The same command that enables the JTAG channel may optionally enable the two additional JTAG signals, TSTAT and TERR.

The RESET input to the μPS3200 should be active during JTAG programming. The active RESET puts the MCU module into RESET Mode while the PSD Module is being programmed. See Application Note AN1153 for more details on JTAG In-System Programming (ISP).

The μPSD33XX Devices supports JTAG In-System-Configuration (ISC) commands, but not Boundary Scan. The PSDsoft Express software tool and FlashLINK JTAG programming cable implement the JTAG In-System-Configuration (ISC) commands.

**Table 100. JTAG Port Signals**

Port C Pin	JTAG Signals	Description
PC0	TMS	Mode Select
PC1	TCK	Clock
PC3	<u>TSTAT</u>	Status (optional)
PC4	<u>TERR</u>	Error flag (optional)
PC5	TDI	Serial Data In
PC6	TDO	Serial Data Out

### JTAG Extensions

TSTAT and TERR are two JTAG extension signals enabled by an “ISC\_ENABLE” command received over the four standard JTAG signals (TMS, TCK,

TDI, and TDO). They are used to speed Program and Erase cycles by indicating status on μPDS signals instead of having to scan the status out serially using the standard JTAG channel. See Application Note AN1153.

TERR indicates if an error has occurred when erasing a sector or programming a byte in Flash memory. This signal goes Low (active) when an Error condition occurs, and stays Low until an “ISC\_CLEAR” command is executed or a chip Reset (RESET) pulse is received after an “ISC\_DISABLE” command.

TSTAT behaves the same as Ready/Busy. TSTAT is High when the PSD Module device is in READ Mode (primary and secondary Flash memory contents can be read). TSTAT is Low when Flash memory Program or Erase cycles are in progress, and also when data is being written to the secondary Flash memory.

TSTAT and TERR can be configured as open-drain type signals during an “ISC\_ENABLE” command.

### Security and Flash Memory Protection

When the Security Bit is set, the device cannot be read on a Device Programmer or through the JTAG Port. When using the JTAG Port, only a Full Chip Erase command is allowed.

All other Program, Erase, and Verify commands are blocked. Full Chip Erase returns the part to a non-secured, blank state. The Security Bit can be set in PSDsoft Express Configuration.

All primary and secondary Flash memory sectors can individually be sector protected against erasures. The sector protect bits can be set in PSDsoft Express Configuration.

### INITIAL DELIVERY STATE

When delivered from ST, the μPSD33XX Devices have all bits in the memory and PLDs set to '1.' The code, configuration, and PLD logic are loaded using the programming procedure. Information for programming the device is available directly from ST. Please contact your local sales representative.

**AC/DC PARAMETERS**

These tables describe the AD and DC parameters of the μPSD33XX Devices:

■ DC Electrical Specification

■ AC Timing Specification

■ PLD Timing

- Combinatorial Timing
- Synchronous Clock Mode
- Asynchronous Clock Mode
- Input Macrocell Timing

■ MCU Module Timing

- READ Timing

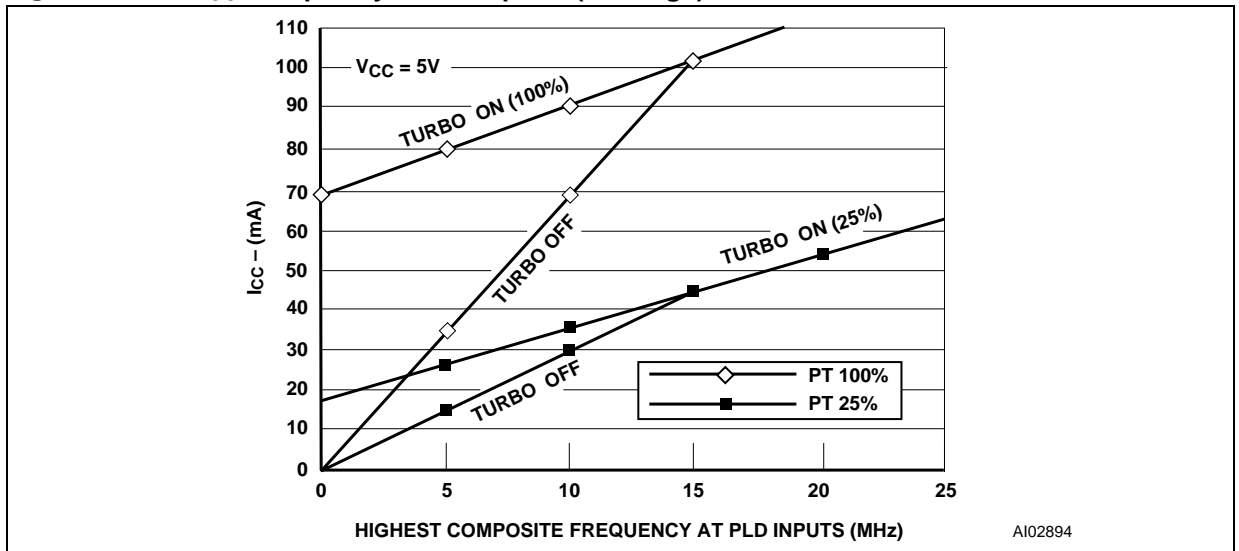
- WRITE Timing

- Power-down and  $\overline{\text{RESET}}$  Timing

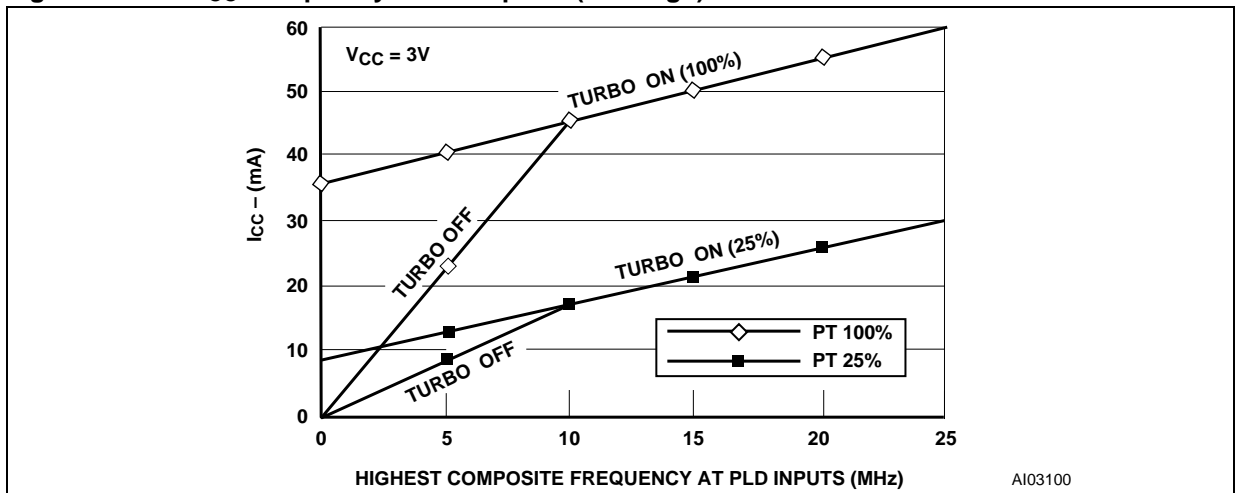
The following are issues concerning the parameters presented:

- In the DC specification the supply current is given for different modes of operation.
- The AC power component gives the PLD, Flash memory, and SRAM mA/MHz specification. Figure 44 and Figure 45 show the PLD mA/MHz as a function of the number of Product Terms (PT) used.
- In the PLD timing parameters, add the required delay when Turbo Bit is '0.'

**Figure 44. PLD  $I_{CC}$  /Frequency Consumption (5V range)**



**Figure 45. PLD  $I_{CC}$  /Frequency Consumption (3V range)**





**Table 101. PSD Module Example, Typ. Power Calculation at V<sub>CC</sub> = 5.0V (Turbo Mode Off)**

Conditions	
MCU Clock Frequency	= 12MHz
Highest Composite PLD input frequency (Freq PLD)	= 8MHz
MCU ALE frequency (Freq ALE)	= 2MHz
% Flash memory Access	= 80%
% SRAM access	= 15%
% I/O access	= 5% (no additional power above base)
Operational Modes	
% Normal	= 40%
% Power-down Mode	= 60%
Number of product terms used	
(from fitter report)	= 45 PT
% of total product terms	= 45/182 = 24.7%
Turbo Mode	= Off
Calculation (using typical values)	
I <sub>CC</sub> total	= I <sub>CC</sub> (MCUactive) x %MCUactive + I <sub>CC</sub> (PSDactive) x %PSDactive + I <sub>PD</sub> (pwrdown) x %pwrdown
I <sub>CC</sub> (MCUactive)	= 20mA
I <sub>PD</sub> (pwrdown)	= 250μA
I <sub>CC</sub> (PSDactive)	= I <sub>CC</sub> (ac) + I <sub>CC</sub> (dc)
	= %flash x 2.5 mA/MHz x Freq ALE
	+ %SRAM x 1.5 mA/MHz x Freq ALE
	+ % PLD x (from graph using Freq PLD)
	= 0.8 x 2.5 mA/MHz x 2MHz + 0.15 x 1.5 mA/MHz x 2MHz + 24 mA
	= (4 + 0.45 + 24) mA
	= 28.45mA
I <sub>CC</sub> total	= 20mA x 40% + 28.45mA x 40% + 250μA x 60%
	= 8mA + 11.38mA + 150μA
	= 19.53mA
<p>This is the operating power with no Flash memory Erase or Program cycles in progress. Calculation is based on all I/O pins being disconnected and I<sub>OUT</sub> = 0 mA.</p>	

**MAXIMUM RATING**

Stressing the device above the rating listed in the Absolute Maximum Ratings” table may cause permanent damage to the device. These are stress ratings only and operation of the device at these or any other conditions above those indicated in the Operating sections of this specification is not im-

plied. Exposure to Absolute Maximum Rating conditions for extended periods may affect device reliability. Refer also to the STMicroelectronics SURE Program and other relevant quality documents.

**Table 102. Absolute Maximum Ratings**

Symbol	Parameter	Min.	Max.	Unit
T <sub>STG</sub>	Storage Temperature	-65	125	°C
T <sub>LEAD</sub>	Lead Temperature during Soldering (20 seconds max.) <sup>(1)</sup>		235	°C
V <sub>IO</sub>	Input and Output Voltage (Q = V <sub>OH</sub> or Hi-Z)	-0.5	6.5	V
V <sub>CC</sub>	Supply Voltage	-0.5	6.5	V
V <sub>PP</sub>	Device Programmer Supply Voltage	-0.5	14.0	V
V <sub>ESD</sub>	Electrostatic Discharge Voltage (Human Body Model) <sup>(2)</sup>	-2000	2000	V

Note: 1. IPC/JEDEC J-STD-020A  
 2. JEDEC Std JESD22-A114A (C1=100pF, R1=1500 Ω, R2=500 Ω)

**DC AND AC PARAMETERS**

This section summarizes the operating and measurement conditions, and the DC and AC characteristics of the device. The parameters in the DC and AC Characteristic tables that follow are derived from tests performed under the Measure-

ment Conditions summarized in the relevant tables. Designers should check that the operating conditions in their circuit match the measurement conditions when relying on the quoted parameters.

**Table 103. Operating Conditions (5V Devices)**

Symbol	Parameter	Min.	Max.	Unit
V <sub>CC</sub>	Supply Voltage	4.5	5.5	V
T <sub>A</sub>	Ambient Operating Temperature (industrial)	-40	85	°C
	Ambient Operating Temperature (commercial)	0	70	°C

**Table 104. Operating Conditions (3.3V Devices)**

Symbol	Parameter	Min.	Max.	Unit
V <sub>CC</sub>	Supply Voltage	3.0	3.6	V
T <sub>A</sub>	Ambient Operating Temperature (industrial)	-40	85	°C
	Ambient Operating Temperature (commercial)	0	70	°C

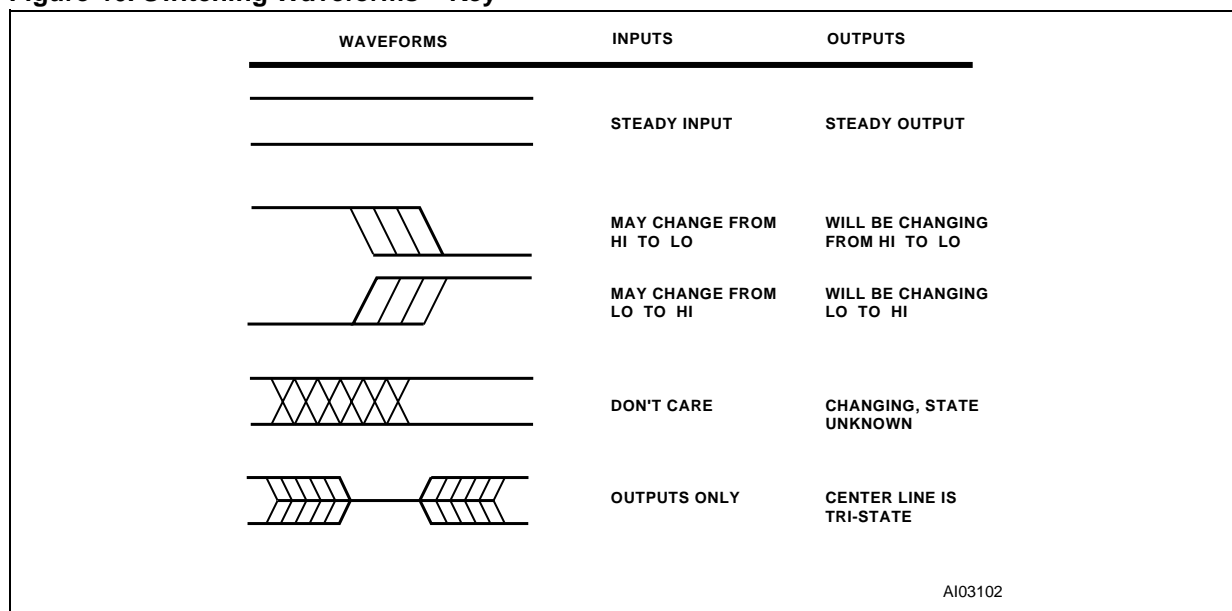
**Table 105. AC Symbols for Timing**

Signal Letters	
A	Address
C	Clock
D	Input Data
I	Instruction
L	ALE
N	RESET Input or Output
P	PSEN signal
Q	Output Data
R	RD signal
W	WR signal
B	V <sub>STBY</sub> Output
M	Output Macrocell

Signal Behavior	
t	Time
L	Logic Level Low or ALE
H	Logic Level High
V	Valid
X	No Longer a Valid Logic Level
Z	Float
PW	Pulse Width

Example:  $t_{AVLX}$  – Time from Address Valid to ALE Invalid.

**Figure 46. Switching Waveforms – Key**



**Table 106. Preliminary MCU Module DC Characteristics**

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V <sub>CC</sub>	Supply Voltage <sup>(1)</sup>		3.0		3.6	V
V <sub>IH</sub>	High Level Input Voltage (Ports 0, 1, 2, 3, 4, XTAL1, RESET) 5V Tolerant - max voltage 5.5V	3.0V < V <sub>CC</sub> < 3.6V	0.7V <sub>CC</sub>		5.5	V
V <sub>IL</sub>	Low Level Input Voltage (Ports 0, 1, 2, 3, 4, XTAL1, RESET)	3.0V < V <sub>CC</sub> < 3.6V	V <sub>SS</sub> - 0.5		0.3V <sub>CC</sub>	V
V <sub>OL1</sub>	Output Low Voltage (Port 4)	I <sub>OL</sub> = 10mA			0.6	V
						V
V <sub>OL2</sub>	Output Low Voltage (Other Ports)	I <sub>OL</sub> = 5mA			0.6	V
						V
V <sub>OH1</sub>	Output High Voltage (Ports 4 push-pull)	I <sub>OH</sub> = -10mA	2.4			V
						V
V <sub>OH2</sub>	Output High Voltage (Other Ports push-pull)	I <sub>OH</sub> = -5mA	2.4			V
						V
V <sub>OP</sub>	XTAL Open Bias Voltage (XTAL1, XTAL2)	I <sub>OL</sub> = 3.2mA	1.0		2.0	V
I <sub>RST</sub>	RESET Pin Pull-up Current (RESET)	V <sub>IN</sub> = V <sub>SS</sub>	-10		-55	μA
I <sub>FR</sub>	XTAL Feedback Resistor Current (XTAL1)	XTAL1 = V <sub>CC</sub> ; XTAL2 = V <sub>SS</sub>	TBD (-20)		TBD (-50)	μA
I <sub>IHL1</sub>	Input High Leakage Current (Port 0)	V <sub>SS</sub> < V <sub>IN</sub> < 5.5V	-10		10	μA
I <sub>IHL2</sub>	Input High Leakage Current (Port 1, 2, 3, 4)	V <sub>IH</sub> = 2.3V	-10		10	μA
I <sub>ILL</sub>	Input Low Leakage Current (Port 1, 2, 3, 4)	V <sub>IL</sub> < 0.5V	-10		10	μA
I <sub>PD</sub> (Note 2)	Power-down Mode	V <sub>CC</sub> = 3.6V LVD Logic disabled			10	μA
		LVD Logic enabled			80	μA
I <sub>CC-CPU</sub> (Note 3,4,5)	Active - 12MHz	V <sub>CC</sub> = 3.6V		TBD		mA
	Idle - 12MHz			TBD		mA
	Active - 24MHz	V <sub>CC</sub> = 3.6V		TBD		mA
	Idle - 24MHz			TBD		mA
	Active - 40MHz	V <sub>CC</sub> = 3.6V		TBD		mA
	Idle - 40MHz			TBD		mA
V <sub>LVD</sub>	LVD Low Voltage Detect Reset Threshold	V <sub>CC</sub> = 3.3V	2.3	2.5	2.7	V

- Note: 1. Power supply (V<sub>CC</sub>) is always 3.0 to 3.6V for the MCU Module. V<sub>DD</sub> for the PSD Module may be 3V or 5V.  
 2. I<sub>PD</sub> (Power-down Mode) is measured with: XTAL1 = V<sub>SS</sub>; XTAL2 = NC; RESET = V<sub>CC</sub>; Port 0 = V<sub>CC</sub>; all other pins are disconnected.  
 3. I<sub>CC-CPU</sub> (Active Mode) is measured with: XTAL1 driven with t<sub>CLCH</sub>, t<sub>CHCL</sub> = 5ns, V<sub>IL</sub> = V<sub>SS</sub> + 0.5V, V<sub>IH</sub> = V<sub>CC</sub> - 0.5V, XTAL2 = NC; RESET = V<sub>SS</sub>; Port 0 = V<sub>CC</sub>; all other pins are disconnected. I<sub>CC</sub> would be slightly higher if a crystal oscillator is used (approximately 1mA).  
 4. I<sub>CC-CPU</sub> (Idle Mode) is measured with: XTAL1 driven with t<sub>CLCH</sub>, t<sub>CHCL</sub> = 5ns, V<sub>IL</sub> = V<sub>SS</sub> + 0.5V, V<sub>IH</sub> = V<sub>CC</sub> - 0.5V, XTAL2 = NC; RESET = V<sub>CC</sub>; Port 0 = V<sub>CC</sub>; all other pins are disconnected. I<sub>CC</sub> would be slightly higher if a crystal oscillator is used (approximately 1mA).  
 5. I/O current = 0mA, all I/O pins are disconnected.

**Table 107. PSD Module DC Characteristics (with 5V V<sub>DD</sub>)**

Symbol	Parameter		Test Condition (in addition to those in Table 103, page 98)	Min.	Typ.	Max.	Unit
V <sub>IH</sub>	Input High Voltage		4.5V < V <sub>DD</sub> < 5.5V	2		V <sub>DD</sub> +0.5	V
V <sub>IL</sub>	Input Low Voltage		4.5V < V <sub>DD</sub> < 5.5V	-0.5		0.8	V
V <sub>LKO</sub>	V <sub>DD</sub> (min) for Flash Erase and Program			2.5		4.2	V
V <sub>OL</sub>	Output Low Voltage		I <sub>OL</sub> = 20μA, V <sub>DD</sub> = 4.5V		0.01	0.1	V
			I <sub>OL</sub> = 8mA, V <sub>DD</sub> = 4.5V		0.25	0.45	V
V <sub>OH</sub>	Output High Voltage Except V <sub>STBY</sub> On		I <sub>OH</sub> = -20μA, V <sub>DD</sub> = 4.5V	4.4	4.49		V
			I <sub>OH</sub> = -2mA, V <sub>DD</sub> = 4.5V	2.4	3.9		V
V <sub>OH1</sub>	Output High Voltage V <sub>STBY</sub> On		I <sub>OH1</sub> = 1μA	V <sub>STBY</sub> - 0.8			V
V <sub>STBY</sub>	SRAM Stand-by Voltage			2.0		V <sub>DD</sub>	V
I <sub>STBY</sub>	SRAM Stand-by Current		V <sub>DD</sub> = 0V		0.5	1	μA
I <sub>IDLE</sub>	Idle Current (V <sub>STBY</sub> input)		V <sub>DD</sub> > V <sub>STBY</sub>	-0.1		0.1	μA
V <sub>DF</sub>	SRAM Data Retention Voltage		Only on V <sub>STBY</sub>	2		V <sub>DD</sub> - 0.2	V
I <sub>SB</sub>	Stand-by Supply Current for Power-down Mode		$\overline{CSI} > V_{DD} - 0.3V$ (Notes 1,2)		50	200	μA
I <sub>LI</sub>	Input Leakage Current		V <sub>SS</sub> < V <sub>IN</sub> < V <sub>DD</sub>	-1	±0.1	1	μA
I <sub>LO</sub>	Output Leakage Current		0.45 < V <sub>OUT</sub> < V <sub>DD</sub>	-10	±5	10	μA
I <sub>CC</sub> (DC) (Note 4)	Operating Supply Current	PLD Only	PLD_TURBO = Off, f = 0MHz (Note 4)		0		μA/PT
			PLD_TURBO = On, f = 0MHz		400	700	μA/PT
		Flash memory	During Flash memory WRITE/Erase Only		15	30	mA
			Read only, f = 0MHz		0	0	mA
		SRAM	f = 0MHz		0	0	mA
I <sub>CC</sub> (AC) (Note 4)	PLD AC Adder					Note 3	
	Flash memory AC Adder				2.5	3.5	mA/ MHz
	SRAM AC Adder				1.5	3.0	mA/ MHz

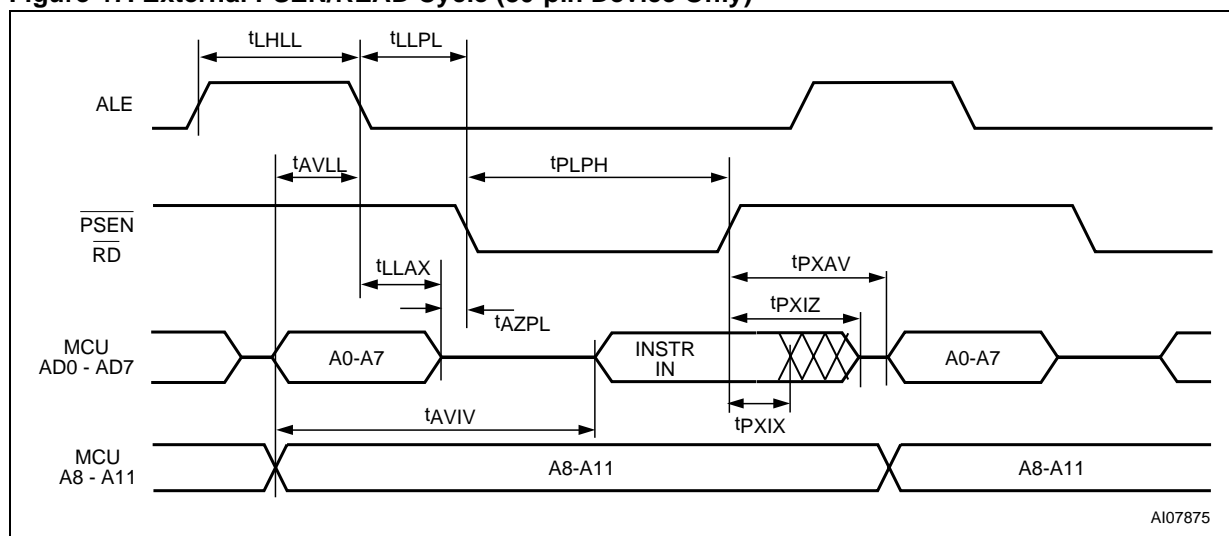
- Note: 1.  $\overline{CSI}$  deselected or internal Power-down mode is active.  
 2. PLD is in non-Turbo mode, and none of the inputs are switching.  
 3. Please see Figure 44 for the PLD current calculation.  
 4. I<sub>OUT</sub> = 0 mA

**Table 108. PSD Module DC Characteristics (with 3.3V V<sub>DD</sub>)**

Symbol	Parameter		Test Condition (in addition to those in Table 104, page 98)	Min.	Typ.	Max.	Unit
V <sub>IH</sub>	High Level Input Voltage		3.0V < V <sub>DD</sub> < 3.6V	0.7V <sub>DD</sub>		V <sub>DD</sub> +0.5	V
V <sub>IL</sub>	Low Level Input Voltage		3.0V < V <sub>DD</sub> < 3.6V	-0.5		0.8	V
V <sub>LKO</sub>	V <sub>DD</sub> (min) for Flash Erase and Program			1.5		2.2	V
V <sub>OL</sub>	Output Low Voltage		I <sub>OL</sub> = 20μA, V <sub>DD</sub> = 3.0V		0.01	0.1	V
			I <sub>OL</sub> = 4mA, V <sub>DD</sub> = 3.0V		0.15	0.45	V
V <sub>OH</sub>	Output High Voltage Except V <sub>STBY</sub> On		I <sub>OH</sub> = -20μA, V <sub>DD</sub> = 3.0V	2.9	2.99		V
			I <sub>OH</sub> = -1mA, V <sub>DD</sub> = 3.0V	2.7	2.8		V
V <sub>OH1</sub>	Output High Voltage V <sub>STBY</sub> On		I <sub>OH1</sub> = 1μA	V <sub>STBY</sub> - 0.8			V
V <sub>STBY</sub>	SRAM Stand-by Voltage			2.0		V <sub>DD</sub>	V
I <sub>STBY</sub>	SRAM Stand-by Current		V <sub>DD</sub> = 0V		0.5	1	μA
I <sub>IDLE</sub>	Idle Current (V <sub>STBY</sub> input)		V <sub>DD</sub> > V <sub>STBY</sub>	-0.1		0.1	μA
V <sub>DF</sub>	SRAM Data Retention Voltage		Only on V <sub>STBY</sub>	2		V <sub>DD</sub> - 0.2	V
I <sub>SB</sub>	Stand-by Supply Current for Power-down Mode		CS <sub>I</sub> > V <sub>DD</sub> - 0.3V (Notes 1,2)		25	100	μA
I <sub>LI</sub>	Input Leakage Current		V <sub>SS</sub> < V <sub>IN</sub> < V <sub>DD</sub>	-1	±0.1	1	μA
I <sub>LO</sub>	Output Leakage Current		0.45 < V <sub>IN</sub> < V <sub>DD</sub>	-10	±5	10	μA
I <sub>CC</sub> (DC) (Note 4)	Operating Supply Current	PLD Only	PLD_TURBO = Off, f = 0MHz (Note 2)		0		μA/PT
			PLD_TURBO = On, f = 0MHz		200	400	μA/PT
		Flash memory	During Flash memory WRITE/Erase Only		10	25	mA
			Read only, f = 0MHz		0	0	mA
SRAM	f = 0MHz		0	0	mA		
I <sub>CC</sub> (AC) (Note 4)	PLD AC Adder				Note 3		
	Flash memory AC Adder				1.5	2.0	mA/ MHz
	SRAM AC Adder				0.8	1.5	mA/ MHz

- Note: 1. CS<sub>I</sub> deselected or internal PD is active.  
 2. PLD is in non-Turbo mode, and none of the inputs are switching.  
 3. Please see Figure 45 for the PLD current calculation.  
 4. I<sub>OUT</sub> = 0 mA

Figure 47. External PSEN/READ Cycle (80-pin Device Only)



AI07875

Table 109. External PSEN or READ Cycle AC Characteristics (3V or 5V Device)

Symbol	Parameter	40MHz Oscillator <sup>(1)</sup>		Variable Oscillator 1/t <sub>CLCL</sub> = 8 to 40MHz		Unit
		Min	Max	Min	Max	
t <sub>LHLL</sub>	ALE pulse width	17		t <sub>CLCL</sub> - 8		ns
t <sub>AVLL</sub>	Address setup to ALE	13		t <sub>CLCL</sub> - 12		ns
t <sub>LLAX</sub>	Address hold after ALE	7.5		0.5t <sub>CLCL</sub> - 5		ns
t <sub>LLPL</sub>	ALE to $\overline{\text{PSEN}}$ or $\overline{\text{RD}}$	7.5		0.5t <sub>CLCL</sub> - 5		ns
t <sub>PLPH</sub>	$\overline{\text{PSEN}}$ or $\overline{\text{RD}}$ pulse width <sup>(2)</sup>	40		n t <sub>CLCL</sub> - 10		ns
t <sub>PXIX</sub>	Input instruction/data hold after $\overline{\text{PSEN}}$ or $\overline{\text{RD}}$	2		2		ns
t <sub>PHIZ</sub>	Input instruction/data float after $\overline{\text{PSEN}}$ or $\overline{\text{RD}}$		10.5		0.5t <sub>CLCL</sub> - 2	ns
t <sub>PXAV</sub>	Address hold after $\overline{\text{PSEN}}$ or $\overline{\text{RD}}$	7.5		0.5t <sub>CLCL</sub> - 5		ns
t <sub>AVIV</sub>	Address to valid instruction/data in <sup>(2)</sup>		70		m t <sub>CLCL</sub> - 5	ns
t <sub>AZPL</sub>	Address float to $\overline{\text{PSEN}}$ or $\overline{\text{RD}}$	-2		-2		ns

Note: 1. BUSCON Register is configured for 4 PFQCLK.

2. Refer to Table 110 for “n” and “m” values.

Table 110. n, m, and x, y Values

# of PFQCLK in BUSCON Reg.	PSEN (code) Cycle		READ Cycle		WRITE Cycle	
	n	m	n	m	x	y
3	1	2	-	-	-	-
4	2	3	2	3	2	1
5	3	4	3	4	3	2
6	4	5	4	5	4	3
7	-	-	5	6	5	4

Figure 48. External WRITE Cycle (80-pin Device Only)

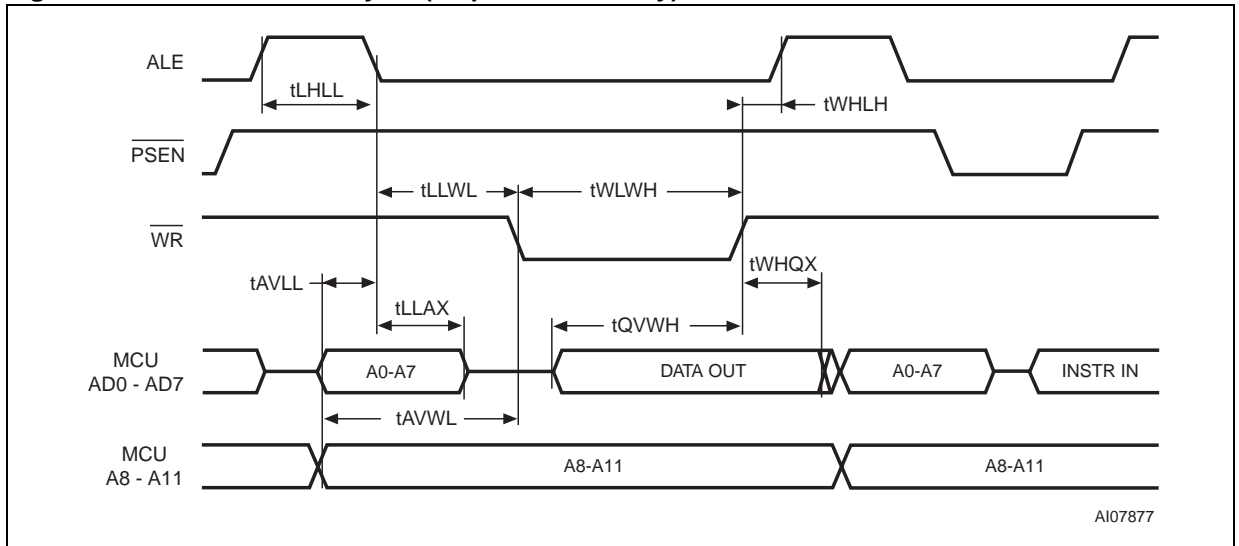


Table 111. External WRITE Cycle AC Characteristics (3V or 5V Device)

Symbol	Parameter	40MHz Oscillator <sup>(1)</sup>		Variable Oscillator 1/t <sub>CLCL</sub> = 8 to 40MHz		Unit
		Min	Max	Min	Max	
t <sub>LHLL</sub>	ALE pulse width	17		t <sub>CLCL</sub> - 8		ns
t <sub>AVLL</sub>	Address Setup to ALE	13		t <sub>CLCL</sub> - 12		ns
t <sub>LLAX</sub>	Address hold after ALE	7.5		0.5t <sub>CLCL</sub> - 5		ns
t <sub>WLWH</sub>	WR pulse width <sup>(2)</sup>	40		x t <sub>CLCL</sub> - 10		ns
t <sub>LLWL</sub>	ALE to $\overline{WR}$	7.5		0.5t <sub>CLCL</sub> - 5		ns
t <sub>AVWL</sub>	Address valid to $\overline{WR}$	27.5		1.5t <sub>CLCL</sub> - 10		ns
t <sub>WHLH</sub>	$\overline{WR}$ High to ALE High	6.5	14.5	0.5t <sub>CLCL</sub> - 6	0.5t <sub>CLCL</sub> + 2	ns
t <sub>QVWH</sub>	Data setup before $\overline{WR}$ <sup>(y)</sup>	20		y t <sub>CLCL</sub> - 5		ns
t <sub>WHQX</sub>	Data hold after $\overline{WR}$	6.5	14.5	0.5t <sub>CLCL</sub> - 6	0.5t <sub>CLCL</sub> + 2	ns

Note: 1. BUSCON Register is configured for 4 PFQCLK.  
 2. Refer to Table 110, page 103 for "n" and "m" values.

Table 112. External Clock Drive

Symbol	Parameter <sup>(1)</sup>	40MHz Oscillator		Variable Oscillator 1/t <sub>CLCL</sub> = 8 to 40MHz		Unit
		Min	Max	Min	Max	
t <sub>CLCL</sub>	Oscillator period			25	125	ns
t <sub>CHCX</sub>	High time			10	t <sub>CLCL</sub> - t <sub>CLCX</sub>	ns
t <sub>CLCX</sub>	Low time			10	t <sub>CLCL</sub> - t <sub>CLCX</sub>	ns
t <sub>CLCH</sub>	Rise time				10	ns
t <sub>CHCL</sub>	Fall time				10	ns



**Table 113. A/D Analog Specification**

Symbol	Parameter	Test Conditions <sup>(1)</sup>	Min.	Typ.	Max.	Unit
I <sub>DD</sub>	Normal	Input = AV <sub>REF</sub>		4.0		mA
	Power-down				40	μA
AV <sub>IN</sub>	Analog Input Voltage		GND		AV <sub>REF</sub>	V
AV <sub>REF</sub> <sup>(2)</sup>	Analog Reference Voltage				3.6	V
Accuracy	Resolution				10	bits
INL	Integral Nonlinearity	Input = 0 – AV <sub>REF</sub> (V)			±1	LSB
DNL	Differential Nonlinearity	Input = 0 – AV <sub>REF</sub> (V)			±1	LSB
SNR	Signal to Noise Ratio	f <sub>SAMPLE</sub> = 500ksps	50	54		dB
SNDR	Signal to Noise Distortion Ratio		48	52		dB
ACLK	ADC Clock		2	8	16	MHz
t <sub>c</sub>	Conversion Time	8MHz	1	4	8	μS
t <sub>CAL</sub>	Power-up Time	Calibration Time		16		mS
f <sub>IN</sub>	Analog Input Frequency				60	kHz
THD	Total Harmonic Distortion		50	54		dB

Note: 1. f<sub>IN</sub> 2kHz, ACLK = 8MHz, AV<sub>REF</sub> = V<sub>CC</sub> = 3.3V  
 2. AV<sub>REF</sub> = V<sub>CC</sub> in 52-pin package.

Figure 49. Input to Output Disable / Enable

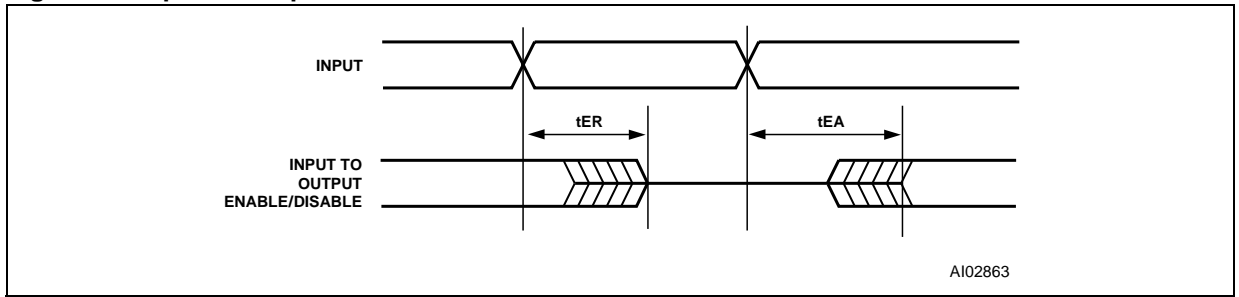


Table 114. CPLD Combinatorial Timing (5V PSD Module)

Symbol	Parameter	Conditions	Min	Max	PT Alloc	Turbo Off	Slew rate <sup>(1)</sup>	Unit
t <sub>PD</sub> <sup>(2)</sup>	CPLD Input Pin/Feedback to CPLD Combinatorial Output			20	+ 2	+ 10	- 2	ns
t <sub>EA</sub>	CPLD Input to CPLD Output Enable			21		+ 10	- 2	ns
t <sub>ER</sub>	CPLD Input to CPLD Output Disable			21		+ 10	- 2	ns
t <sub>ARP</sub>	CPLD Register Clear or Preset Delay			21		+ 10	- 2	ns
t <sub>ARPW</sub>	CPLD Register Clear or Preset Pulse Width		10			+ 10		ns
t <sub>ARD</sub>	CPLD Array Delay	Any macrocell		11	+ 2			ns

Note: 1. Fast Slew Rate output available on PA3-PA0, PB3-PB0, and PD2-PD1. Decrement times by given amount  
 2. t<sub>PD</sub> for MCU address and control signals refers to delay from pins on Port 0, Port 2, RD WR, PSEN and ALE to CPLD combinatorial output (80-pin package only)

Table 115. CPLD Combinatorial Timing (3V PSD Module)

Symbol	Parameter	Conditions	Min	Max	PT Alloc	Turbo Off	Slew rate <sup>(1)</sup>	Unit
t <sub>PD</sub> <sup>(2)</sup>	CPLD Input Pin/Feedback to CPLD Combinatorial Output			40	+ 4	+ 20	- 6	ns
t <sub>EA</sub>	CPLD Input to CPLD Output Enable			43		+ 20	- 6	ns
t <sub>ER</sub>	CPLD Input to CPLD Output Disable			43		+ 20	- 6	ns
t <sub>ARP</sub>	CPLD Register Clear or Preset Delay			40		+ 20	- 6	ns
t <sub>ARPW</sub>	CPLD Register Clear or Preset Pulse Width		25			+ 20		ns
t <sub>ARD</sub>	CPLD Array Delay	Any macrocell		25	+ 4			ns

Note: 1. Fast Slew Rate output available on PA3-PA0, PB3-PB0, and PD2-PD1. Decrement times by given amount  
 2. t<sub>PD</sub> for MCU address and control signals refers to delay from pins on Port 0, Port 2, RD WR, PSEN and ALE to CPLD combinatorial output (80-pin package only)

Figure 50. Synchronous Clock Mode Timing – PLD

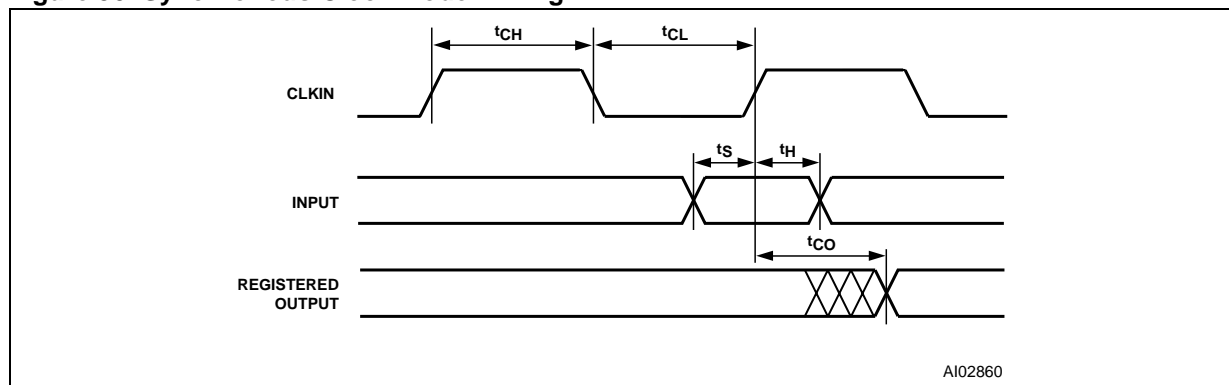


Table 116. CPLD Macrocell Synchronous Clock Mode Timing (5V PSD Module)

Symbol	Parameter	Conditions	Min	Max	PT Alloc	Turbo Off	Slew rate <sup>(1)</sup>	Unit
f <sub>MAX</sub>	Maximum Frequency External Feedback	1/(t <sub>S</sub> +t <sub>CO</sub> )		40.0				MHz
	Maximum Frequency Internal Feedback (f <sub>CNT</sub> )	1/(t <sub>S</sub> +t <sub>CO</sub> -10)		66.6				MHz
	Maximum Frequency Pipelined Data	1/(t <sub>CH</sub> +t <sub>CL</sub> )		83.3				MHz
t <sub>S</sub>	Input Setup Time		12		+ 2	+ 10		ns
t <sub>H</sub>	Input Hold Time		0					ns
t <sub>CH</sub>	Clock High Time	Clock Input	6					ns
t <sub>CL</sub>	Clock Low Time	Clock Input	6					ns
t <sub>CO</sub>	Clock to Output Delay	Clock Input		13			- 2	ns
t <sub>ARD</sub>	CPLD Array Delay	Any macrocell		11	+ 2			ns
t <sub>MIN</sub>	Minimum Clock Period <sup>(2)</sup>	t <sub>CH</sub> +t <sub>CL</sub>	12					ns

Note: 1. Fast Slew Rate output available on PA3-PA0, PB3-PB0, and PD2-PD1. Decrement times by given amount.  
 2. CLKIN (PD1) t<sub>CLCL</sub> = t<sub>CH</sub> + t<sub>CL</sub>.

Table 117. CPLD Macrocell Synchronous Clock Mode Timing (3V PSD Module)

Symbol	Parameter	Conditions	Min	Max	PT Alloc	Turbo Off	Slew rate <sup>(1)</sup>	Unit
f <sub>MAX</sub>	Maximum Frequency External Feedback	1/(t <sub>S</sub> +t <sub>CO</sub> )		22.2				MHz
	Maximum Frequency Internal Feedback (f <sub>CNT</sub> )	1/(t <sub>S</sub> +t <sub>CO</sub> -10)		28.5				MHz
	Maximum Frequency Pipelined Data	1/(t <sub>CH</sub> +t <sub>CL</sub> )		40.0				MHz
t <sub>S</sub>	Input Setup Time		20		+ 4	+ 20		ns
t <sub>H</sub>	Input Hold Time		0					ns
t <sub>CH</sub>	Clock High Time	Clock Input	15					ns
t <sub>CL</sub>	Clock Low Time	Clock Input	10					ns
t <sub>CO</sub>	Clock to Output Delay	Clock Input		25			- 6	ns
t <sub>ARD</sub>	CPLD Array Delay	Any macrocell		25	+ 4			ns
t <sub>MIN</sub>	Minimum Clock Period <sup>(2)</sup>	t <sub>CH</sub> +t <sub>CL</sub>	25					ns

Note: 1. Fast Slew Rate output available on PA3-PA0, PB3-PB0, and PD2-PD1. Decrement times by given amount.  
 2. CLKIN (PD1) t<sub>CLCL</sub> = t<sub>CH</sub> + t<sub>CL</sub>.

Figure 51. Asynchronous RESET / Preset

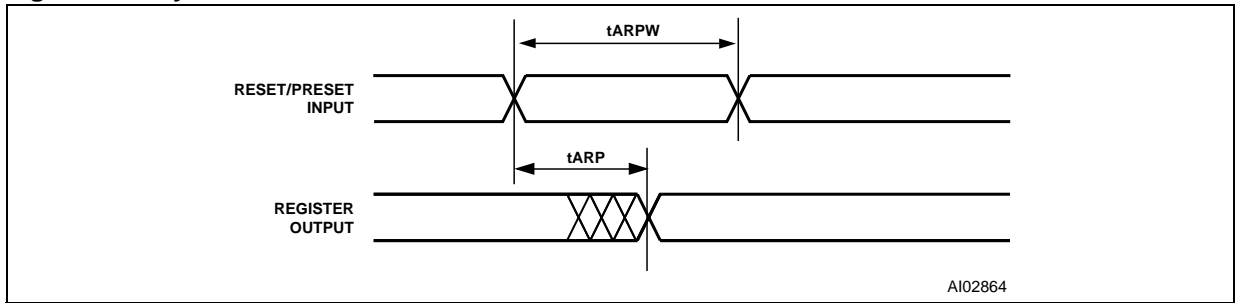


Figure 52. Asynchronous Clock Mode Timing (product term clock)

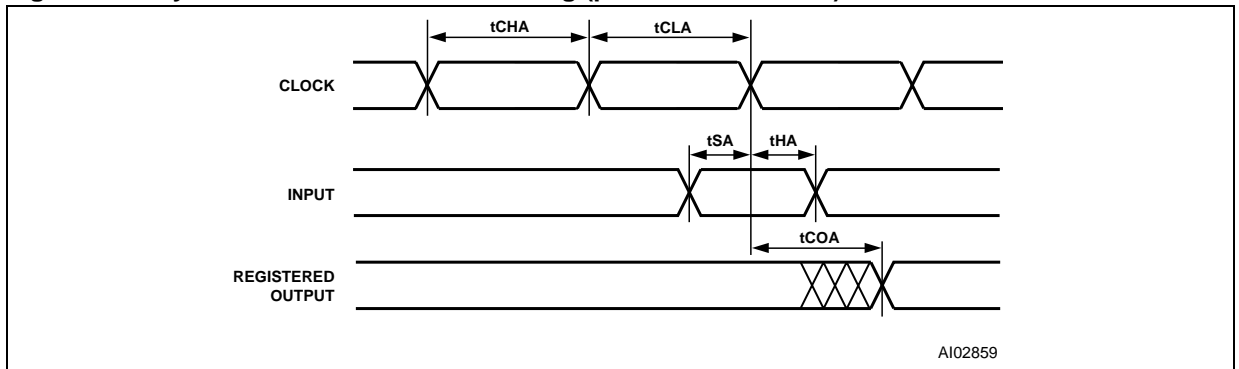


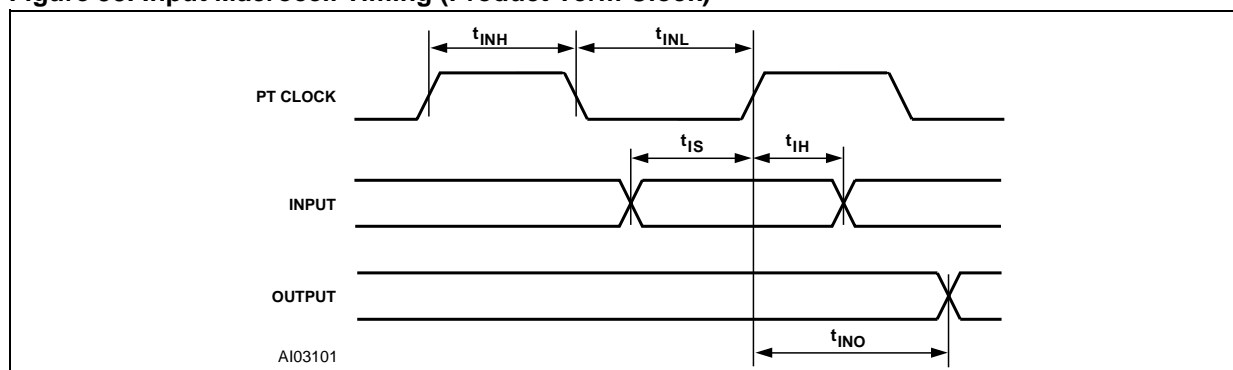
Table 118. CPLD Macrocell Asynchronous Clock Mode Timing (5V PSD Module)

Symbol	Parameter	Conditions	Min	Max	PT Alloc	Turbo Off	Slew Rate	Unit
f <sub>MAXA</sub>	Maximum Frequency External Feedback	1/(t <sub>SA</sub> +t <sub>COA</sub> )		38.4				MHz
	Maximum Frequency Internal Feedback (f <sub>CNTA</sub> )	1/(t <sub>SA</sub> +t <sub>COA</sub> -10)		62.5				MHz
	Maximum Frequency Pipelined Data	1/(t <sub>CHA</sub> +t <sub>CLA</sub> )		71.4				MHz
t <sub>SA</sub>	Input Setup Time		7		+ 2	+ 10		ns
t <sub>HA</sub>	Input Hold Time		8					ns
t <sub>CHA</sub>	Clock Input High Time		9			+ 10		ns
t <sub>CLA</sub>	Clock Input Low Time		9			+ 10		ns
t <sub>COA</sub>	Clock to Output Delay			21		+ 10	- 2	ns
t <sub>ARDA</sub>	CPLD Array Delay	Any macrocell		11	+ 2			ns
t <sub>MINA</sub>	Minimum Clock Period	1/f <sub>CNTA</sub>	16					ns

**Table 119. CPLD Macrocell Asynchronous Clock Mode Timing (3V PSD Module)**

Symbol	Parameter	Conditions	Min	Max	PT Alloc	Turbo Off	Slew Rate	Unit
f <sub>MAXA</sub>	Maximum Frequency External Feedback	1/(t <sub>SA</sub> +t <sub>COA</sub> )		21.7				MHz
	Maximum Frequency Internal Feedback (f <sub>CNTA</sub> )	1/(t <sub>SA</sub> +t <sub>COA</sub> -10)		27.8				MHz
	Maximum Frequency Pipelined Data	1/(t <sub>CHA</sub> +t <sub>CLA</sub> )		33.3				MHz
t <sub>SA</sub>	Input Setup Time		10		+ 4	+ 20		ns
t <sub>HA</sub>	Input Hold Time		12					ns
t <sub>CHA</sub>	Clock High Time		17			+ 20		ns
t <sub>CLA</sub>	Clock Low Time		13			+ 20		ns
t <sub>COA</sub>	Clock to Output Delay			36		+ 20	- 6	ns
t <sub>ARD</sub>	CPLD Array Delay	Any macrocell		25	+ 4			ns
t <sub>MINA</sub>	Minimum Clock Period	1/f <sub>CNTA</sub>	36					ns

**Figure 53. Input Macrocell Timing (Product Term Clock)**



**Table 120. Input Macrocell Timing (5V PSD Module)**

Symbol	Parameter	Conditions	Min	Max	PT Alloc	Turbo Off	Unit
t <sub>IS</sub>	Input Setup Time	(Note 1)	0				ns
t <sub>IH</sub>	Input Hold Time	(Note 1)	15			+ 10	ns
t <sub>INH</sub>	NIB Input High Time	(Note 1)	9				ns
t <sub>INL</sub>	NIB Input Low Time	(Note 1)	9				ns
t <sub>INO</sub>	NIB Input to Combinatorial Delay	(Note 1)		34	+ 2	+ 10	ns

Note: 1. Inputs from Port A, B, and C relative to register/ latch clock from the PLD. ALE/AS latch timings refer to t<sub>AVLX</sub> and t<sub>LXAX</sub>.

**Table 121. Input Macrocell Timing (3V PSD Module)**

Symbol	Parameter	Conditions	Min	Max	PT Alloc	Turbo Off	Unit
t <sub>IS</sub>	Input Setup Time	(Note 1)	0				ns
t <sub>IH</sub>	Input Hold Time	(Note 1)	25			+ 20	ns
t <sub>INH</sub>	NIB Input High Time	(Note 1)	12				ns
t <sub>INL</sub>	NIB Input Low Time	(Note 1)	12				ns
t <sub>INO</sub>	NIB Input to Combinatorial Delay	(Note 1)		46	+ 4	+ 20	ns

Note: 1. Inputs from Port A, B, and C relative to register/latch clock from the PLD. ALE latch timings refer to t<sub>AVLX</sub> and t<sub>LXAX</sub>.

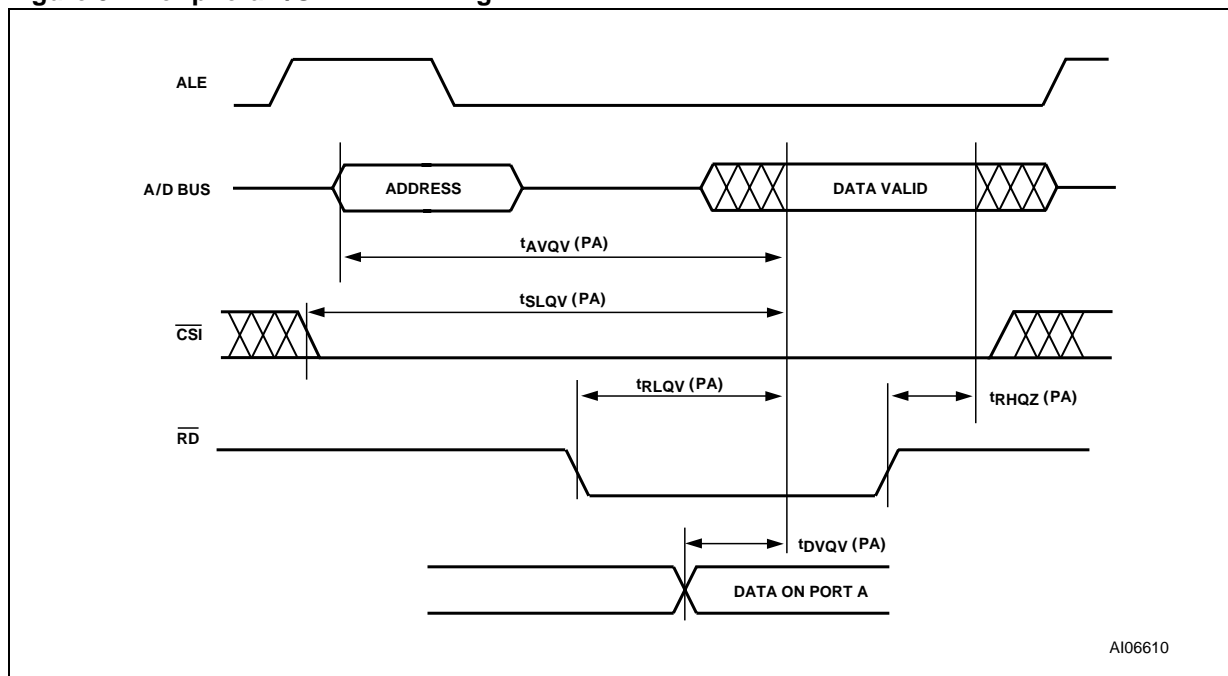
**Table 122. Program, WRITE and Erase Times (5V, 3V PSD Modules)**

Symbol	Parameter	Min.	Typ.	Max.	Unit
	Flash Program		8.5		s
	Flash Bulk Erase <sup>(1)</sup> (pre-programmed)		3	30	s
	Flash Bulk Erase (not pre-programmed)		5		s
t <sub>WHQV3</sub>	Sector Erase (pre-programmed)		1	30	s
t <sub>WHQV2</sub>	Sector Erase (not pre-programmed)		2.2		s
t <sub>WHQV1</sub>	Byte Program		14	150	μs
	Program / Erase Cycles (per Sector)	100,000			cycles
t <sub>WHWLO</sub>	Sector Erase Time-Out		100		μs
t <sub>Q7VQV</sub>	DQ7 Valid to Output (DQ7-DQ0) Valid (Data Polling) <sup>(2)</sup>			30	ns

Note: 1. Programmed to all zero before erase.

2. The polling status, DQ7, is valid t<sub>Q7VQV</sub> time units before the data byte, DQ0-DQ7, is valid for reading.

Figure 54. Peripheral I/O READ Timing



AI06610

Table 123. Port A Peripheral Data Mode READ Timing (5V PSD Module)

Symbol	Parameter	Conditions	Min	Max	Turbo Off	Unit
$t_{AVQV-PA}$	Address Valid to Data Valid	(Note 1)		37	+ 10	ns
$t_{SLQV-PA}$	$\overline{CSI}$ Valid to Data Valid			27	+ 10	ns
$t_{RLQV-PA}$	$\overline{RD}$ to Data Valid	(Note 2)		32		ns
$t_{DVQV-PA}$	Data In to Data Out Valid			22		ns
$t_{RHQZ-PA}$	$\overline{RD}$ to Data High-Z			23		ns

Note: 1. Any input used to select Port A Data Peripheral Mode.  
 2. Data is already stable on Port A.

Table 124. Port A Peripheral Data Mode READ Timing (3V PSD Module)

Symbol	Parameter	Conditions	Min	Max	Turbo Off	Unit
$t_{AVQV-PA}$	Address Valid to Data Valid	(Note 1)		50	+ 20	ns
$t_{SLQV-PA}$	$\overline{CSI}$ Valid to Data Valid			37	+ 20	ns
$t_{RLQV-PA}$	$\overline{RD}$ to Data Valid	(Note 2)		45		ns
$t_{DVQV-PA}$	Data In to Data Out Valid			38		ns
$t_{RHQZ-PA}$	$\overline{RD}$ to Data High-Z			36		ns

Note: 1. Any input used to select Port A Data Peripheral Mode.  
 2. Data is already stable on Port A.

Figure 55. Peripheral I/O WRITE Timing

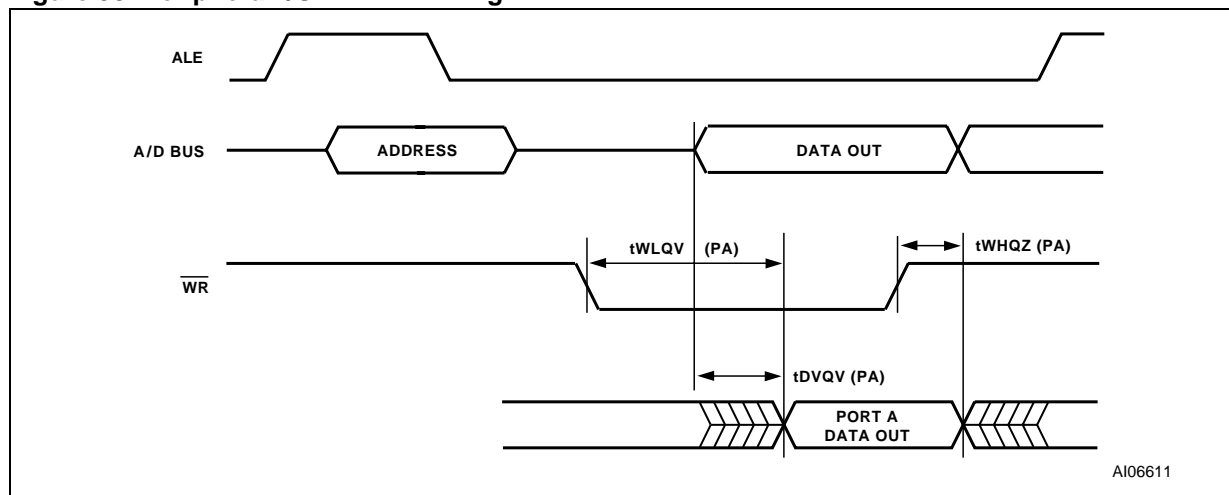


Table 125. Port A Peripheral Data Mode WRITE Timing (5V PSD Module)

Symbol	Parameter	Conditions	Min	Max	Unit
$t_{WLQV-PA}$	$\overline{WR}$ to Data Propagation Delay			25	ns
$t_{DVQV-PA}$	Data to Port A Data Propagation Delay	(Note 1)		22	ns
$t_{WHQZ-PA}$	$\overline{WR}$ Invalid to Port A Tri-state			20	ns

Note: 1. Data stable on Port 0 pins to data on Port A.

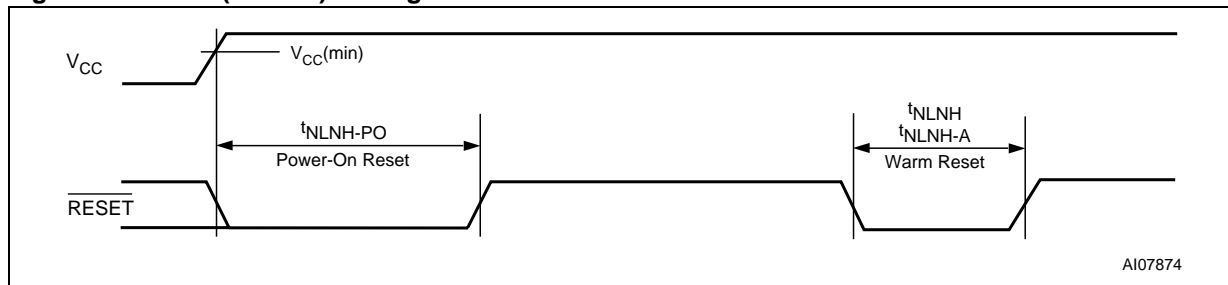
Table 126. Port A Peripheral Data Mode WRITE Timing (3V PSD Module)

Symbol	Parameter	Conditions	Min	Max	Unit
$t_{WLQV-PA}$	$\overline{WR}$ to Data Propagation Delay			42	ns
$t_{DVQV-PA}$	Data to Port A Data Propagation Delay	(Note 1)		38	ns
$t_{WHQZ-PA}$	$\overline{WR}$ Invalid to Port A Tri-state			33	ns

Note: 1. Data stable on Port 0 pins to data on Port A.



**Figure 56. Reset (RESET) Timing**



**Table 127. Reset (RESET) Timing (5V PSD Module)**

Symbol	Parameter	Conditions	Min	Max	Unit
t <sub>NLNH</sub>	RESET Active Low Time <sup>(1)</sup>		150		ns
t <sub>NLNH-PO</sub>	Power-on Reset Active Low Time		1		ms
t <sub>NLNH-A</sub>	Warm RESET <sup>(2)</sup>		25		μs
t <sub>OPR</sub>	RESET High to Operational Device			120	ns

Note: 1. Reset (RESET) does not reset Flash memory Program or Erase cycles.  
 2. Warm RESET aborts Flash memory Program or Erase cycles, and puts the device in READ Mode.

**Table 128. Reset (RESET) Timing (3V PSD Module)**

Symbol	Parameter	Conditions	Min	Max	Unit
t <sub>NLNH</sub>	RESET Active Low Time <sup>(1)</sup>		300		ns
t <sub>NLNH-PO</sub>	Power-on Reset Active Low Time		1		ms
t <sub>NLNH-A</sub>	Warm RESET <sup>(2)</sup>		25		μs
t <sub>OPR</sub>	RESET High to Operational Device			300	ns

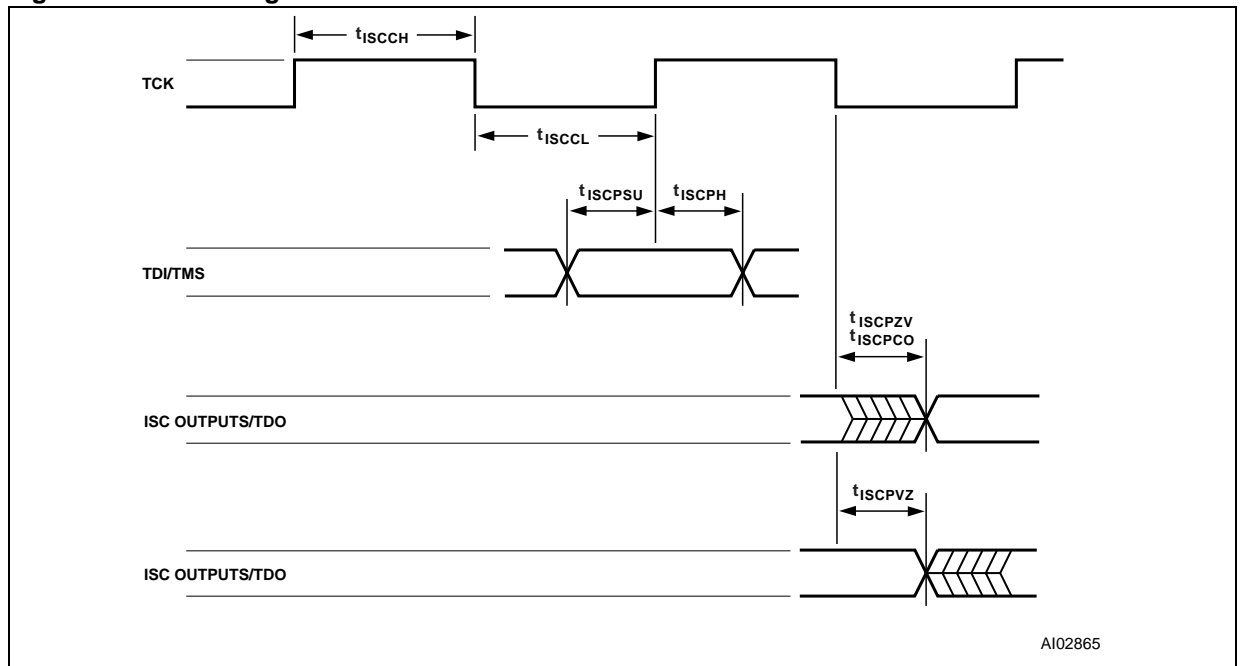
Note: 1. Reset (RESET) does not reset Flash memory Program or Erase cycles.  
 2. Warm RESET aborts Flash memory Program or Erase cycles, and puts the device in READ Mode.

**Table 129. V<sub>STBYON</sub> Definitions Timing (5V, 3V PSD Modules)**

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
t <sub>BVBH</sub>	V <sub>STBY</sub> Detection to V <sub>STBYON</sub> Output High	(Note 1)		20		μs
t <sub>BXBL</sub>	V <sub>STBY</sub> Off Detection to V <sub>STBYON</sub> Output Low	(Note 1)		20		μs

Note: 1. V<sub>STBYON</sub> timing is measured at V<sub>CC</sub> ramp rate of 2ms.

Figure 57. ISC Timing



AI02865

Table 130. ISC Timing (5V PSD Module)

Symbol	Parameter	Conditions	Min	Max	Unit
$t_{ISCCF}$	Clock (TCK, PC1) Frequency (except for PLD)	(Note 1)		20	MHz
$t_{ISCCH}$	Clock (TCK, PC1) High Time (except for PLD)	(Note 1)	23		ns
$t_{ISCCL}$	Clock (TCK, PC1) Low Time (except for PLD)	(Note 1)	23		ns
$t_{ISCCFP}$	Clock (TCK, PC1) Frequency (PLD only)	(Note 2)		2	MHz
$t_{ISCCHP}$	Clock (TCK, PC1) High Time (PLD only)	(Note 2)	240		ns
$t_{ISCCLP}$	Clock (TCK, PC1) Low Time (PLD only)	(Note 2)	240		ns
$t_{ISCPSU}$	ISC Port Set Up Time		7		ns
$t_{ISCPH}$	ISC Port Hold Up Time		5		ns
$t_{ISPCO}$	ISC Port Clock to Output			21	ns
$t_{ISCPZV}$	ISC Port High-Impedance to Valid Output			21	ns
$t_{ISCPVZ}$	ISC Port Valid Output to High-Impedance			21	ns

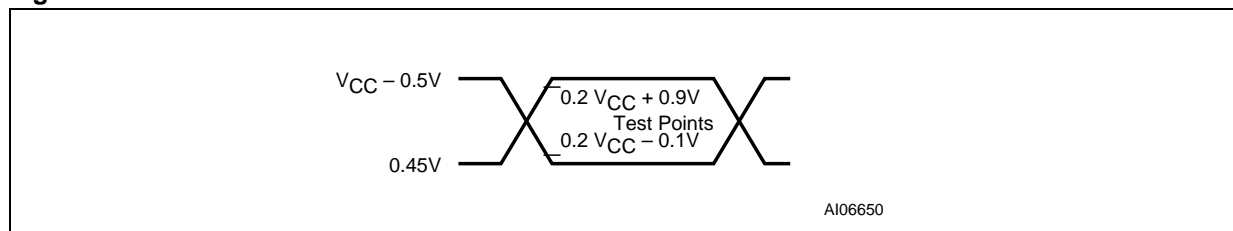
Note: 1. For non-PLD Programming, Erase or in ISC By-pass Mode.  
 2. For Program or Erase PLD only.

**Table 131. ISC Timing (3V PSD Module)**

Symbol	Parameter	Conditions	Min	Max	Unit
t <sub>ISCCF</sub>	Clock (TCK, PC1) Frequency (except for PLD)	(Note 1)		12	MHz
t <sub>ISCCH</sub>	Clock (TCK, PC1) High Time (except for PLD)	(Note 1)	40		ns
t <sub>ISCCL</sub>	Clock (TCK, PC1) Low Time (except for PLD)	(Note 1)	40		ns
t <sub>ISCCFP</sub>	Clock (TCK, PC1) Frequency (PLD only)	(Note 2)		2	MHz
t <sub>ISCCHP</sub>	Clock (TCK, PC1) High Time (PLD only)	(Note 2)	240		ns
t <sub>ISCCLP</sub>	Clock (TCK, PC1) Low Time (PLD only)	(Note 2)	240		ns
t <sub>ISCPSTU</sub>	ISC Port Set Up Time		12		ns
t <sub>ISCPH</sub>	ISC Port Hold Up Time		5		ns
t <sub>ISPCO</sub>	ISC Port Clock to Output			30	ns
t <sub>ISCPZV</sub>	ISC Port High-Impedance to Valid Output			30	ns
t <sub>ISCPVZ</sub>	ISC Port Valid Output to High-Impedance			30	ns

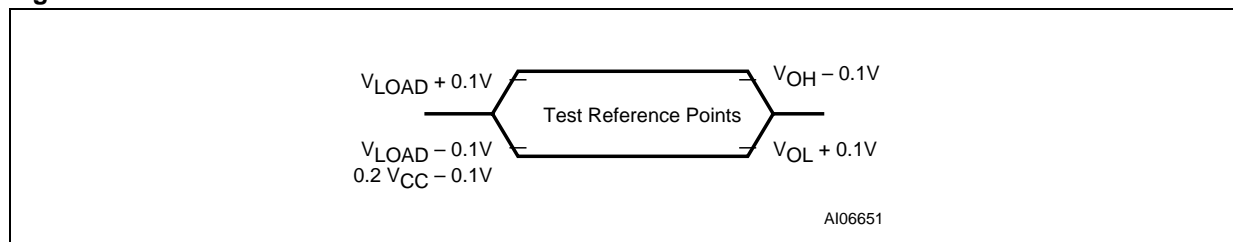
Note: 1. For non-PLD Programming, Erase or in ISC By-pass Mode.  
 2. For Program or Erase PLD only.

**Figure 58. MCU Module AC Measurement I/O Waveform**



Note: AC inputs during testing are driven at V<sub>CC</sub>-0.5V for a logic '1,' and 0.45V for a logic '0.'  
 Timing measurements are made at V<sub>IH</sub>(min) for a logic '1,' and V<sub>IL</sub>(max) for a logic '0'

**Figure 59. PSD Module AC Float I/O Waveform**



Note: For timing purposes, a Port pin is considered to be no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded V<sub>OH</sub> or V<sub>OL</sub> level occurs  
 I<sub>OL</sub> and I<sub>OH</sub> ≥ 20mA

Figure 60. External Clock Cycle

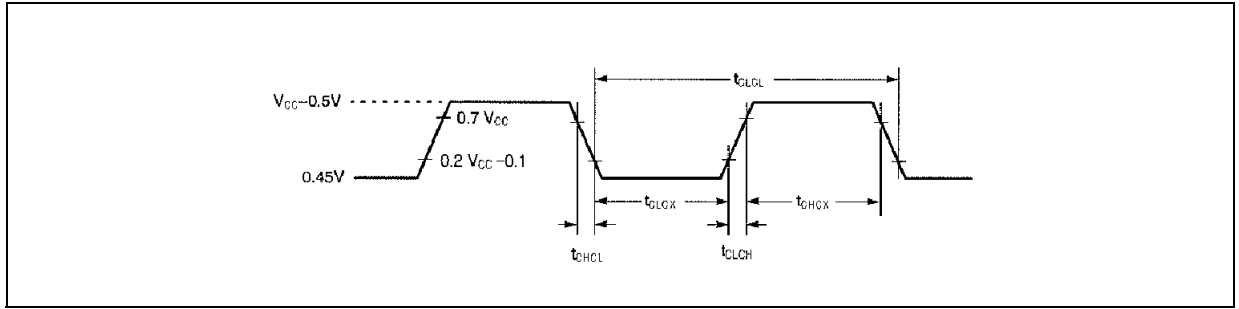
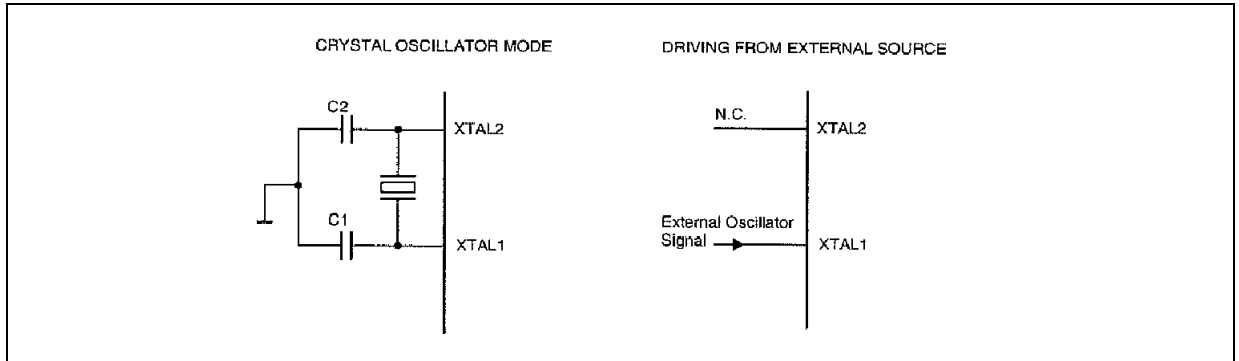


Figure 61. Recommended Oscillator Circuits



Note: C1, C2 = 30pF ± 10pF for crystals

For ceramic resonators, contact resonator manufacturer

Oscillation circuit is designed to be used either with a ceramic resonator or crystal oscillator. Since each crystal and ceramic resonator have their own characteristics, the user should consult the crystal manufacturer for appropriate values of external components.

Figure 62. PSD Module AC Measurement I/O Waveform

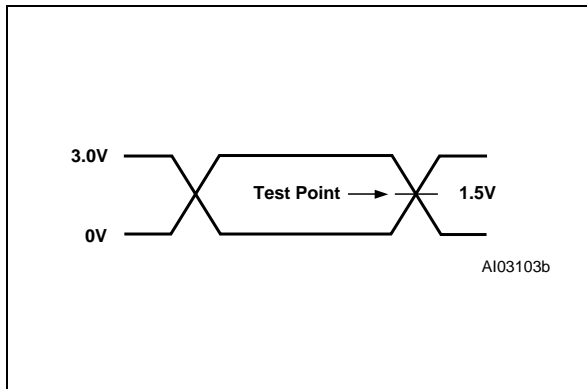


Figure 63. PSD Module AC Measurement Load Circuit

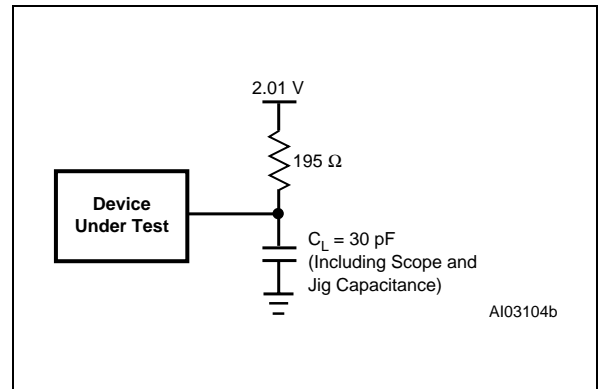


Table 132. I/O Pin Capacitance

Symbol	Parameter	Test Condition	Typ. <sup>2</sup>	Max.	Unit
C <sub>IN</sub>	Input Capacitance (for input pins)	V <sub>IN</sub> = 0V	4	6	pF
C <sub>OUT</sub>	Output Capacitance (for input/output pins)	V <sub>OUT</sub> = 0V	8	12	pF

Note: 1. Sampled only, not 100% tested.

2. Typical values are for T<sub>A</sub> = 25°C and nominal supply voltages.

**PART NUMBERING**

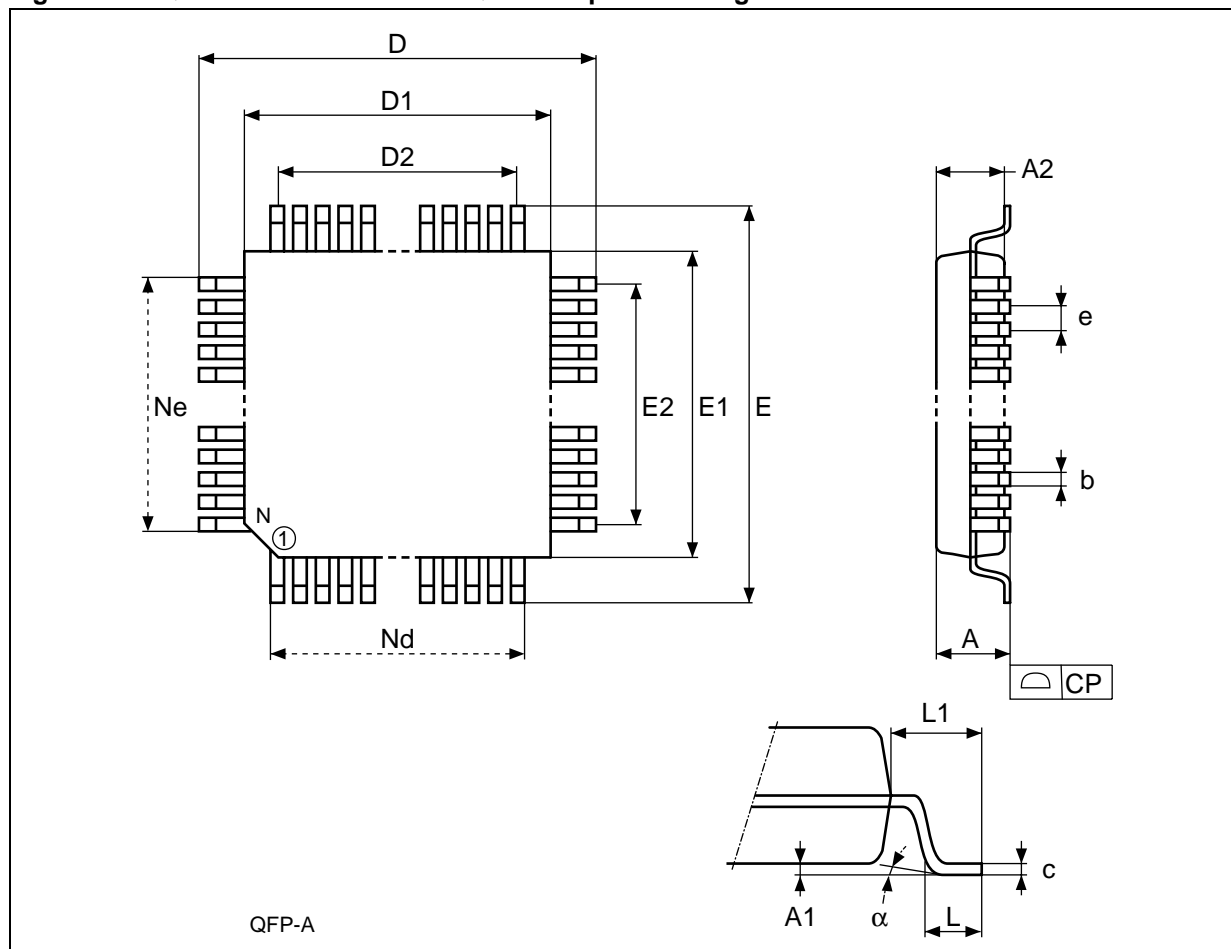
**Table 133. Ordering Information Scheme**

Example:	μPSD	33	3	4	D	V	-	24	U	6	T
<b>Device Type</b>											
μPSD = Microcontroller PSD											
<b>Family</b>											
32 = 8032 core											
33 = Turbo core											
<b>SRAM Size</b>											
1 = 16Kbit											
3 = 64Kbit											
5 = 256Kbit											
<b>Main Flash Memory Size</b>											
2 = 512Kbit											
3 = 1Mbit											
4 = 2Mbit											
<b>IP Mix</b>											
D = IP Mix: I <sup>2</sup> C, SPI, UART (2), IrDA, ADC, Supervisor, PCA											
<b>Operating Voltage</b>											
blank = V <sub>CC</sub> = 4.5 to 5.5V											
V = V <sub>CC</sub> = 3.0 to 3.6V											
<b>Speed</b>											
-24 = 24MHz											
-40 = 40MHz											
<b>Package</b>											
T = 52-pin TQFP											
U = 80-pin TQFP											
<b>Temperature Range</b>											
1 = 0 to 70°C											
6 = -40 to 85°C											
<b>Shipping Option</b>											
Tape & Reel Packing = T											

For a list of available options (e.g., Speed, Package) or for further information on any aspect of this device, please contact the ST Sales Office nearest to you.

PACKAGE MECHANICAL INFORMATION

Figure 64. TQFP52 – 52-lead Plastic Quad Flatpack Package Outline



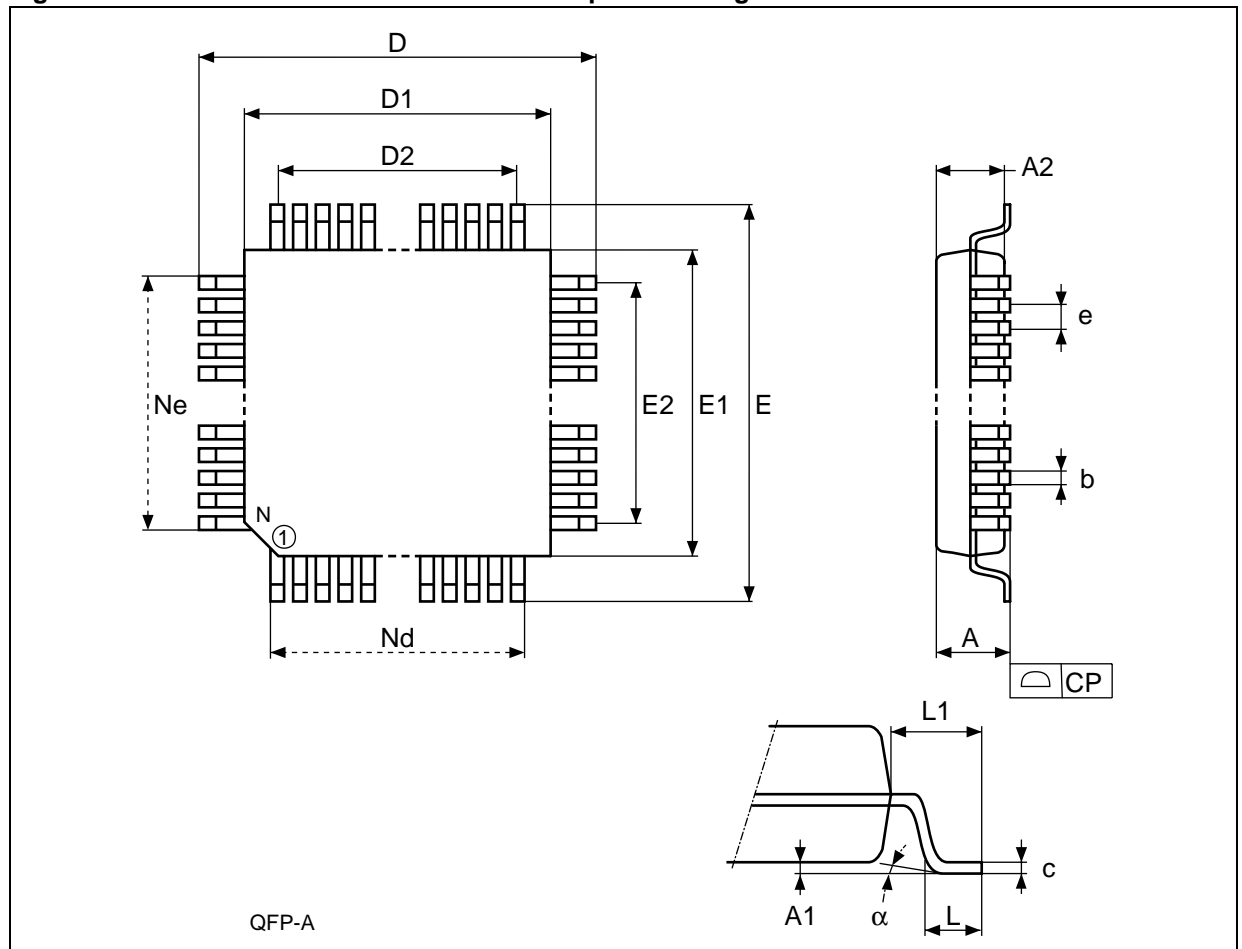
QFP-A

Note: Drawing is not to scale.

**Table 134. TQFP52 – 52-lead Plastic Quad Flatpack Package Mechanical Data**

Symb	mm			inches		
	Typ	Min	Max	Typ	Min	Max
A	–	–	1.75	–	–	0.069
A1	–	0.05	0.20	–	0.002	0.008
A2	–	1.25	1.55	–	0.049	0.061
b	–	0.20	0.40	–	0.008	0.016
c	–	0.07	0.23	–	0.002	0.009
D	12.00	–	–	0.473	–	–
D1	10.00	–	–	0.394	–	–
D2						
E	12.00	–	–	0.473	–	–
E1	10.00	–	–	0.394	–	–
E2	7.80			0.307		
e	0.65	–	–	0.026	–	–
L	–	0.45	0.75	–	0.018	0.030
L1	1.00	–	–	0.039	–	–
α	–	0°	7°	–	0°	7°
n	52			52		
Nd	13			13		
Ne	13			13		
CP	–	–	0.10	–	–	0.004

Figure 65. TQFP80 – 80-lead Plastic Quad Flatpack Package Outline



Note: Drawing is not to scale.



**Table 135. TQFP80 – 80-lead Plastic Quad Flatpack Package Mechanical Data**

Symb	mm			inches		
	Typ	Min	Max	Typ	Min	Max
A	–	–	1.60	–	–	0.063
A1	–	0.05	0.15	–	0.002	0.006
A2	1.40	1.35	1.45	0.055	0.053	0.057
b	0.22	0.17	0.27	0.009	0.007	0.011
c	–	0.09	0.20	–	0.004	0.008
D	14.00	–	–	0.551	–	–
D1	12.00	–	–	0.472	–	–
D2	9.50	–	–	0.374	–	–
E	14.00	–	–	0.473	–	–
E1	12.00	–	–	0.394	–	–
E2	9.50	–	–	0.374	–	–
e	0.50	–	–	0.020	–	–
L	0.60	0.45	0.75	0.024	0.018	0.030
L1	1.00	–	–	0.039	–	–
α	3.5	0°	7°	3.5	0°	7°
n	80			80		
Nd	20			20		
Ne	20			20		
CP	–	–	0.08	–	–	0.003

**REVISION HISTORY**

**Table 136. Document Revision History**

Date	Rev. #	Revision Details
July 2003	1.0	First Issue

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

The ST logo is registered trademark of STMicroelectronics  
All other names are the property of their respective owners.

© 2003 STMicroelectronics - All Rights Reserved

STMicroelectronics GROUP OF COMPANIES

Australia - Brazil - Canada - China - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia -  
Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - U.S.A.

[www.st.com](http://www.st.com)