Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	8032
Core Size	8-Bit
Speed	40MHz
Connectivity	I²C, IrDA, SPI, UART/USART
Peripherals	LVD, POR, PWM, WDT
Number of I/O	45
Program Memory Size	288KB (288K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	32K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/upsd3354d-40u6

Contents

Table 2. Pin definitions (continued)

Port pin	Signal name	80-Pin num.	52-Pin num.(1)	In/Out	Function		
					Basic	Alternate 1	Alternate 2
P4.6	SPITXD TCM5	19	15	I/O	General I/O port pin	PCA1-TCM5	SPI Transmit (SPITxD)
P4.7	SPISEL PCACLK1	18	14	I/O	General I/O port pin	PCACLK1	SPI Slave Select ($\overline{SPISEL}$)
AV_{REF}		70	N/A	I	Reference voltage input for ADC. Connect AV_{REF} to V_{CC} if the ADC is not used.		
$\overline{RD}$		65	N/A	O	READ Signal, external bus		
$\overline{WR}$		62	N/A	O	WRITE Signal, external bus		
$\overline{PSEN}$		63	N/A	O	$\overline{PSEN}$ Signal, external bus		
ALE		4	N/A	O	Address Latch signal, external bus		
$\overline{RESET\_IN}$		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		

4 Memory organization

The 8032 MCU core views memory on the MCU module as “internal” memory and it views memory on the PSD module as “external” memory, see *Figure 5*

Internal memory on the MCU module consists of DATA, IDATA, and SFRs. These standard 8032 memories reside in 384 bytes of SRAM located at a fixed address space starting at address 0x0000.

External memory on the PSD module consists of four types: main Flash (64, 128, or 256 Kbytes), a smaller secondary Flash (16 or 32 Kbytes), SRAM (2, 8, or 32 Kbytes), and a block of PSD module control registers called CSIOP (256 bytes). These external memories reside at programmable address ranges, specified using the software tool PSDsoft Express. See the *Section 27: PSD module on page 164* of this document for more details on these memories.

External memory is accessed by the 8032 in two separate 64 Kbyte address spaces. One address space is for program memory and the other address space is for data memory. Program memory is accessed using the 8032 signal, $\overline{\text{PSEN}}$. Data memory is accessed using the 8032 signals, $\overline{\text{RD}}$ and $\overline{\text{WR}}$. If the 8032 needs to access more than 64 Kbytes of external program or data memory, it must use paging (or banking) techniques provided by the Page register in the PSD module.

Note: When referencing program and data memory spaces, it has nothing to do with 8032 internal SRAM areas of DATA, IDATA, and SFR on the MCU module. Program and data memory spaces only relate to the external memories on the PSD module.

External memory on the PSD module can overlap the internal SRAM memory on the MCU module in the same physical address range (starting at 0x0000) without interference because the 8032 core does not assert the $\overline{\text{RD}}$ or $\overline{\text{WR}}$ signals when accessing internal SRAM.

Figure 5. UPSD33xx memories

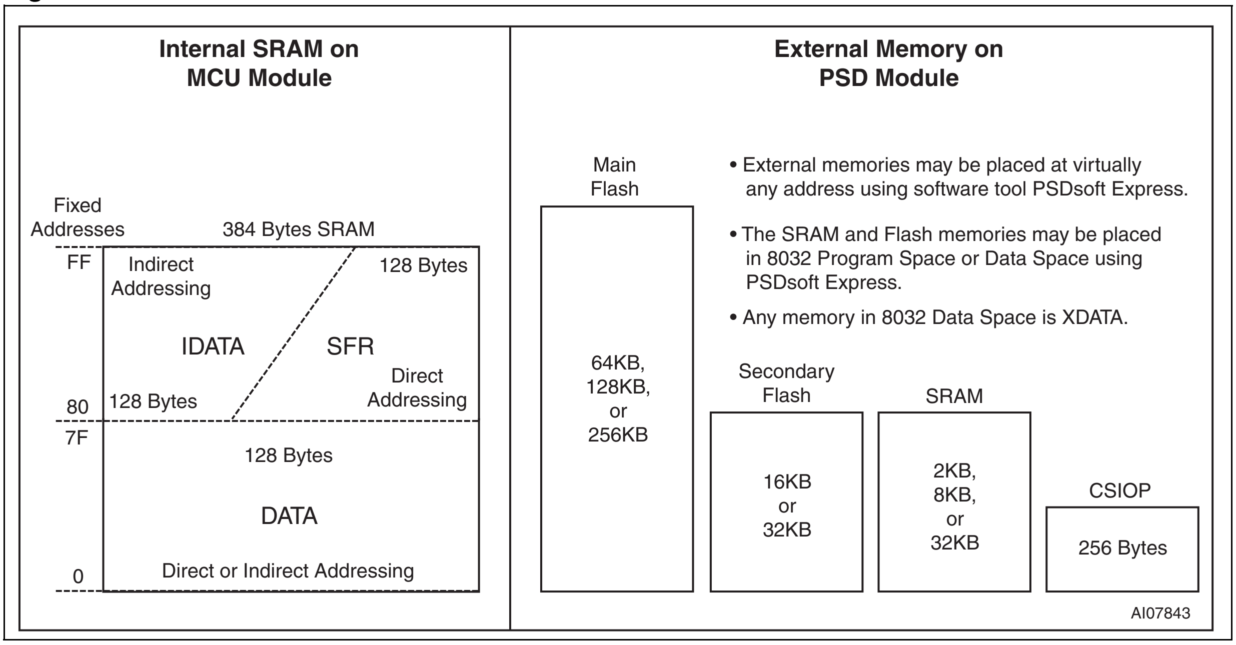


Figure 8. PFQ operation on multi-cycle instructions

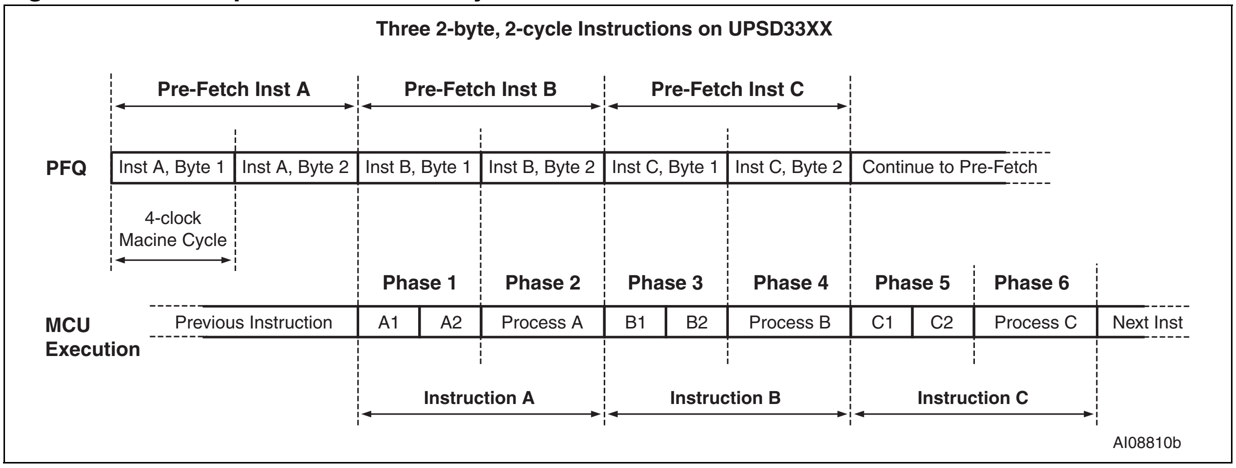


Figure 9. UPSD33xx multi-cycle instructions compared to standard 8032

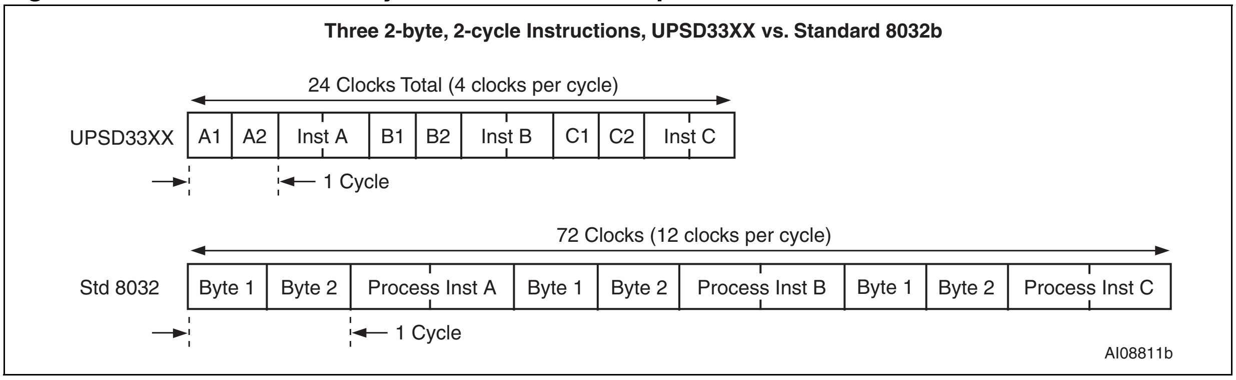


Table 5. SFR memory map with direct address and reset value

SFR addr (hex)	SFR name	Bit name and <Bit Address>								Reset value (hex)	Reg. descr. with link
		7	6	5	4	3	2	1	0		
80	RESERVED										
81	SP	SP[7:0]								07	*Section 7.1 on page 37*
82	DPL	DPL[7:0]								00	*Section 7.2 on page 37*
83	DPH	DPH[7:0]								00	
84	RESERVED										
85	DPTC	–	AT	–	–	–	DPSEL[2:0]			00	*Table 13 on page 56*
86	DPTM	–	–	–	–	MD1[1:0]		MD0[1:0]		00	*Table 15 on page 57*
87	PCON	SMOD0	SMOD1	–	POR	RCLK1	TCLK1	PD	IDLE	00	*Table 31 on page 72*
88(1)	TCON	TF1 <8Fh>	TR1 <8Eh>	TF0 <8Dh>	TR0 <8Ch>	IE1 <8Bh>	IT1 <8Ah>	IE0 <89h>	IT0 <88h>	00	*Table 54 on page 95*
89	TMOD	GATE	C/T	M1	M0	GATE	C/T	M1	M0	00	*Table 56 on page 97*
8A	TL0	TL0[7:0]								00	*Section 20.1 on page 94*
8B	TL1	TL1[7:0]								00	
8C	TH0	TH0[7:0]								00	
8D	TH1	TH1[7:0]								00	
8E	P1SFS0	P1SFS0[7:0]								00	*Table 41 on page 83*
8F	P1SFS1	P1SFS1[7:0]								00	*Table 42 on page 83*
90(1)	P1	P1.7 <97h>	P1.6 <96h>	P1.5 <95h>	P1.4 <94h>	P1.3 <93h>	P1.2 <92h>	P1.1 <91h>	P1.0 <90h>	FF	*Table 33 on page 80*
91	P3SFS	P3SFS[7:0]								00	*Table 39 on page 83*
92	P4SFS0	P4SFS0[7:0]								00	*Table 44 on page 84*
93	P4SFS1	P4SFS1[7:0]								00	*Table 45 on page 84*
94	ADCPS	–	–	–	–	ADCCE	ADCPS[2:0]			00	*Table 95 on page 152*
95	ADAT0	ADATA[7:0]								00	*Table 96 on page 152*
96	ADAT1	–	–	–	–	–	–	ADATA[9:8]		00	*Table 97 on page 152*
97	ACON	AINTF	AINTEN	ADEN	ADS[2:0]			ADST	ADSF	00	*Table 93 on page 151*
98(1)	SCON0	SM0 <9Fh>	SM1 <9Eh>	SM2 <9Dh>	REN <9Ch>	TB8 <9Bh>	RB8 <9Ah>	TI <99h>	RI <9h8>	00	*Table 63 on page 108*
99	SBUF0	SBUF0[7:0]								00	*Figure 24 on page 104*
9A	RESERVED										
9B	RESERVED										

19 Supervisory functions

Supervisory circuitry on the MCU module will issue an internal reset signal to the MCU module and simultaneously to the PSD module as a result of any of the following four events:

- The external RESET_IN pin is asserted
- The low voltage Detect (LVD) circuitry has detected a voltage on V_{CC} below a specific threshold (power-on or voltage sags)
- The JTAG debug interface has issued a reset command
- The Watch Dog Timer (WDT) has timed out

The resulting internal reset signal, MCU_RESET, will force the 8032 into a known reset state while asserted, and then 8032 program execution will jump to the reset vector at program address 0000h just after MCU_RESET is deasserted. The MCU module will also assert an active low internal reset signal, RESET, to the PSD module. If needed, the signal RESET can be driven out to external system components through any PLD output pin on the PSD module. When driving this “RESET_OUT” signal from a PLD output, the user can choose to make it either active-high or active-low logic, depending on the PLD equation.

19.1 External reset input pin, RESET_IN

The RESET_IN pin can be connected directly to a mechanical reset switch or other device which pulls the signal to ground to invoke a reset.

RESET_IN is pulled up internally and enters a Schmitt trigger input buffer with a voltage hysteresis of V_{RST_HYS} for immunity to the effects of slow signal rise and fall times, as shown in *Figure 19*. RESET_IN is also filtered to reject a voltage spike less than a duration of t_{RST_FIL}. The RESET_IN signal must be maintained at a logic '0' for at least a duration of $t_{RST\_LO\_IN}$ while the oscillator is running. The resulting MCU_RESET signal will last only as long as the RESET_IN signal is active (it is not stretched). Refer to the Supervisor AC specifications in *Table 187 on page 262* at the end of this document for these parameter values.

Figure 19. Supervisor Reset generation

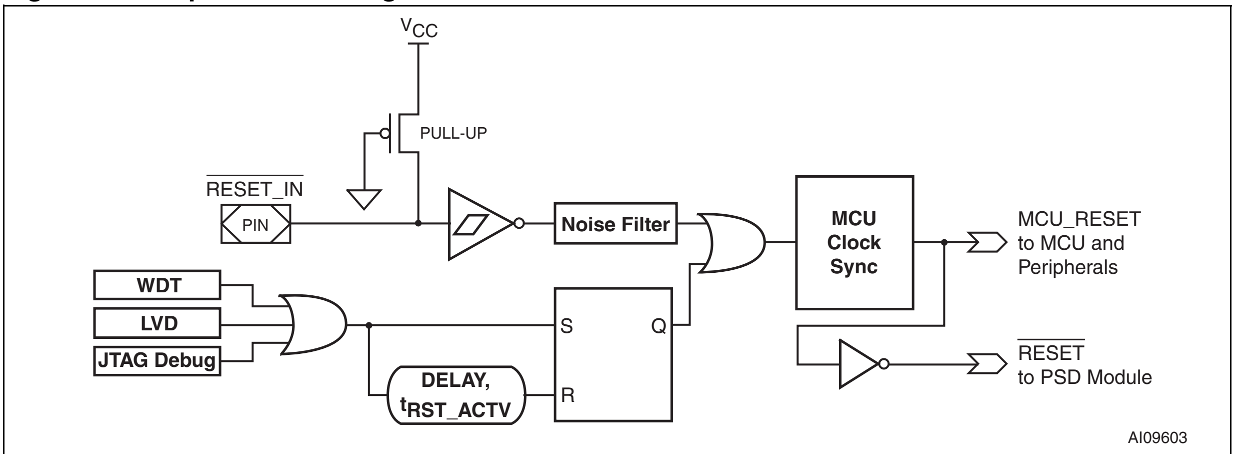


Table 64. SCON0 register bit definition (continued)

Bit	Symbol	R/W	Definition
1	TI	R,W	**Transmit Interrupt flag.** Causes interrupt at end of 8th bit time when transmitting in mode 0, or at beginning of stop bit transmission in other modes. Must clear flag with firmware.
0	RI	R,W	**Receive Interrupt flag.** Causes interrupt at end of 8th bit time when receiving in mode 0, or halfway through stop bit reception in other modes (see SM2 for exception). Must clear this flag with firmware.

Table 65. SCON1: Serial Port UART1 Control register (SFR D8h, reset value 00h)

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

Table 66. SCON1 register bit definition

Bit	Symbol	R/W	Definition
7	SM0	R,W	**Serial Mode Select**, See *Table 62 on page 107*. Important, notice bit order of SM0 and SM1. **[SM0:SM1] = 00b**, mode 0 **[SM0:SM1] = 01b**, mode 1 **[SM0:SM1] = 10b**, mode 2 **[SM0:SM1] = 11b**, mode 3
6	SM1	R,W	
5	SM2	R,W	**Serial Multiprocessor Communication Enable.** **Mode 0**: SM2 has no effect but should remain 0. **Mode 1**: If SM2 = 0 then stop bit ignored. SM2 =1 then RI active if stop bit = 1. **Mode 2 and 3**: Multiprocessor Comm Enable. If SM2=0, 9th bit is ignored. If SM2=1, RI active when 9th bit = 1.
4	REN	R,W	**Receive Enable.** If REN=0, UART reception disabled. If REN=1, reception is enabled
3	TB8	R,W	TB8 is assigned to the 9th transmission bit in mode 2 and 3. Not used in mode 0 and 1.
2	RB8	R,W	**Mode 0**: RB8 is not used. **Mode 1**: If SM2 = 0, the RB8 is the level of the received stop bit. **Mode 2 and 3**: RB8 is the 9th data bit that was received in mode 2 and 3.

Table 67. Commonly used baud rates generated from Timer 1 (continued)

UART mode	f_{OSC} (MHz)	Desired baud rate	Resultant baud rate	Baud rate deviation	SMOD bit in PCON	Timer 1		
						C/$\overline{T}$ Bit in TMOD	Timer mode in TMOD	TH1 Reload value (hex)
Modes 1 or 3	3.6864	9600 Hz	9600 Hz	0	1	0	2	FE
Modes 1 or 3	1.8432	9600 Hz	9600 Hz	0	1	0	2	FF
Modes 1 or 3	1.8432	4800 Hz	4800 Hz	0	1	0	2	FE

21.4 More about UART mode 0

Refer to the block diagram in *Figure 27 on page 113*, and timing diagram in *Figure 28 on page 113*.

Transmission is initiated by any instruction which writes to the SFR named SBUF. At the end of a write operation to SBUF, a 1 is loaded into the 9th position of the transmit shift register and tells the TX Control unit to begin a transmission. Transmission begins on the following MCU machine cycle, when the "SEND" signal is active in *Figure 28 on page 113*.

SEND enables the output of the shift register to the alternate function on the port containing pin RxD, and also enables the SHIFT CLOCK signal to the alternate function on the port containing the pin, TxD. At the end of each SHIFT CLOCK in which SEND is active, the contents of the transmit shift register are shifted to the right one position.

As data bits shift out to the right, zeros come in from the left. When the MSB of the data byte is at the output position of the shift register, then the '1' that was initially loaded into the 9th position, is just to the left of the MSB, and all positions to the left of that contain zeros. This condition flags the TX Control unit to do one last shift, then deactivate SEND, and then set the interrupt flag TI. Both of these actions occur at S1P1.

Reception is initiated by the condition REN = 1 and RI = 0. At the end of the next MCU machine cycle, the RX Control unit writes the bits 11111110 to the receive shift register, and in the next clock phase activates RECEIVE. RECEIVE enables the SHIFT CLOCK signal to the alternate function on the port containing the pin, TxD. Each pulse of SHIFT CLOCK moves the contents of the receive shift register one position to the left while RECEIVE is active. The value that comes in from the right is the value that was sampled at the RxD pin. As data bits come in from the right, 1s shift out to the left. When the 0 that was initially loaded into the right-most position arrives at the left-most position in the shift register, it flags the RX Control unit to do one last shift, and then it loads SBUF. After this, RECEIVE is cleared, and the receive interrupt flag RI is set.

Figure 27. UART mode 0, block diagram

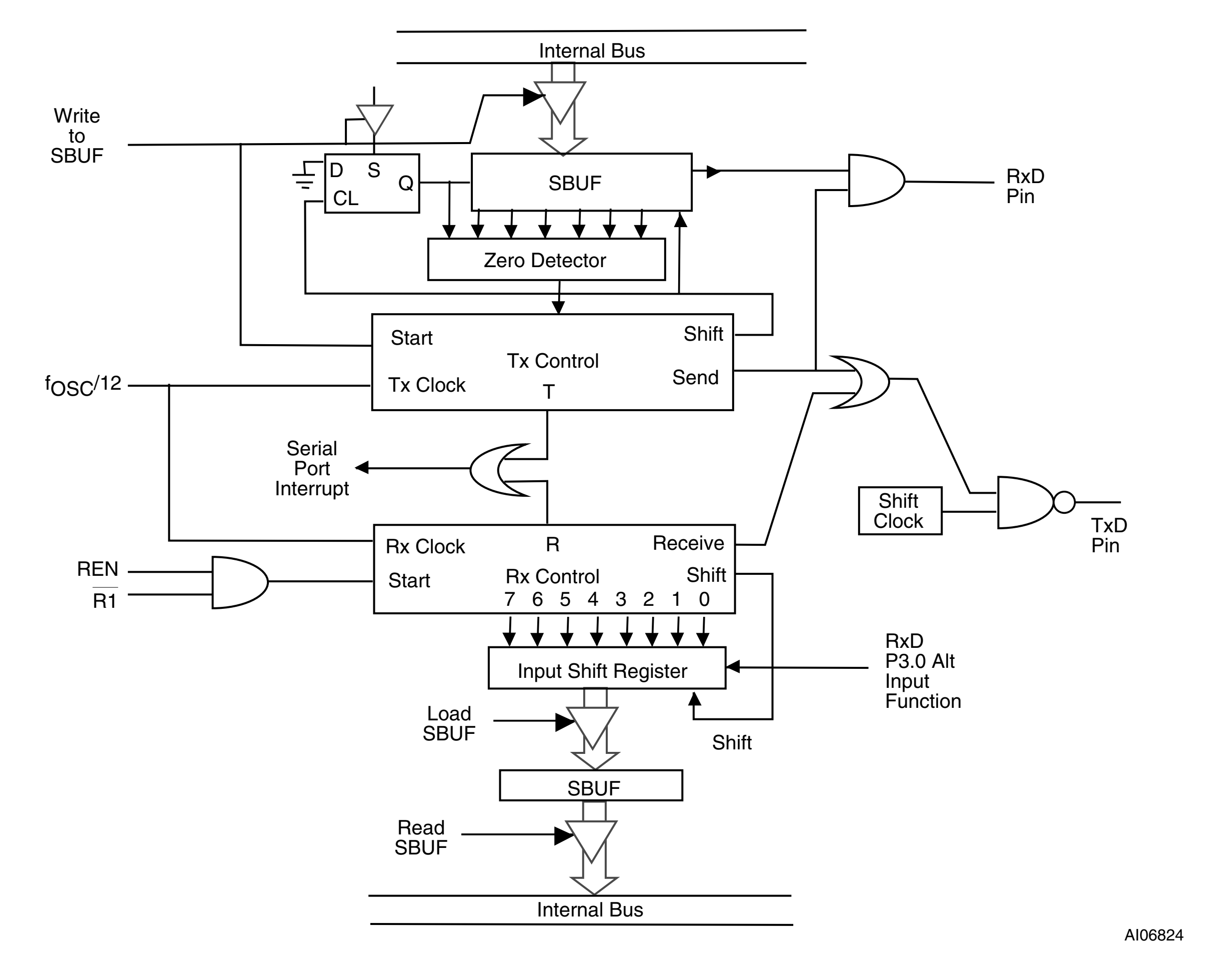


Figure 28. UART mode 0, timing diagram

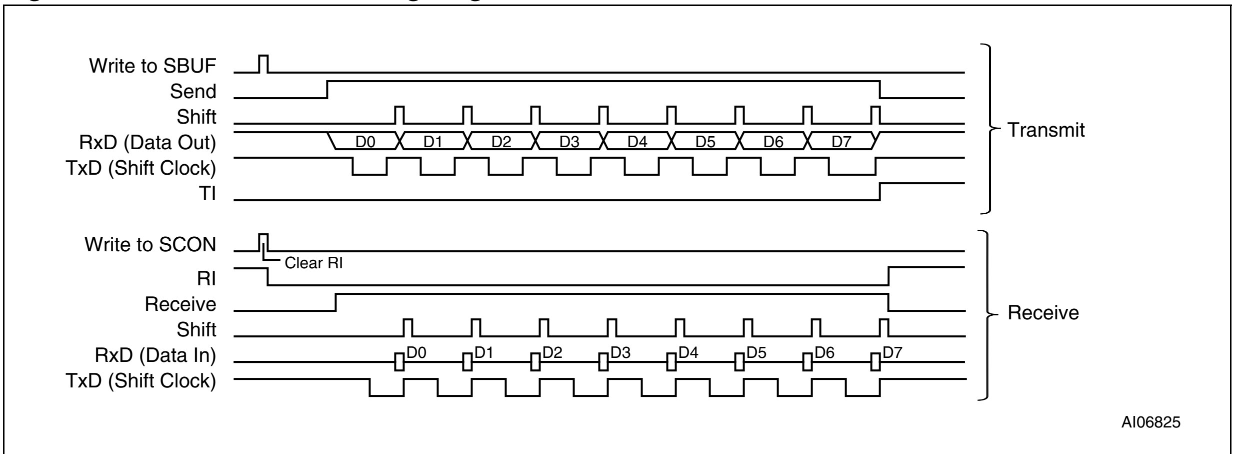


Figure 39. I²C interface SIOE block diagram

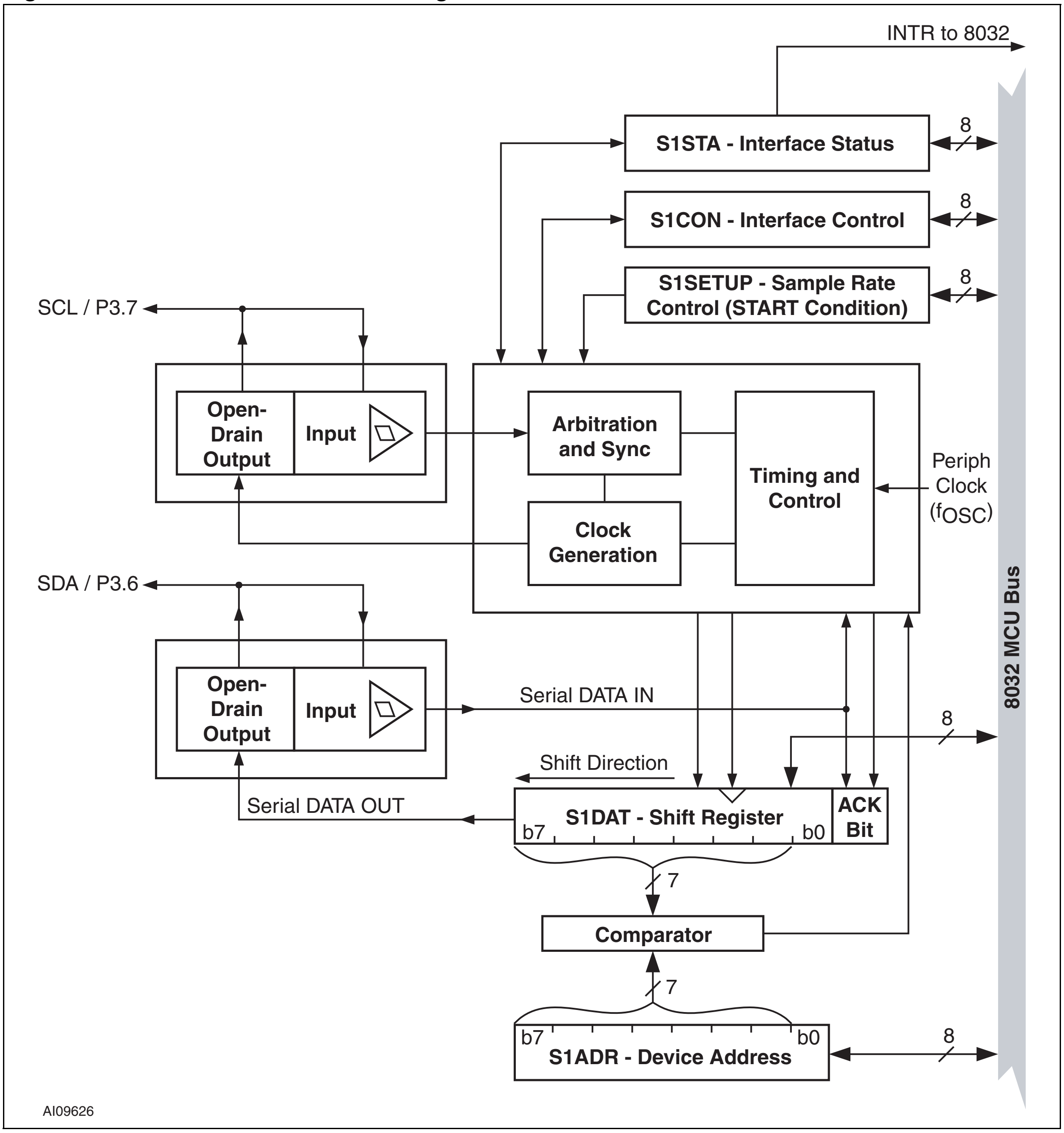


Table 79. S1ADR register bit definition

Bit	Symbol	R/W	Function
7:1	SLA[6:0]	R/W	Stores desired 7-bit device address, used when SIOE is in Slave mode.
0	–	–	Not used

23.12 I²C START sample setting (S1SETUP)

The S1SETUP register (*Table 80 on page 132*) determines how many times an I²C bus START condition will be sampled before the SIOE validates the START condition, giving the SIOE the ability to reject noise or illegal transmissions.

Because the minimum duration of an START condition varies with I²C bus speed (f_{SCL}), and also because the UPSD33xx may be operated with a wide variety of frequencies (f_{OSC}), it is necessary to scale the number of samples per START condition based on f_{OSC} and f_{SCL}.

In Slave mode, the SIOE recognizes the beginning of a START condition when it detects a 1-to-0 transition on the SDA bus line while the SCL line is high (see *Figure 38 on page 124*). The SIOE must then validate the START condition by sampling the bus lines to ensure SDA remains low and SCL remains high for a minimum amount of hold time, t_{HLDSTA}. Once validated, the SIOE begins receiving the address byte that follows the START condition.

If the EN_SS Bit (in the S1SETUP register) is not set, then the SIOE will sample only once after detecting the 1-to-0 transition on SDA. This single sample is taken $1/f_{OSC}$ seconds after the initial 1-to-0 transition was detected. However, more samples should be taken to ensure there is a valid START condition.

To take more samples, the SIOE should be initialized such that the EN_SS Bit is set, and a value is written to the SMPL_SET[6:0] field of the S1SETUP register to specify how many samples to take. The goal is to take a good number of samples during the minimum START condition hold time, t_{HLDSTA}, but no so many samples that the bus will be sampled after t_{HLDSTA} expires.

Table 82 describes the relationship between the contents of S1SETUP and the resulting number of I²C bus samples that SIOE will take after detecting the 1-to-0 transition on SDA of a START condition.

Note: ***Important:*** *Keep in mind that the time between samples is always $1/f_{OSC}$.*

The minimum START condition hold time, t_{HLDSTA}, is different for the three common I²C speed categories per *Table 83 on page 133*.

Table 80. S1SETUP: I²C START Condition Sample Setup register (SFR DBh, reset value 00h)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EN_SS	SMPL_SET[6:0]						

27.1.14 I/O ports

For 80-pin UPSD33xx devices, the PSD module has 22 individually configurable I/O pins distributed over four ports (these I/O are in addition to I/O on MCU module). For 52-pin UPSD33xx devices, the PSD module has 13 individually configurable I/O pins distributed over three ports. See *Figure 73 on page 219* for I/O port pin availability on these two packages.

I/O port pins on the PSD module (Ports A, B, C, and D) are completely separate from the port pins on the MCU module (Ports 1, 3, and 4). They even have different electrical characteristics. I/O port pins on the PSD module are accessed by csiop registers, or they are controlled by PLD equations. Conversely, I/O Port pins on the MCU module are controlled by the 8032 SFR registers.

Table 113. General I/O pins on PSD module

Pkg	Port A	Port B	Port D	Port D	Total
52-pin	0	8	4	1	13
80-pin	8	8	4	2	22

Note: Four pins on Port C are dedicated to JTAG, leaving four pins for general I/O.

Each I/O pin on the PSD module can be individually configured for different functions on a pin-by-pin basis (*Figure 68 on page 208*). Following are the available functions on PSD module I/O pins.

- **MCU I/O:** 8032 controls the output state of each port pin or it reads input state of each port pin, by accessing csiop registers at run-time. The direction (in or out) of each pin is also controlled by csiop registers at run-time.
- **PLD I/O:** PSDsoft Express logic equations and pin configuration selections determine if pins are connected to OMC outputs or IMC inputs. This is a static and non-volatile configuration. Port pins connected to PLD outputs can no longer be driven by the 8032 using MCU I/O output mode.
- **Latched MCU Address Output:** Port A or Port B can output de-multiplexed 8032 address signals A0 - A7 on a pin-by-pin basis as specified in csiop registers at run-time. In addition, Port B can also be configured to output de-multiplexed A8-A15 in PSDsoft Express.
- **Data Bus Repeater:** Port A can bi-directionally buffer the 8032 data bus (de-multiplexed) for a specified address range in PSDsoft Express. This is referred to as ***Peripheral I/O mode*** in this document.
- **Open Drain Outputs:** Some port pins can function as open-drain as specified in csiop registers at run-time.
- Pins on Port D can be used for **external chip-select** outputs originating from the DPLD, without consuming OMC resources within the GPLD.

27.2.5 Alternative mapping schemes

Here are more possible memory maps for the UPSD3333.

Note: Mapping examples would be slightly different for UPSD3312, UPSD3334, and UPSD3354 because of the different sizes of individual Flash memory sectors and SRAM as defined in Table 119 on page 193.

- ***Figure 54 on page 174*** Place the larger main Flash memory into program space, but split the secondary Flash in half, placing two of it's sectors into XDATA space and remaining two sectors into program space. This method allows the designer to put IAP code (or boot code) into two sectors of secondary Flash in program space, and use the other two secondary Flash sectors for data storage, such as EEPROM emulation in XDATA space.
- ***Figure 55 on page 175*** Place both the Main and secondary Flash memories into program space for maximum code storage, with no Flash memory in XDATA space.

Figure 54. Mapping: split second Flash in half

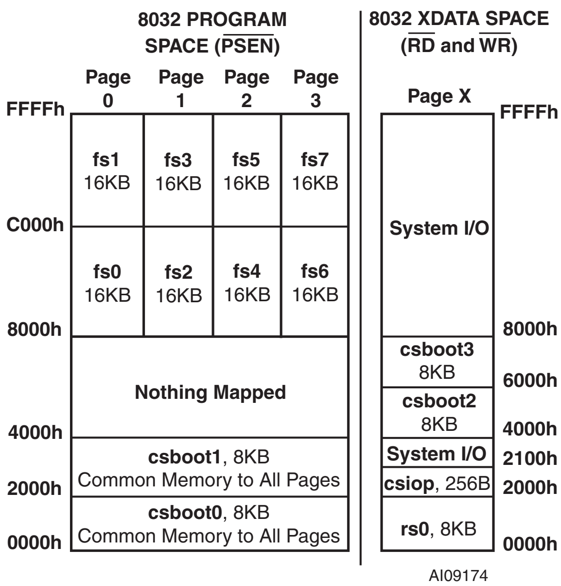
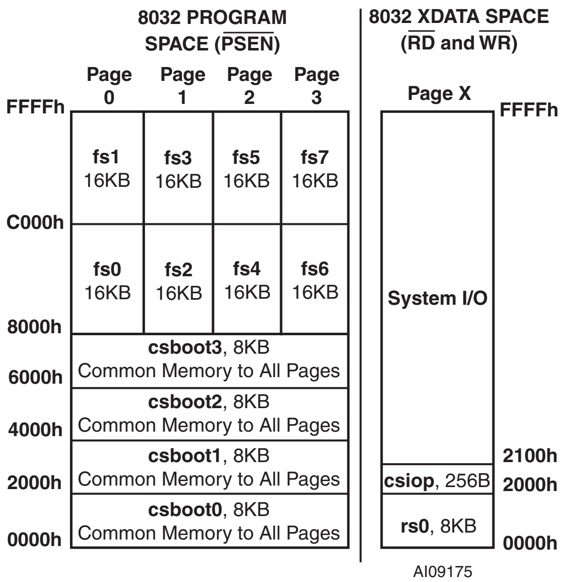
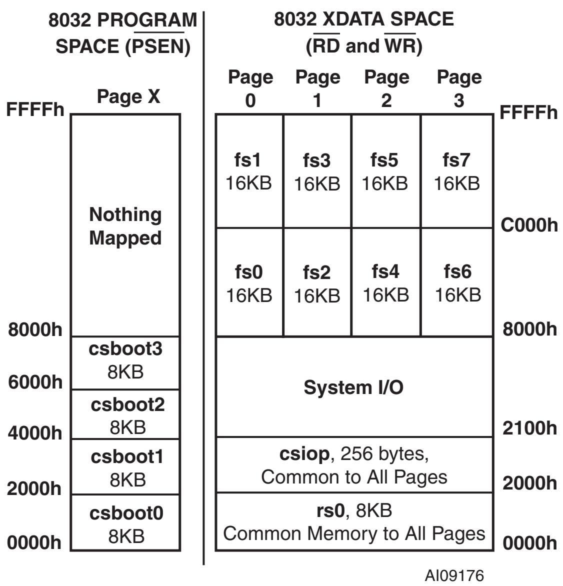


Figure 55. Mapping: all Flash in code space



- *Figure 56 on page 175* Place the larger main Flash memory into XDATA space and the smaller secondary Flash into program space for systems that need a large amount of Flash for data recording or large look-up tables, and not so much Flash for 8032 firmware.

Figure 56. Mapping: small code / big data



It is also possible to “reclassify” the Flash memories during runtime, moving the memories between XDATA memory space and program memory space on-the-fly. This essentially means that the user can override the initial setting during run-time by writing to a csiop register (the VM register). This is useful for IAP, because standard 8051 architecture does not allow writing to program space. For example, if the user wants to update firmware in main Flash memory that is residing in program space, the user can temporarily “reclassify” the main Flash memory into XDATA space to erase and rewrite it while executing IAP code

The Port Data Buffer (PDB) provides feedback to the 8032 and allows only one source at a time to be read when the 8032 reads various csiop registers. There is one PDB for each port pin enabling the 8032 to read the following on a pin-by-pin basis:

1. MCU I/O signal direction setting (csiop Direction reg)
2. Pin drive type setting (csiop Drive Select reg)
3. Latched Addr Out mode setting (csiop Control reg)
4. MCU I/O pin output setting (csiop Data Out reg)
5. Output Enable of pin driver (csiop Enable Out reg)
6. MCU I/O pin input (csiop Data In reg)

A port pin's output enable signal is controlled by a two input OR gate whose inputs come from: a product term of the AND-OR array; the output of the csiop Direction register. If an output enable from the AND-OR Array is not defined, and the port pin is not defined as an OMC output, and if Peripheral I/O mode is not used, then the csiop Direction register has sole control of the OE signal.

As shown in *Figure 68 on page 208*, a physical port pin is connected to the I/O Port logic and is also separately routed to an IMC, allowing the 8032 to read a port pin by two different methods (MCU I/O input mode or read the IMC).

27.4.36 Port operating modes

I/O Port logic has several modes of operation. *Table 125 on page 204* summarizes which modes are available on each port. Each of the port operating modes are described in following sections. Some operating modes can be defined using PSDsoft Express, and some by the 8032 writing to the csiop registers at run-time, and some require both. For example, PLD I/O, Latched Address Out, and Peripheral I/O modes must be defined in PSDsoft Express and programmed into the device using JTAG, but an additional step must happen at run-time to activate Latched Address Out mode and Peripheral I/O mode, but not needed for PLD I/O. In another example, MCU I/O mode is controlled completely by the 8032 at run-time and only a simple pin name declaration is needed in PSDsoft Express for documentation.

Table 131 on page 209 summarizes what actions are needed in PSDsoft Express and what actions are required by the 8032 at run-time to achieve the various port functions.

Table 138. MCU I/O Mode Port C Data Out register (address = csiop + offset 12h)(1)(2)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PC7	N/A	N/A	PC4	PC3	PC2	N/A	N/A

1. For each bit, 1 = drive port pin to logic '1,' 0 = drive port pin to logic '0'
2. Default state of register is 00h after reset or power-up

Table 139. MCU I/O Mode Port D Data Out register (address = csiop + offset 13h)(1)(2)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
N/A	N/A	N/A	N/A	N/A	PD2(3)	PD1	N/A

1. For each bit, 1 = drive port pin to logic '1,' 0 = drive port pin to logic '0'
2. Default state for register is 00h after reset or power-up
3. Not available on 52-pin UPSD33xx devices

Table 140. MCU I/O Mode Port A Direction register (address=csiop+offset 06h)(1)(2)(3)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0

1. Port A not available on 52-pin UPSD33xx devices
2. For each bit, 1 = out from UPSD33xx port pin1, 0 = in to PSD33xx port pin
3. Default state for register is 00h after reset or power-up

Table 141. MCU I/O Mode Port B Direction Inregister (address=csiop+offset 07h)(1)(2)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0

1. For each bit, 1 = out from UPSD33xx port pin1, 0 = in to PSD33xx port pin
2. Default state for register is 00h after reset or power-up

Table 142. MCU I/O Mode Port C Direction register (address = csiop + offset 14h)(1)(2)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PC7	N/A	N/A	PC4	PC3	PC2	N/A	N/A

1. For each bit, 1 = out from UPSD33xx port pin1, 0 = in to PSD33xx port pin
2. Default state for register is 00h after reset or power-up

Table 143. MCU I/O Mode Port DDirection register (address = csiop + offset 15h)(1)(2)

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
N/A	N/A	N/A	N/A	N/A	PD2(3)	PD1	N/A

1. For each bit, 1 = out from UPSD33xx port pin1, 0 = in to PSD33xx port pin
2. Default state for register is 00h after reset or power-up
3. Not available on 52-pin UPSD33xx devices

The Debug signal should always be pulled up externally with a weak pull-up (100K minimum) to V_{CC} even if nothing is connected to it, as shown in *Figure 80* and *Figure 81 on page 238*.

27.5.9 JTAG security setting

A programmable security bit in the PSD module protects its contents from unauthorized viewing and copying. The security bit is set by clicking on the “Additional PSD Settings” box in the main flow diagram of PSDsoft Express, then choosing to set the security bit. Once a file with this setting is programmed into a UPSD33xx using JTAG ISP, any further attempts to communicate with the UPSD33xx using JTAG will be limited. Once secured, the only JTAG operation allowed is a full-chip erase. No reading or modifying Flash memory or PLD logic is allowed. debugging operations to the MCU module are also not allowed. The only way to defeat the security bit is to perform a JTAG ISP full-chip erase operation, after which the device is blank and may be used again. The 8032 on the MCU module will always have access to PSM module memory contents through the 8-bit 8032 data bus connecting the two die, even while the security bit is set.

27.5.10 Initial delivery state

When delivered from STMicroelectronics, UPSD33xx devices are erased, meaning all Flash memory and PLD configuration bits are logic '1.' Firmware and PLD logic configuration must be programmed at least the first time using JTAG ISP. Subsequent programming of Flash memory may be performed using JTAG ISP, JTAG debugging, or the 8032 may run firmware to program Flash memory (IAP).

Table 178. CPLD macrocell asynchronous Clock mode timing (5 V PSD module)

Symbol	Parameter	Conditions	Min	Max	PT Aloc	Turbo Off	Slew rate	Unit
f_{MAXA}	Maximum frequency external feedback	$1/(t_{SA}+t_{COA})$		38.4				MHz
	Maximum frequency internal feedback (f_{CNTA})	$1/(t_{SA}+t_{COA}-10)$		62.5				MHz
	Maximum frequency pipelined data	$1/(t_{CHA}+t_{CLA})$		71.4				MHz
t_{SA}	Input setup time		7		+ 2	+ 10		ns
t_{HA}	Input hold time		8					ns
t_{CHA}	Clock input high time		9			+ 10		ns
t_{CLA}	Clock input low time		9			+ 10		ns
t_{COA}	Clock to output delay			21		+ 10	– 2	ns
t_{ARDA}	CPLD array delay	Any macrocell		11	+ 2			ns
t_{MINA}	Minimum clock period	$1/f_{CNTA}$	16					ns

Table 179. CPLD macrocell asynchronous Clock mode timing (3timeV PSD module)

Symbol	Parameter	Conditions	Min	Max	PT Aloc	Turbo Off	Slew rate	Unit
f_{MAXA}	Maximum frequency external feedback	$1/(t_{SA}+t_{COA})$		21.7				MHz
	Maximum frequency internal feedback (f_{CNTA})	$1/(t_{SA}+t_{COA}-10)$		27.8				MHz
	Maximum frequency pipelined data	$1/(t_{CHA}+t_{CLA})$		33.3				MHz
t_{SA}	Input setup time		10		+ 4	+ 15		ns
t_{HA}	Input hold time		12					ns
t_{CHA}	Clock high time		17			+ 15		ns
t_{CLA}	Clock low time		13			+ 15		ns
t_{COA}	Clock to output delay			31		+ 15	– 6	ns
t_{ARD}	CPLD array delay	Any macrocell		20	+ 4			ns
t_{MINA}	Minimum clock period	$1/f_{CNTA}$	36					ns