# PSD813F1V

## Flash in-system programmable (ISP) peripherals for 8-bit MCUs, 3.3 V

**NOT FOR NEW DESIGN**

## FEATURES SUMMARY

- DUAL BANK FLASH MEMORIES
  - 1 Mbit of Primary Flash Memory (8 Uniform Sectors)
  - 256 Kbit Secondary EEPROM (4 Uniform Sectors)
  - Concurrent operation: read from one memory while erasing and writing the other
- 16 Kbit SRAM
- PLD WITH MACROCELLS
  - Over 3,000 Gates Of PLD: DPLD and CPLD
  - DPLD - User-defined Internal chip-select decoding
  - CPLD with 16 Output Macrocells (OMCs) and 24 Input Macrocells (IMCs)
- 27 RECONFIGURABLE I/Os
  - 27 individually configurable I/O port pins that can be used for the following functions:

    MCU I/Os;

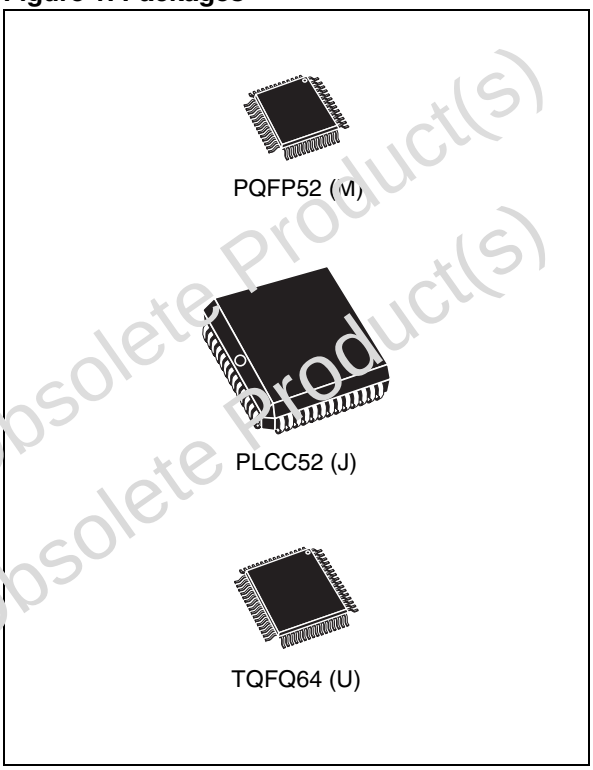
    PLD I/Os;

    Latched MCU address outputs; and

    Special function I/Os.

    **Note:** 16 of the I/O ports may be configured as open drain outputs.
- ENHANCED JTAG SERIAL PORT
  - Built-in JTAG-compliant serial port allows full-chip In-System Programmability (ISP)
  - Efficient manufacturing allows for easy product testing and programming
- PAGE REGISTER
  - Internal page register that can be used to expand the microcontroller address space by a factor of 256.
- PROGRAMMABLE POWER MANAGEMENT

**Figure 1. Packages**




- HIGH ENDURANCE:
  - 100,000 Erase/WRITE Cycles of Flash Memory
  - 10,000 Erase/WRITE Cycles of EEPROM
  - 1,000 Erase/WRITE Cycles of PLD
  - Data Retention: 15-year minimum at 90°C (for Main Flash, Boot, PLD and Configuration bits).
- SINGLE SUPPLY VOLTAGE:
  - 3.3V±10% for PSD813F1V
- STANDBY CURRENT AS LOW AS 50µA
- Packages are ECOPACK®

Obsolete Product(s) - Obsolete Product(s)



This is information on a product still in production but not recommended for new designs.

TABLE OF CONTENTS

features summary . . . . . 1
In-System Programming (ISP) via JTAG . . . . . 6
In-Application Programming (IAP) . . . . . 6
PSDsoft Express . . . . . 6
PIN DESCRIPTION . . . . . 10
PSD ARCHITECTURAL OVERVIEW . . . . . 14
Memory . . . . . 14
PLDs . . . . . 14
I/O Ports . . . . . 14
Microcontroller Bus Interface . . . . . 14
JTAG Port . . . . . 15
In-System Programming (ISP) . . . . . 15
Page Register . . . . . 15
Power Management Unit (PMU) . . . . . 15
DEVELOPMENT SYSTEM . . . . . 16
PSD Register Description and Address Offset . . . . . 17
DETAILED OPERATION . . . . . 18
MEMORY BLOCKS . . . . . 18
Primary Flash Memory and Secondary EEPROM Description . . . . . 18
Memory Operation . . . . . 19
INSTRUCTIONS . . . . . 21
Power-down Instruction and Power-up Mode . . . . . 21
READ . . . . . 22
Data Polling Flag (DQ7) . . . . . 23
Toggle Flag (DQ6) . . . . . 23
Error Flag (DQ5) . . . . . 23
Erase Time-out Flag DQ3 (Flash Memory only) . . . . . 23
Writing to the EEPROM . . . . . 24
Writing the OTP Row . . . . . 25
PROGRAMMING FLASH MEMORY . . . . . 27
Data Polling . . . . . 27
Data Toggle . . . . . 28
ERASING FLASH MEMORY . . . . . 29
Flash Bulk Erase . . . . . 29
Flash Erase Suspend . . . . . 29
Flash Erase Resume . . . . . 29

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

FLASH AND EEPROM MEMORY SPECIFIC FEATURES . . . . . 30
Flash Memory and EEPROM Sector Protect . . . . . 30
Reset . . . . . 30

SRAM . . . . . 30

MEMORY SELECT SIGNALS . . . . . 31
Example . . . . . 31
Memory Select Configuration for MCUs with Separate Program and Data Spaces . . . . . 31
Separate Space Modes . . . . . 32
Combined Space Modes . . . . . 32

PAGE REGISTER . . . . . 33

PLD'S . . . . . 34
The Turbo Bit in PSD . . . . . 34

DECODE PLD (DPLD) . . . . . 36

COMPLEX PLD (CPLD) . . . . . 37
Output Macrocell (OMC) . . . . . 38
Product Term Allocator . . . . . 39
The OMC Mask Register . . . . . 39
The Output Enable of the OMC . . . . . 39
Input Macrocells (IMC) . . . . . 41

MCU BUS INTERFACE . . . . . 44
PSD Interface to a Multiplexed 8-Bit Bus . . . . . 45
PSD Interface to a Non-Multiplexed 8-Bit Bus . . . . . 46
Data Byte Enable Reference . . . . . 47
MCU Bus Interface Examples . . . . . 47
80C31 . . . . . 47
80C251 . . . . . 48
80C51XA . . . . . 50
68HC11 . . . . . 51

I/O PORTS . . . . . 52
General Port Architecture . . . . . 52
Port Operating Modes . . . . . 52
MCU I/O Mode . . . . . 54
PLD I/O Mode . . . . . 54
Address Out Mode . . . . . 54
Address In Mode . . . . . 56
Data Port Mode . . . . . 56
Peripheral I/O Mode . . . . . 56
JTAG In-System Programming (ISP) . . . . . 57

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

Port Configuration Registers (PCR) . . . 57
Control Register . . . 57
Direction Register . . . 58
Drive Select Register . . . 58
Port Data Registers . . . 59
Data In . . . 59
Data Out Register . . . 59
Output Macrocells (OMC) . . . 59
Mask Macrocell Register . . . 59
Input Macrocells (IMC) . . . 59
Enable Out . . . 60
Ports A and B – Functionality and Structure . . . 60
Port C – Functionality and Structure . . . 61
Port D – Functionality and Structure . . . 62
External Chip Select . . . 63

POWER MANAGEMENT . . . 64
Automatic Power-down (APD) Unit and Power-down Mode . . . 65
Power-down Mode . . . 65
For Users of the HC11 (or compatible) . . . 66
Other Power Saving Options . . . 66
PLD Power Management . . . 66
PSD Chip Select Input (CSI, PD2) . . . 66
Input Clock . . . 66
Input Control Signals . . . 68

RESET TIMING AND DEVICE STATUS AT RESET . . . 69
Power-On Reset . . . 69
Warm Reset . . . 69
I/O Pin, Register and PLD Status at Reset . . . 69

PROGRAMMING IN-CIRCUIT USING THE JTAG SERIAL INTERFACE . . . 71
Standard JTAG Signals . . . 71
JTAG Extensions . . . 72
Security, Flash memory and EEPROM Protection . . . 72

INITIAL DELIVERY STATE . . . 72

AC/DC PARAMETERS . . . 73

MAXIMUM RATING . . . 76

DC AND AC PARAMETERS . . . 77

PACKAGE MECHANICAL . . . 100

PART NUMBERING . . . 105

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

APPENDIX A.PQFP52 PIN ASSIGNMENTS . . . . . 106

APPENDIX B.PLCC52 PIN ASSIGNMENTS . . . . . 107

APPENDIX C.TQFP64 PIN ASSIGNMENTS . . . . . 108

REVISION HISTORY . . . . . 109

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

## SUMMARY DESCRIPTION

The PSD family of Programmable Microcontroller (MCU) Peripherals brings In-System Programmability (ISP) to Flash memory and programmable logic. The result is a simple and flexible solution for embedded designs. PSD devices combine many of the peripheral functions found in MCU based applications.

PSD devices integrate an optimized “microcontroller macrocell” logic architecture. The Macrocell was created to address the unique requirements of embedded system designs. It allows direct connection between the system address/data bus and the internal PSD registers to simplify communication between the MCU and other supporting devices.

The PSD family offers two methods to program PSD Flash memory while the PSD is soldered to a circuit board.

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

An IEEE 1149.1 compliant JTAG interface is included on the PSD enabling the entire device (Flash memory, EEPROM, the PLD, and all configuration) to be rapidly programmed while soldered to the circuit board. This requires no MCU participation, which means the PSD can be programmed anytime, even while completely blank.

The innovative JTAG interface to Flash memories is an industry first, solving key problems faced by designers and manufacturing houses, such as:

**First time programming.** How do I get firmware into the Flash the very first time? JTAG is the answer, program the PSD while blank with no MCU involvement.

**Inventory build-up of pre-programmed devices.** How do I maintain an accurate count of pre-programmed Flash memory and PLD devices based on customer demand? How many and what version? JTAG is the answer, build your hardware with blank PSDs soldered directly to the board and then custom program just before they are shipped to customer. No more labels on chips and no more wasted inventory.

**Expensive sockets.** How do I eliminate the need for expensive and unreliable sockets? JTAG is the answer. Solder the PSD directly to the circuit board. Program first time and subsequent times with JTAG. No need to handle devices and bend the fragile leads.

**In-Application Programming (IAP)**

Two independent memory arrays (Flash and EEPROM) are included so the MCU can execute code from one memory while erasing and programming the other. Robust product firmware updates in the field are possible over any communication channel (CAN, Ethernet, UART, J1850, etc.) using this unique architecture. Designers are relieved of these problems:

**Simultaneous read and write to Flash memory.** How can the MCU program the same memory from which it is executing code? It cannot. The PSD allows the MCU to operate the two memories concurrently, reading code from one while erasing and programming the other during IAP.

**Complex memory mapping.** I have only a 64K-byte address space to start with. How can I map these two memories efficiently? A Programmable Decode PLD is the answer. The concurrent PSD memories can be mapped anywhere in MCU address space, segment by segment with extremely high address resolution. As an option, the secondary Flash memory can be swapped out of the system memory map when IAP is complete. A built-in page register breaks the 64K-byte address limit.

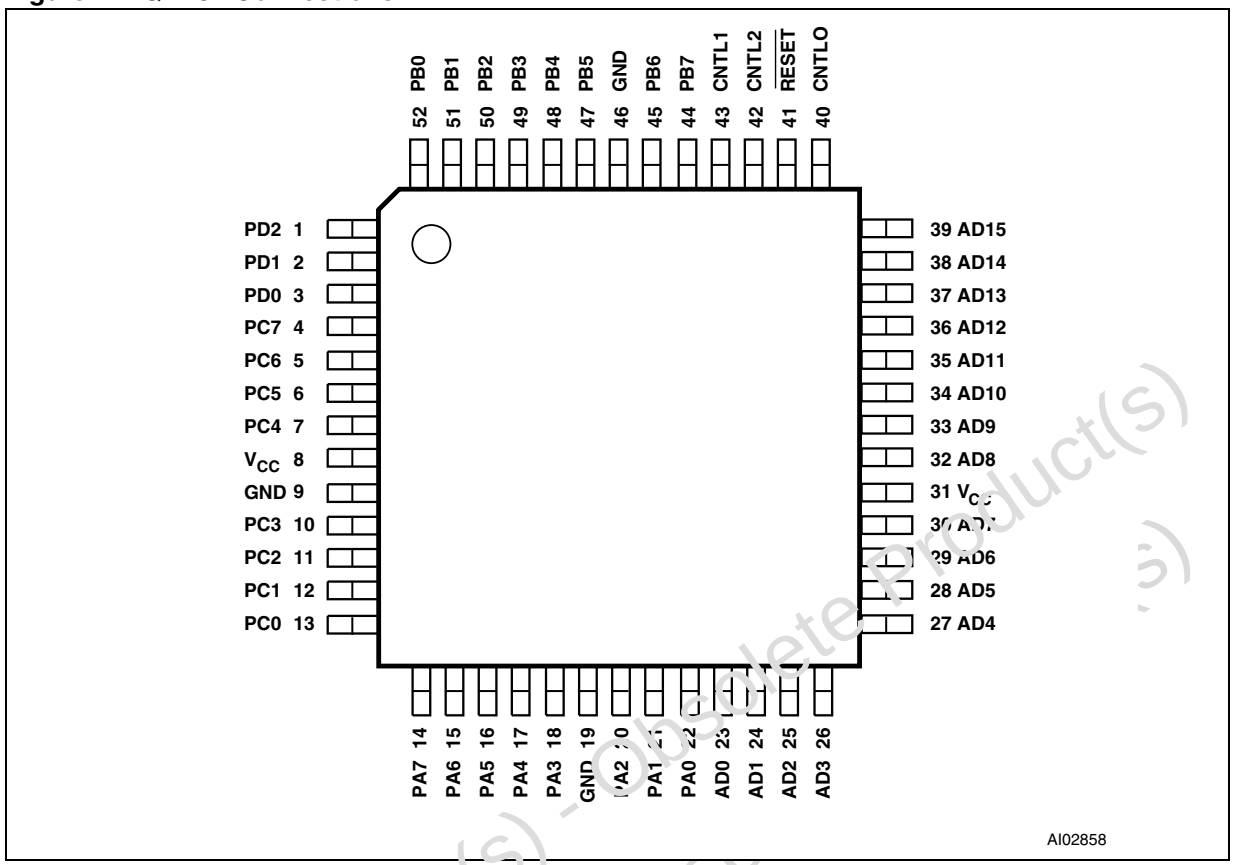
**Separate program and data space.** How can I write to Flash or EEPROM memory while it resides in “program” space during field firmware updates, my MCU won’t allow it! The Flash PSD provides means to “reclassify” Flash or EEPROM memory as “data” space during IAP, then back to “program” space when complete.

**PSDsoft Express**

PSDsoft Express, a software development tool from ST, guides you through the design process step-by-step making it possible to complete an embedded MCU design capable of ISP/IAP in just hours. Select your MCU and PSDsoft Express takes you through the remainder of the design with point and click entry, covering PSD selection, pin definitions, programmable logic inputs and outputs, MCU memory map definition, ANSI-C code generation for your MCU, and merging your MCU firmware with the PSD design. When complete, two different device programmers are supported directly from PSDsoft Express: FlashLINK (JTAG) and PSDpro.

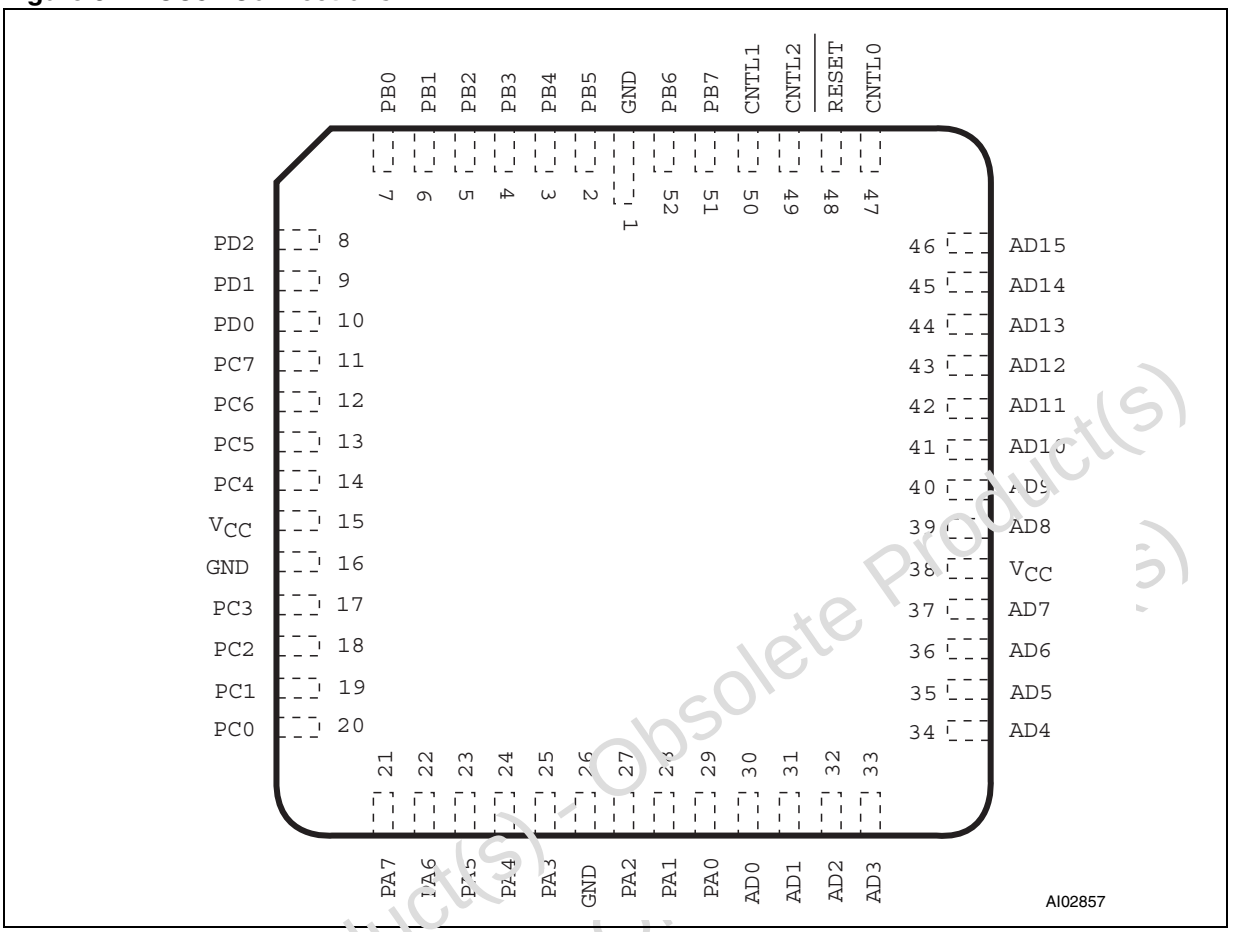
Obsolete Product(s) - Obsolete Product(s)

**Figure 2. PQFP52 Connections**



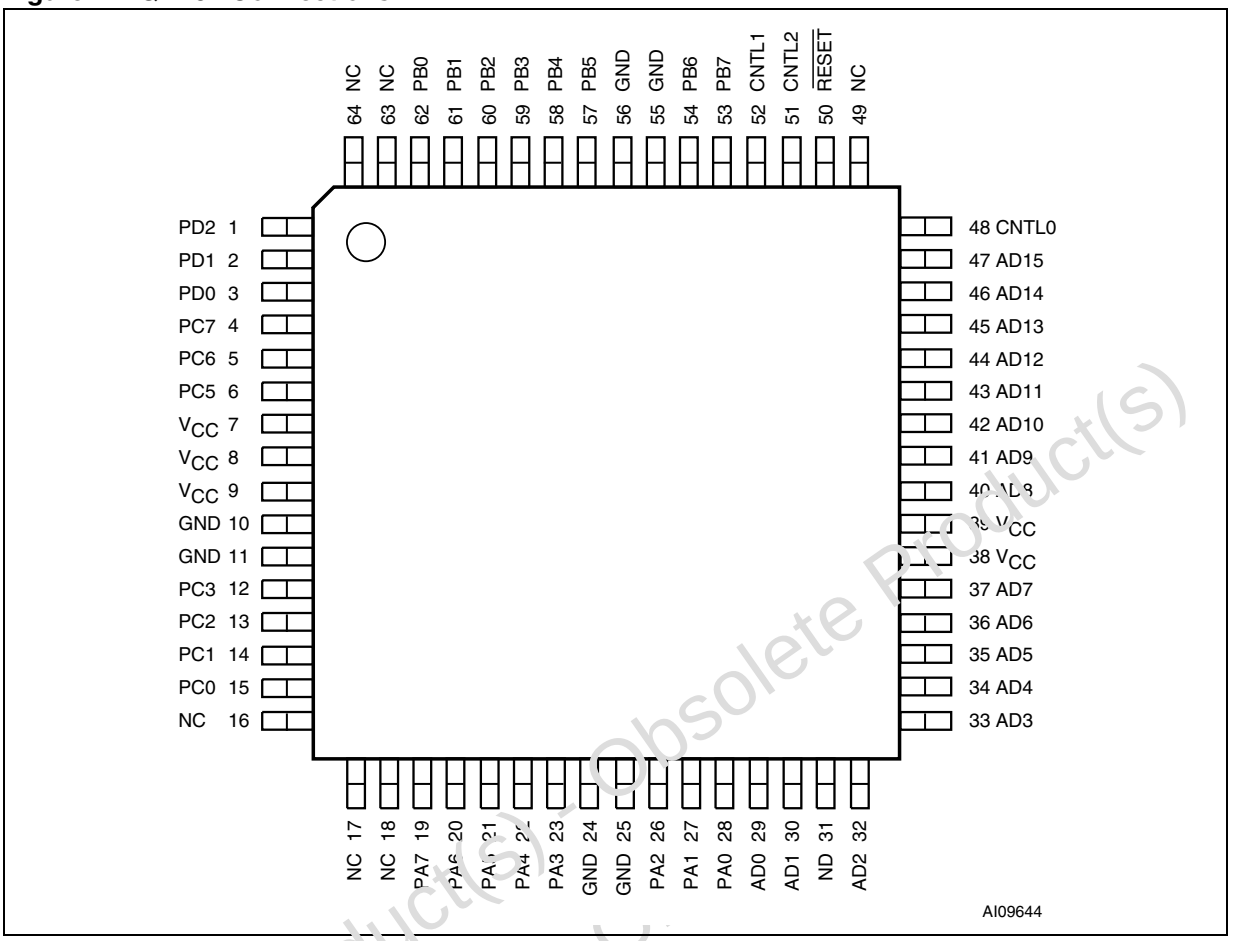

Obsolete Product(s) - Obsolete Product(s)

**Figure 3. PLCC52 Connections**




Obsolete Product(s) - Obsolete Product(s)

**Figure 4. TQFP64 Connections**




Obsolete Product(s) - Obsolete Product(s)

## PIN DESCRIPTION

**Table 1. Pin Description (for the PLCC52 package)**

| Pin Name | Pin | Type | Description(1) |
|---|---|---|---|
| ADIO0-7 | 30-37 | I/O | This is the lower Address/Data port. Connect your MCU address or address/data bus according to the following rules:<br>1. If your MCU has a multiplexed address/data bus where the data is multiplexed with the lower address bits, connect AD0-AD7 to this port.<br>2. If your MCU does not have a multiplexed address/data bus, or you are using an 80C251 in page mode, connect A0-A7 to this port.<br>3. If you are using an 80C51XA in burst mode, connect A4/D0 through A11/D7 to this port.<br>ALE or AS latches the address. The PSD drives data out only if the READ signal is active and one of the PSD functional blocks was selected. The addresses on this port are passed to the PLDs. |
| ADIO8-15 | 39-46 | I/O | This is the upper Address/Data port. Connect your MCU address or address/data bus according to the following rules:<br>1. If your MCU has a multiplexed address/data bus where the data is multiplexed with the lower address bits, connect A8-A15 to this port.<br>2. If your MCU does not have a multiplexed address/data bus, connect A8-A15 to this port.<br>3. If you are using an 80C251 in page mode, connect AD8-AD15 to this port.<br>4. If you are using an 80C51XA in burst mode, connect A12/D8 through A19/D15 to this port.<br>ALE or AS latches the address. The PSD drives data out only if the READ signal is active and one of the PSD functional blocks was selected. The addresses on this port are passed to the PLDs. |
| CNTL0 | 47 | I | The following control signals can be connected to this port, based on your MCU:<br>1. $\overline{WR}$ – active Low Write Strobe input.<br>2. R_$\overline{W}$ – active High READ/active Low write input.<br>This port is connected to the PLDs. Therefore, these signals can be used in decode and other logic equations. |
| CNTL1 | 50 | I | The following control signals can be connected to this port, based on your MCU:<br>1. $\overline{RD}$ – active Low Read Strobe input.<br>2. E – E clock input.<br>3. $\overline{DS}$ – active Low Data Strobe input.<br>4. $\overline{PSEN}$ – connect $\overline{PSEN}$ to this port when it is being used as an active Low $\overline{READ}$ signal. For example, when the 80C251 outputs more than 16 address bits, $\overline{PSEN}$ is actually the READ signal.<br>This port is connected to the PLDs. Therefore, these signals can be used in decode and other logic equations. |
| CNTL2 | 49 | I | This port can be used to input the $\overline{PSEN}$ (Program Select Enable) signal from any MCU that uses this signal for code exclusively. If your MCU does not output a Program Select Enable signal, this port can be used as a generic input. This port is connected to the PLDs. |
| Reset | 48 | I | Active Low Reset input. Resets I/O Ports, PLD macrocells and some of the Configuration Registers. Must be Low at Power-up. |

Obsolete Product(s) - Obsolete Product(s)

| Pin Name | Pin | Type | Description[1] |
|---|---|---|---|
| PA0<br>PA1<br>PA2<br>PA3<br>PA4<br>PA5<br>PA6<br>PA7 | 29<br>28<br>27<br>25<br>24<br>23<br>22<br>21 | I/O | These pins make up Port A. These port pins are configurable and can have the following functions:<br>1. MCU I/O – write to or read from a standard output or input port.<br>2. CPLD macrocell (McellAB0-7) outputs.<br>3. Inputs to the PLDs.<br>4. Latched address outputs (see Table 5).<br>5. Address inputs. For example, PA0-3 could be used for A0-A3 when using an 80C51XA in burst mode.<br>6. As the data bus inputs D0-D7 for non-multiplexed address/data bus MCUs.<br>7. D0/A16-D3/A19 in M37702M2 mode.<br>8. Peripheral I/O mode.<br>Note: PA0-PA3 can only output CMOS signals with an option for high slew rate. However, PA4-PA7 can be configured as CMOS or Open Drain Outputs. |
| PB0<br>PB1<br>PB2<br>PB3<br>PB4<br>PB5<br>PB6<br>PB7 | 7<br>6<br>5<br>4<br>3<br>2<br>52<br>51 | I/O | These pins make up Port B. These port pins are configurable and can have the following functions:<br>1. MCU I/O – write to or read from a standard output or input port.<br>2. CPLD macrocell (McellAB0-7 or McellBC0-7) outputs.<br>3. Inputs to the PLDs.<br>4. Latched address outputs (see Table 5).<br>Note: PB0-PB3 can only output CMOS signals with an option for high slew rate. However, PB4-PB7 can be configured as CMOS or Open Drain Outputs. |
| PC0 | 20 | I/O | PC0 pin of Port C. This port pin can be configured to have the following functions:<br>1. MCU I/O – write to or read from a standard output or input port.<br>2. CPLD macrocell (McellBC0) output.<br>3. Input to the PLDs.<br>4. TMS Input[2] for the JTAG Interface.<br>This pin can be configured as a CMOS or Open Drain output. |
| PC1 | 19 | I/O | PC1 pin of Port C. This port pin can be configured to have the following functions:<br>1. MCU I/O – write to or read from a standard output or input port.<br>2. CPLD macrocell (McellBC1) output.<br>3. Input to the PLDs.<br>4. TCK Input[2] for the JTAG Interface.<br>This pin can be configured as a CMOS or Open Drain output. |
| PC2 | 18 | I/O | PC2 pin of Port C. This port pin can be configured to have the following functions:<br>1. MCU I/O – write to or read from a standard output or input port.<br>2. CPLD macrocell (McellBC2) output.<br>3. Input to the PLDs.<br>This pin can be configured as a CMOS or Open Drain output. |
| PC3 | 17 | I/O | PC3 pin of Port C. This port pin can be configured to have the following functions:<br>1. MCU I/O – write to or read from a standard output or input port.<br>2. CPLD macrocell (McellBC3) output.<br>3. Input to the PLDs.<br>4. $\overline{\text{TSTAT}}$ output[2] for the JTAG Serial Interface.<br>5. Ready/$\overline{\text{Busy}}$ output for In-System parallel programming.<br>This pin can be configured as a CMOS or Open Drain output. |
| PC4 | 14 | I/O | PC4 pin of Port C. This port pin can be configured to have the following functions:<br>1. MCU I/O – write to or read from a standard output or input port.<br>2. CPLD macrocell (McellBC4) output.<br>3. Input to the PLDs.<br>4. $\overline{\text{TERR}}$ output[2] for the JTAG Interface.<br>This pin can be configured as a CMOS or Open Drain output. |

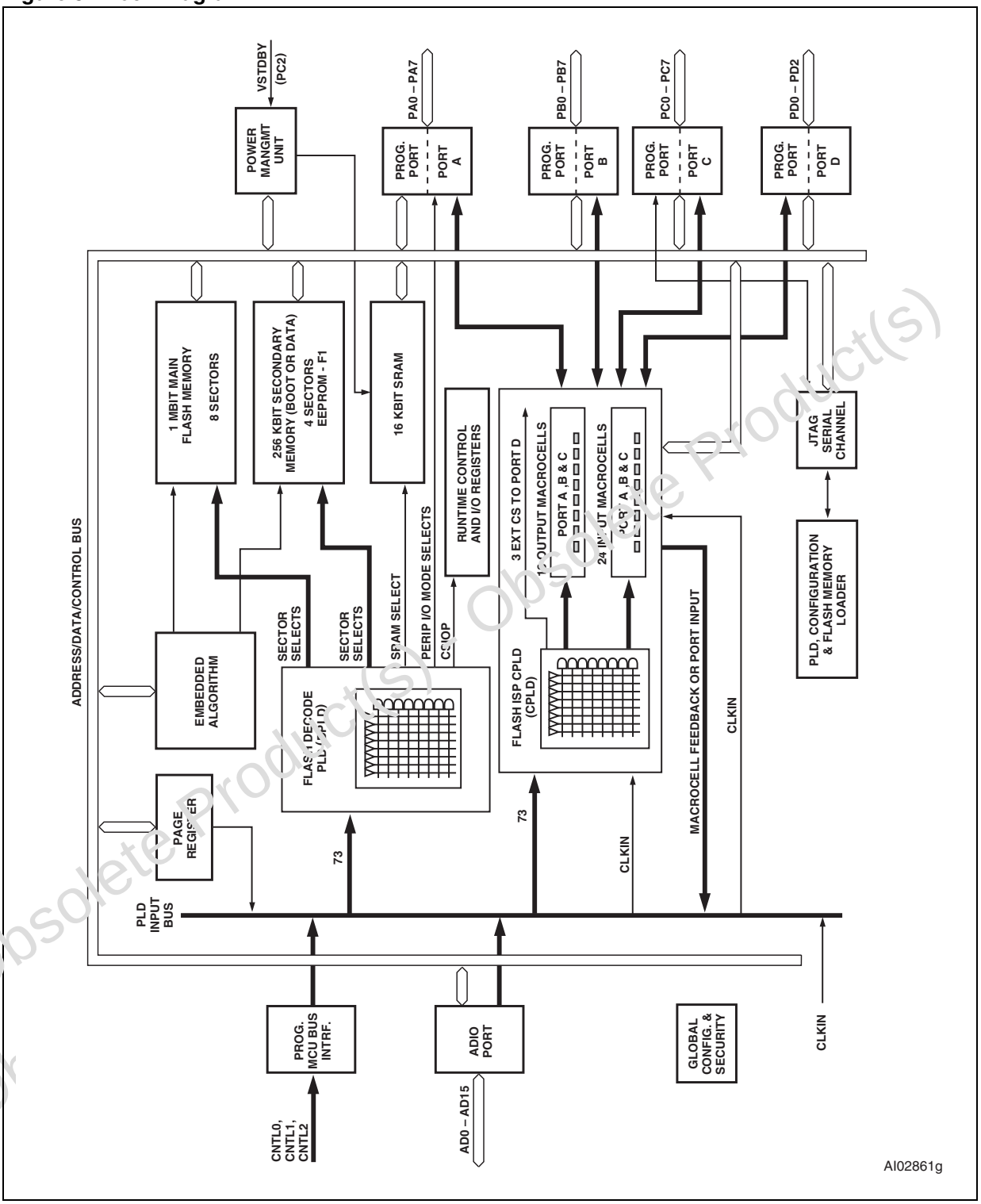
Obsolete Product(s) - Obsolete Product(s)

| Pin Name | Pin | Type | Description(1) |
|---|---|---|---|
| PC5 | 13 | I/O | PC5 pin of Port C. This port pin can be configured to have the following functions:<br>1. MCU I/O – write to or read from a standard output or input port.<br>2. CPLD macrocell (McellBC5) output.<br>3. Input to the PLDs.<br>4. TDI input[2] for the JTAG Interface.<br>This pin can be configured as a CMOS or Open Drain output. |
| PC6 | 12 | I/O | PC6 pin of Port C. This port pin can be configured to have the following functions:<br>1. MCU I/O – write to or read from a standard output or input port.<br>2. CPLD macrocell (McellBC6) output.<br>3. Input to the PLDs.<br>4. TDO output[2] for the JTAG Interface.<br>This pin can be configured as a CMOS or Open Drain output. |
| PC7 | 11 | I/O | PC7 pin of Port C. This port pin can be configured to have the following functions:<br>1. MCU I/O – write to or read from a standard output or input port.<br>2. CPLD macrocell (McellBC7) output.<br>3. Input to the PLDs.<br>4. DBE – active Low Data Byte Enable input from 68HC912 type MCUs.<br>This pin can be configured as a CMOS or Open Drain output. |
| PD0 | 10 | I/O | PD0 pin of Port D. This port pin can be configured to have the following functions:<br>1. ALE/AS input latches address output from the MCU.<br>2. MCU I/O – write or read from a standard output or input port.<br>3. Input to the PLDs.<br>4. CPLD output (External Chip Select). |
| PD1 | 9 | I/O | PD1 pin of Port D. This port pin can be configured to have the following functions:<br>1. MCU I/O – write to or read from a standard output or input port.<br>2. Input to the PLDs.<br>3. CPLD output (External Chip Select).<br>4. CLKIN – clock input to the CPLD macrocells, the APD Unit's Power-down counter, and the CPLD AND Array. |
| PD2 | 8 | I/O | PD2 pin of Port D. This port pin can be configured to have the following functions:<br>1. MCU I/O – write to or read from a standard output or input port.<br>2. Input to the PLDs.<br>3. CPLD output (External Chip Select).<br>4. PSD Chip Select Input ($\overline{CSI}$). When Low, the MCU can access the PSD memory and I/O. When High, the PSD memory blocks are disabled to conserve power. |
| $V_{CC}$ | 15, 38 | | Supply Voltage |
| GND | 1, 16, 26 | | Ground pins |

Note: 1. The pin numbers in this table are for the PLCC package only. See the Figure 2., page 7, for pin numbers on other package type.
2. These functions can be multiplexed with other functions.

Obsolete Product(s) - Obsolete Product(s)

**Figure 5. Block Diagram**




Obsolete Product(s) - Obsolete Product(s)

## PSD ARCHITECTURAL OVERVIEW

PSD devices contain several major functional blocks. Figure 5 shows the architecture of the PSD device. The functions of each block are described briefly in the following sections. Many of the blocks perform multiple functions and are user configurable.

### Memory

The PSD contains the following memories:

- a 1 Mbit Flash memory
- a secondary 256 Kbit EEPROM memory
- a 16 Kbit SRAM

Each of the memory blocks is briefly discussed in the following paragraphs. A more detailed discussion can be found in the section entitled MEMORY BLOCKS, page 18.

The 1 Mbit Flash memory is the main memory of the PSD. It is divided into 8 equally-sized sectors that are individually selectable.

The 256 Kbit EEPROM or Flash memory is divided into 4 equally-sized sectors. Each sector is individually selectable.

The 16 Kbit SRAM is intended for use as a scratchpad memory or as an extension to the microcontroller SRAM.

Each sector of memory can be located in a different address space as defined by the user. The access times for all memory types includes the address latching and DPLD decoding time.

### PLDs

The device contains two PLD blocks, each optimized for a different function, as shown in Table 2. The functional partitioning of the PLDs reduces power consumption, optimizes cost/performance, and eases design entry.

The Decode PLD (DPLD) is used to decode addresses and generate chip selects for the PSD internal memory and registers. The CPLD can implement user-defined logic functions. The DPLD has combinatorial outputs. The CPLD has 16 Output macrocells and 3 combinatorial outputs. The PSD also has 24 Input macrocells that can be configured as inputs to the PLDs. The PLDs receive their inputs from the PLD Input Bus and are differentiated by their output destinations, number of Product Terms, and macrocells.

The PLDs consume minimal power by using Zero-Power design techniques. The speed and power consumption of the PLD is controlled by the Turbo Bit (ZPSD only) in the PMMR0 register and other bits in the PMMR2 registers. These registers are set by the microcontroller at runtime. There is a slight penalty to PLD propagation time when invoking the ZPSD features.

### I/O Ports

The PSD has 27 I/O pins divided among four ports (Port A, B, C, and D). Each I/O pin can be individually configured for different functions. Ports A, B, C and D can be configured as standard MCU I/O ports, PLD I/O, or latched address outputs for microcontrollers using multiplexed address/data busses.

The JTAG pins can be enabled on Port C for In-System Programming (ISP).

Ports A and B can also be configured as a data port for a n on-multiplexed bus or multiplexed Address/Data buses for certain types of 16-bit microcontrollers.

### Microcontroller Bus Interface

The PSD easily interfaces with most 8-bit microcontrollers that have either multiplexed or non-multiplexed address/data busses. The device is configured to respond to the microcontroller's control signals, which are also used as inputs to the PLDs. Where there is a requirement to use a 16-bit data bus to interface to a 16-bit microcontroller, two PSDs must be used. For examples, please see the section entitled MCU Bus Interface Examples, page 47.

**Table 2. PLD I/O**

| Name | Inputs | Outputs | Product Terms |
|---|---|---|---|
| Decode PLD (DPLD) | 73 | 17 | 42 |
| Complex PLD (CPLD) | 73 | 19 | 140 |

Obsolete Product(s) - Obsolete Product(s)

### JTAG Port

In-System Programming can be performed through the JTAG pins on Port C. This serial interface allows complete programming of the entire PSD device. A blank device can be completely programmed. The JTAG signals (TMS, TCK, TSTAT, TERR, TDI, TDO) can be multiplexed with other functions on Port C. Table 3 indicates the JTAG signals pin assignments.

### In-System Programming (ISP)

Using the JTAG signals on Port C, the entire PSD device can be programmed or erased without the use of the microcontroller. The main Flash memory can also be programmed in-system by the microcontroller executing the programming algorithms out of the EEPROM or SRAM. The EEPROM can be programmed the same way by executing out of the main Flash memory. The PLD logic or other PSD configuration can be programmed through the JTAG port or a device programmer. Table 4 indicates which programming methods can program different functional blocks of the PSD.

### Page Register

The 8-bit Page Register expands the address range of the microcontroller by up to 256 times. The paged address can be used as part of the address space to access external memory and peripherals, or internal memory and I/O. The Page Register can also be used to change the address mapping of blocks of Flash memory into different memory spaces for in-circuit programming.

### Power Management Unit (PMU)

The Power Management Unit (PMU) in the PSD gives the user control of the power consumption on selected functional blocks based on system requirements. The PMU includes an Automatic Power Down unit (APD) that will turn off device functions due to microcontroller inactivity. The APD unit has a Power Down Mode that helps reduce power consumption.

The PSD also has some bits that are configured at run-time by the MCU to reduce power consumption of the CPLD. The turbo bit in the PMMR0 register can be turned off and the CPLD will latch its outputs and go to sleep until the next transition on its inputs.

Additionally, bits in the PMMR2 register can be set by the MCU to block signals from entering the CPLD to reduce power consumption. Please see the section entitled POWER MANAGEMENT, page 34 for more details.

**Table 3. JTAG Signals on Port C**

| Port C Pins | JTAG Signal |
|---|---|
| PC0 | TMS |
| PC1 | TCK |
| PC3 | $\overline{\text{TSTAT}}$ |
| PC4 | $\overline{\text{TERR}}$ |
| PC5 | TDI |
| PC6 | TDO |

**Table 4. Methods of Programming Different Functional Blocks of the PSD**

| Functional Block | JTAG Programming | Device Programmer | In-System Parallel Programming |
|---|---|---|---|
| Main Flash Memory | Yes | Yes | Yes |
| EEPROM Memory | Yes | Yes | Yes |
| PLD Array (DPLD and CPLD) | Yes | Yes | No |
| PSD Configuration | Yes | Yes | No |
| Optional OTP Row | No | Yes | Yes |

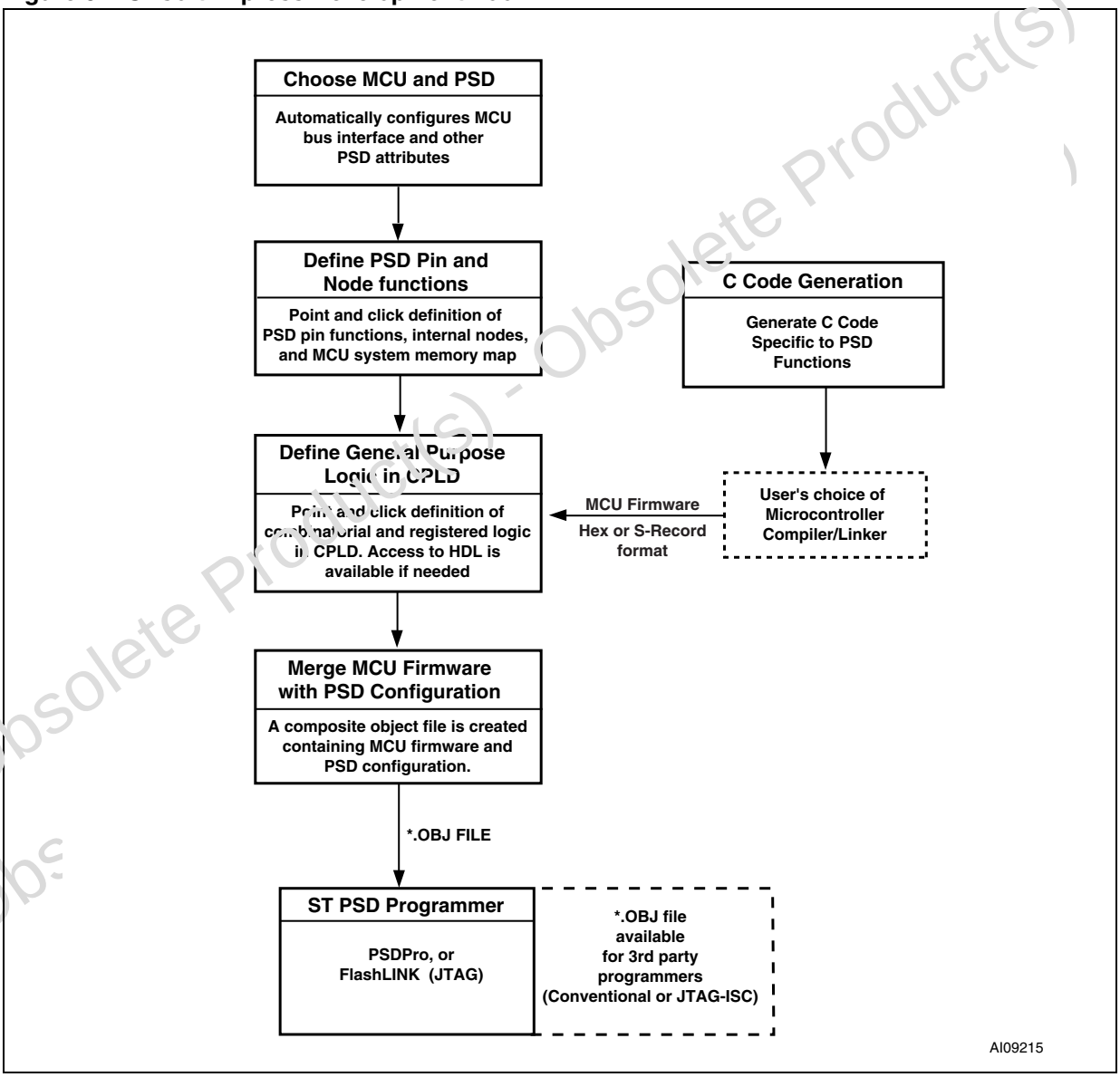
Obsolete Product(s) - Obsolete Product(s)

## DEVELOPMENT SYSTEM

The PSD is supported by PSDsoft Express a Windows-based (95, 98, NT) software development tool. A PSD design is quickly and easily produced in a point and click environment. The designer does not need to enter Hardware Definition Language (HDL) equations (unless desired) to define PSD pin functions and memory map information. The general design flow is shown in Figure 6 below. PSDsoft Express is available from our web site (www.st.com/psm) or other distribution channels.

PSDsoft Express directly supports two low cost device programmers from ST, PSDpro and FlashLINK (JTAG). Both of these programmers may be purchased through your local distributor/representative, or directly from our web site using a credit card. The PSD is also supported by third party device programmers, see web site for current list.

**Figure 6. PSDsoft Express Development Tool**




Obsolete Product(s) - Obsolete Product(s)

## PSD REGISTER DESCRIPTION AND ADDRESS OFFSET

Table 5 shows the offset addresses to the PSD 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 registers.

Table 6 provides brief descriptions of the registers in CSIOP space. The following section gives a more detailed description.

**Table 5. I/O Port Latched Address Output Assignments**

| MCU(1) | Port A(2) | | Port B(2) | |
|---|---|---|---|---|
| | Port A (3:0) | Port A (7:4) | Port B (3:0) | Port B (7:4) |
| 8051XA (8-bit) | N/A | Address a7-a4 | Address a11-a8 | N/A |
| 80C251 (page mode) | N/A | N/A | Address a11-a8 | Address a15-a12 |
| All other 8-bit multiplexed | Address a3-a0 | Address a7-a4 | Address a3-a0 | Address a7-a4 |
| 8-bit non-multiplexed bus | N/A | N/A | Address a3-a0 | Address a7-a4 |

Note: 1. See the section entitled I/O PORTS, page 52, on how to enable the Latched Address Output function.
2. N/A = Not Applicable

**Table 6. Register Address Offset**

| Register Name | Port A | Port B | Port C | Port D | Other(1) | 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<br>WRITE – loads macrocell flip-flops |
| Output Macrocells BC | | 21 | 21 | | | READ – reads output of macrocells BC<br>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 – Flash Sector Protection |
| Secondary Flash memory Protection | | | | | C2 | Read only – PSD Security and EEPROM Sector Protection |
| JTAG Enable | | | | | C7 | Enables JTAG Port |
| PMMR0 | | | | | B0 | Power Management Register 0 |
| PMMR2 | | | | | B4 | Power Management Register 2 |
| Page | | | | | E0 | Page Register |
| VM | | | | | E2 | Places PSD 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.

Obsolete Product(s) - Obsolete Product(s)

## DETAILED OPERATION

As shown in Figure 5., page 13, the PSD consists of six major types of functional blocks:

- Memory Blocks
- PLD Blocks
- MCU Bus Interface
- 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 has the following memory blocks (see Table 7):

– The Main Flash memory
– Secondary EEPROM 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 EEPROM Description

The 1Mb primary Flash memory is divided evenly into eight 16-KByte sectors. The EEPROM memory is divided into four sectors of eight KBytes each. Each sector of either memory can be separately protected from Program and Erase operations.

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

EEPROM may be programmed byte-by-byte or sector-by-sector, and erasing is automatic and transparent. The integrity of the data can be secured with the help of Software Data Protection (SDP). Any write operation to the EEPROM is inhibited during the first five milliseconds following power-up.

During a program or erase of Flash, or during a write of the EEPROM, the status can be output on the Ready/Busy (PC3) pin of Port C3. This pin is set up using PSDsoft Express Configuration.

**Memory Block Select Signals.** The decode PLD in the PSD generates the chip selects for all the internal memory blocks (refer to the section entitled PLD'S, page 34). Each of the eight Flash memory sectors have a Flash Select signal (FS0-FS7) which can contain up to three product terms. Each of the four EEPROM memory sectors have a Select signal (EES0-3 or CSBOOT0-3) which can contain up to three product terms. Having three product terms for each sector select signal allows a given sector to be mapped in different areas of system memory. When using a microcontroller with separate Program and Data space, these flexible select signals allow dynamic re-mapping of sectors from one space to the other.

**Ready/Busy Pin (PC3).** Pin PC3 can be used to output the Ready/Busy status of the PSD. The output on the pin will be a '0' (Busy) when Flash or EEPROM memory blocks are being written to, or when the Flash memory block is being erased. The output will be a '1' (Ready) when no write or erase operation is in progress.

**Table 7. Memory Blocks**

| Device | Main Flash | EEPROM | SRAM |
|---|---|---|---|
| PSD813F1 | 128KB | 32KB | 2KB |

Obsolete Product(s) - Obsolete Product(s)

### Memory Operation

The main Flash and EEPROM memory are addressed through the microcontroller interface on the PSD device. The microcontroller can access these memories in one of two ways:

– The microcontroller can execute a typical bus WRITE or READ operation just as it would if accessing a RAM or ROM device using standard bus cycles.
– The microcontroller can execute a specific instruction that consists of several WRITE and READ operations. This involves writing specific data patterns to special addresses within the Flash or EEPROM to invoke an embedded algorithm. These instructions are summarized in Table 8., page 20.

Typically, Flash memory can be read by the microcontroller using READ operations, just as it would read a ROM device. However, Flash memory can only be erased and programmed with specific instructions. For example, the microcontroller cannot write a single byte directly to Flash memory as one would write a byte to RAM. To program a byte into Flash memory, the microcontroller must execute a program instruction sequence, then test the status of the programming event. This status test is achieved by a READ operation or polling the Ready/Busy pin (PC3).

The Flash memory can also be read by using special instructions to retrieve particular Flash device information (sector protect status and ID).

The EEPROM is a bit different. Data can be written to EEPROM memory using write operations, like writing to a RAM device, but the status of each WRITE event must be checked by the microcontroller. A WRITE event can be one to 64 contiguous bytes. The status test is very similar to that used for Flash memory (READ operation or Ready/Busy). Optionally, the EEPROM memory may be put into a Software Data Protect (SDP) mode where it requires instructions, rather than operations, to alter its contents. SDP mode makes writing to EEPROM much like writing to Flash memory.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Table 8. Instructions**

| Instruction | EEPROM Sector Select (EESi) | Flash Sector Select (FSi)(2) | Cycle 1 | Cycle 2 | Cycle 3 | Cycle 4 | Cycle 5 | Cycle 6 | Cycle 7 |
|---|---|---|---|---|---|---|---|---|---|
| Read Flash Identifier[3,5] | 0 | 1 | AAh@ X555h | 55h@ XAAAh | 90h@ X555h | Read Identifier with (A6,A1,A0 at 0,0,1) | | | |
| Read OTP row[4] | 1 | 0 | AAh@ X555h | 55h@ XAAAh | 90h@ X555h | Read byte 1 | Read byte 2 | | Read byte N |
| Read Sector Protection Status[3,5] | 0 | 1 | AAh@ X555h | 55h@ XAAAh | 90h@ X555h | Read identifier with (A6, A1; A0 = 0,1,0) | | | |
| Program a Flash Byte[5] | 0 | 1 | AAh@ X555h | 55h@ XAAAh | A0h@ X555h | Data@ address | | | |
| Erase one Flash Sector[5] | 0 | 1 | AAh@ X555h | 55h@ XAAAh | 80h@ X555h | AAh@ X555h | 55h@ XAAAh | 30h@ Sector address | 30h@ Sector address[1] |
| Erase the Whole Flash[5] | 0 | 1 | AAh@ X555h | 55h@ XAAAh | 80h@ X555h | AAh@ X555h | 55h@ XAAAh | 10h@ X555h | |
| Suspend Sector Erase[5] | 0 | 1 | B0h@ XXXXh | | | | | | |
| Resume Sector Erase[5] | 0 | 1 | 30h@ XXXXh | | | | | | |
| EEPROM Power Down[4] | 1 | 0 | AAh@ X555h | 55h@ XAAAh | 30h@ X555h | | | | |
| SDP Enable/ EEPROM Write[4] | 1 | 0 | AAh@ X555h | 55h@ XAAAh | A0h@ X555h | Write byte 1 | Write byte 2 | | Write byte N |
| SDP Disable[4] | 1 | 0 | AAh@ X555h | 55h@ XAAAh | 80h@ X555h | AAh@ X555h | 55h@ XAAAh | 20h@ X555h | |
| Write in OTP Row[4,6] | 1 | 0 | AAh@ X555h | 55h@ XAAAh | B0h@ X555h | Write byte 1 | Write byte 2 | | Write byte N |
| Return (from OTP Read or EEPROM Power-Down)[4] | 1 | 0 | F0h@ XXXX | | | | | | |
| Reset[3,5] | 0 | 1 | AAh@ X555h | 55h@ XAAAh | F0h@ XXXX | | | | |
| Reset (short instruction)[5] | 0 | 1 | F0h@ XXXX | | | | | | |

Note: 1. Additional sectors to be erased must be entered within 80 µs. A Sector Address is any address within the Sector.
2. Flash and EEPROM Sector Selects are active high. Addresses A15-A12 are don't cares in Instruction Bus Cycles.
3. The Reset instruction is required to return to the normal READ mode if DQ5 goes high or after reading the Flash Identifier or Protection status.
4. The MCU cannot invoke these instructions while executing code from EEPROM. The MCU must be operating from some other memory when these instructions are performed.
5. The MCU cannot invoke these instructions while executing code from the same Flash memory for which the instruction is intended. The MCU must operate from some other memory when these instructions are executed.
6. Writing to OTP Row is allowed only when SDP mode is disabled.

Obsolete Product(s) Obsolete Product(s) Obsolete Product(s) Obsolete Product(s)

## INSTRUCTIONS

An instruction is defined as a sequence of specific operations. Each received byte is sequentially decoded by the PSD and not executed as a standard write operation. The instruction is executed when the correct number of bytes are properly received and the time between two consecutive bytes is shorter than the time-out value. Some instructions are structured to include READ operations after the initial WRITE operations.

The sequencing of any instruction must be followed exactly. Any invalid combination of instruction bytes or time-out between two consecutive bytes while addressing Flash memory will reset the device logic into READ mode (Flash memory reads like a ROM device). An invalid combination or time-out while addressing the EEPROM block will cause the offending byte to be interpreted as a single operation.

The PSD supports these instructions (see Table 8., page 20):

Flash memory:

- Erase memory by chip or sector
- Suspend or resume sector erase
- Program a Byte
- Reset to READ mode
- Read Flash Identifier value
- Read Sector Protection Status

EEPROM:

- Write data to OTP Row
- Read data from OTP Row
- Power down memory
- Enable Software Data Protect (SDP)
- Disable SDP
- Return from read OTP Row read mode or power down mode.

These instructions are detailed in Table 8., page 20. For efficient decoding of the instructions, the first two bytes of an instruction are the coded cycles and are followed by a command 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 lines A15-A12 are don't cares during the instruction WRITE cycles. However, the appropriate sector select signal (FSi or EESi) must be selected.

### Power-down Instruction and Power-up Mode

**EEPROM Power Down Instruction.** The EEPROM can enter power down mode with the help of the EEPROM power down instruction (see Table 8., page 20). Once the EEPROM power down instruction is decoded, the EEPROM memory cannot be accessed unless a Return instruction (also in Table 8., page 20) is decoded. Alternately, this power down mode will automatically occur when the APD circuit is triggered (see section entitled Automatic Power-down (APD) Unit and Power-down Mode, page 65). Therefore, this instruction is not required if the APD circuit is used.

**Power-up Mode.** The PSD internal logic is reset upon power-up to the READ mode. Any write operation to the EEPROM is inhibited during the first 5ms following power-up. The FSi and EESi select signals, along with the write strobe signal, must be in the false state during power-up for maximum security of the data contents and to remove the possibility of a byte being written on the first edge of a write strobe signal. Any write cycle initiation is locked when $V_{CC}$ is below $V_{LKO}$.

Obsolete Product(s) - Obsolete Product(s)

### READ

Under typical conditions, the microcontroller may read the Flash or EEPROM memory using READ operations just as it would a ROM or RAM device. Alternately, the microcontroller may use READ operations to obtain status information about a Program or Erase operation in progress. Lastly, the microcontroller may use instructions to read special data from these memories. The following sections describe these READ functions.

**Read Memory Contents.** Main Flash is placed in the READ mode after power-up, chip reset, or a Reset Flash instruction (see Table 8., page 20). The microcontroller can read the memory contents of main Flash or EEPROM by using READ operations any time the READ operation is not part of an instruction sequence.

**Read Main Flash Memory Identifier.** The main Flash memory identifier is read with an instruction composed of 4 operations:

3 specific write operations and a READ operation (see Table 8). During the READ operation, address bits A6, A1, and A0 must be 0,0,1, respectively, and the appropriate sector select signal (FSi) must be active. The Flash ID is E3h for the PSD. The MCU can read the ID only when it is executing from the EEPROM.

**Read Main Flash Memory Sector Protection Status.** The main Flash memory sector protection status is read with an instruction composed of 4 operations: 3 specific WRITE operations and a READ operation (see Table 8., page 20). During the READ operation, address bits A6, A1, and A0 must be 0,1,0, respectively, while the chip select FSi designates the Flash sector whose protection has to be verified. The READ operation will produce 01h if the Flash sector is protected, or 00h if the sector is not protected.

The sector protection status for all NVM blocks (main Flash or EEPROM) can be read by the microcontroller accessing the Flash Protection and PSD/EE Protection registers in PSD I/O space. See Flash Memory and EEPROM Sector Protect, page 30 for register definitions.

**Reading the OTP Row.** There are 64 bytes of One-Time-Programmable (OTP) memory that reside in EEPROM. These 64 bytes are in addition to the 32 Kbytes of EEPROM memory. A READ of the OTP row is done with an instruction composed of at least 4 operations: 3 specific WRITE operations and one to 64 READ operations (see Table 8., page 20). During the READ operation(s), address bit A6 must be zero, while address bits A5-A0 define the OTP Row byte to be read while any EEPROM sector select signal (EESi) is active. After reading the last byte, an EEPROM Return instruction must be executed (see Table 8., page 20).

**Reading the Erase/Program Status Bits.** The PSD provides several status bits to be used by the microcontroller to confirm the completion of an erase or programming instruction of Flash memory. Bits are also available to show the status of WRITES to EEPROM. These status bits minimize the time that the microcontroller spends performing these tasks and are defined in Table 9. The status bits can be read as many times as needed.

For Flash memory, the microcontroller 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 27 for details.

For EEPROM not in SDP mode, the microcontroller can perform a READ operation to obtain these status bits just after a data WRITE operation. The microcontroller may write one to 64 bytes before reading the status bits. See the section entitled Writing to the EEPROM, page 24 for details.

For EEPROM in SDP mode, the microcontroller will perform a READ operation to obtain these status bits while an SDP write instruction is being executed by the embedded algorithm. See section entitled EEPROM Software Data Protect (SDP), page 24 for details.

Obsolete Product(s) - Obsolete Product(s)

**Table 9. Status Bit**

| Device | FSi/ CSBOOTi | EESi | DQ7 | DQ6 | DQ5 | DQ4 | DQ3 | DQ2 | DQ1 | DQ0 |
|---|---|---|---|---|---|---|---|---|---|---|
| Flash | $V_{IH}$ | $V_{IL}$ | Data Polling | Toggle Flag | Error Flag | X | Erase Timeout | X | X | X |
| EEPROM | $V_{IL}$ | $V_{IH}$ | Data Polling | Toggle Flag | X | X | X | 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. FSi and EESi are active High.

**Data Polling Flag (DQ7)**

When Erasing or Programming the Flash memory (or when Writing into the EEPROM memory), bit DQ7 outputs the complement of the bit being entered for Programming/Writing on DQ7. Once the Program instruction or the WRITE operation is completed, the true logic value is read on DQ7 (in a Read operation). Flash memory specific features:

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

**Toggle Flag (DQ6)**

The PSD offers another way for determining when the EEPROM write or the Flash memory Program instruction is completed. During the internal WRITE operation and when either the FSi or EESi is true, the DQ6 will toggle 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 will stop and the data read on the Data Bus D0-7 is the addressed memory byte. The device is now accessible for a new READ or WRITE operation.

The operation is finished when two successive reads yield the same output data. Flash memory specific features:

- The Toggle bit is effective after the fourth WRITE pulse (for programming) or after the sixth WRITE pulse (for Erase).
- If the byte to be programmed belongs to a protected Flash sector, the instruction is ignored.
- If all the Flash sectors selected for erasure are protected, DQ6 will toggle to '0' for about 100 µs and then return to the previous addressed byte.

**Error Flag (DQ5)**

During a correct Program or Erase, the Error bit will set to '0.' This bit is set to '1' when there is a failure during Flash byte programming, Sector erase, or Bulk Erase.

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

In case of an error in Flash sector erase or byte program, the Flash sector in which the error occurred or to which the programmed byte belongs must no longer be used. Other Flash sectors may still be used. The Error bit resets after the Reset instruction.

**Erase Time-out Flag DQ3 (Flash Memory only)**

The Erase Timer bit reflects the time-out period allowed between two consecutive Sector Erase instructions. The Erase timer bit is set to '0' after a Sector Erase instruction 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, DQ3 is set to '1.'

Obsolete Product(s) Obsolete Product(s)

### Writing to the EEPROM

Data may be written a byte at a time to the EEPROM using simple write operations, much like writing to an SRAM. Unlike SRAM though, the completion of each byte write must be checked before the next byte is written. To speed up this process, the PSD offers a Page write feature to allow writing of several bytes before checking status.

To prevent inadvertent writes to EEPROM, the PSD offers a Software Data Protect (SDP) mode. Once enabled, SDP forces the MCU to "unlock" the EEPROM before altering its contents, much like Flash memory programming.

**Writing a Byte to EEPROM.** A write operation is initiated when an EEPROM select signal (EESi) is true and the write strobe signal ($\overline{WR}$) into the PSD is true. If the PSD detects no additional writes within 120µsec, an internal storage operation is initiated. Internal storage to EEPROM memory technology typically takes a few milliseconds to complete.

The status of the write operation is obtained by the MCU reading the Data Polling or Toggle bits (as detailed in section entitled READ, page 22), or the Ready/$\overline{Busy}$ output pin (section Ready/Busy Pin (PC3), page 18).

Keep in mind that the MCU does not need to erase a location in EEPROM before writing it. Erasure is performed automatically as an internal process.

**Writing a Page to EEPROM.** Writing data to EEPROM using page mode is more efficient than writing one byte at a time. The PSD EEPROM has a 64 byte volatile buffer that the MCU may fill before an internal EEPROM storage operation is initiated. Page mode timing approaches a 64:1 advantage over the time it takes to write individual bytes.

To invoke page mode, the MCU must write to EEPROM locations within a single page, with no more than 120µs between individual byte writes. A single page means that address lines A14 to A6 must remain constant. The MCU may write to the 64 locations on a page in any order, which is determined by address lines A5 to A0. As soon as 120µs have expired after the last page write, the internal EEPROM storage process begins and the MCU checks programming status. Status is checked the same way it is for byte writes, described above.

**Note:** Be aware that if the upper address bits (A14 to A6) change during page write operations, loss of data may occur. Ensure that all bytes for a given page have been successfully stored in the EEPROM before proceeding to the next page. Correct management of MCU interrupts during EEPROM page write operations is essential.

**EEPROM Software Data Protect (SDP).** The SDP feature is useful for protecting the contents of EEPROM from inadvertent write cycles that may occur during uncontrolled MCU bus conditions. These may happen if the application software gets lost or when VCC is not within normal operating range.

Instructions from the MCU are used to enable and disable SDP mode (see Table 8., page 20). Once enabled, the MCU must write an instruction sequence to EEPROM before writing data (much like writing to Flash memory). SDP mode can be used for both byte and page writes to EEPROM. The device will remain in SDP mode until the MCU issues a valid SDP disable instruction.

PSD devices are shipped with SDP mode disabled. However, within PSDsoft Express, SDP mode may be enabled as part of programming the device with a device programmer (PSDpro).

To enable SDP mode at run time, the MCU must write three specific data bytes at three specific memory locations, as shown in Figure 7., page 25. Any further writes to EEPROM when SDP is set will require this same sequence, followed by the byte(s) to write. The first SDP enable sequence can be followed directly by the byte(s) to be written.

To disable SDP mode, the MCU must write specific bytes to six specific locations, as shown in Figure 8., page 26.

The MCU must not be executing code from EEPROM when these instructions are invoked. The MCU must be operating from some other memory when enabling or disabling SDP mode.

The state of SDP mode is not changed by power on/off sequences (nonvolatile). When either the SDP enable or SDP disable instructions are issued from the MCU, the MCU must use the Toggle bit (status bit DQ6) or the Ready/$\overline{Busy}$ output pin to check programming status. The Ready/$\overline{Busy}$ output is driven low from the first write of AAh @ 555h until the completion of the internal storage sequence. Data Polling (status bit DQ7) is not supported when issuing the SDP enable or SDP disable commands.

**Note:** Using the SDP sequence (enabling, disabling, or writing data) is initiated when specific bytes are written to addresses on specific "pages" of EEPROM memory, with no more than 120µs between WRITES. The addresses 555h and AAAh are located on different pages of EEPROM. This is how the PSD distinguishes these instruction sequences from ordinary writes to EEPROM, which are expected to be within a single EEPROM page.

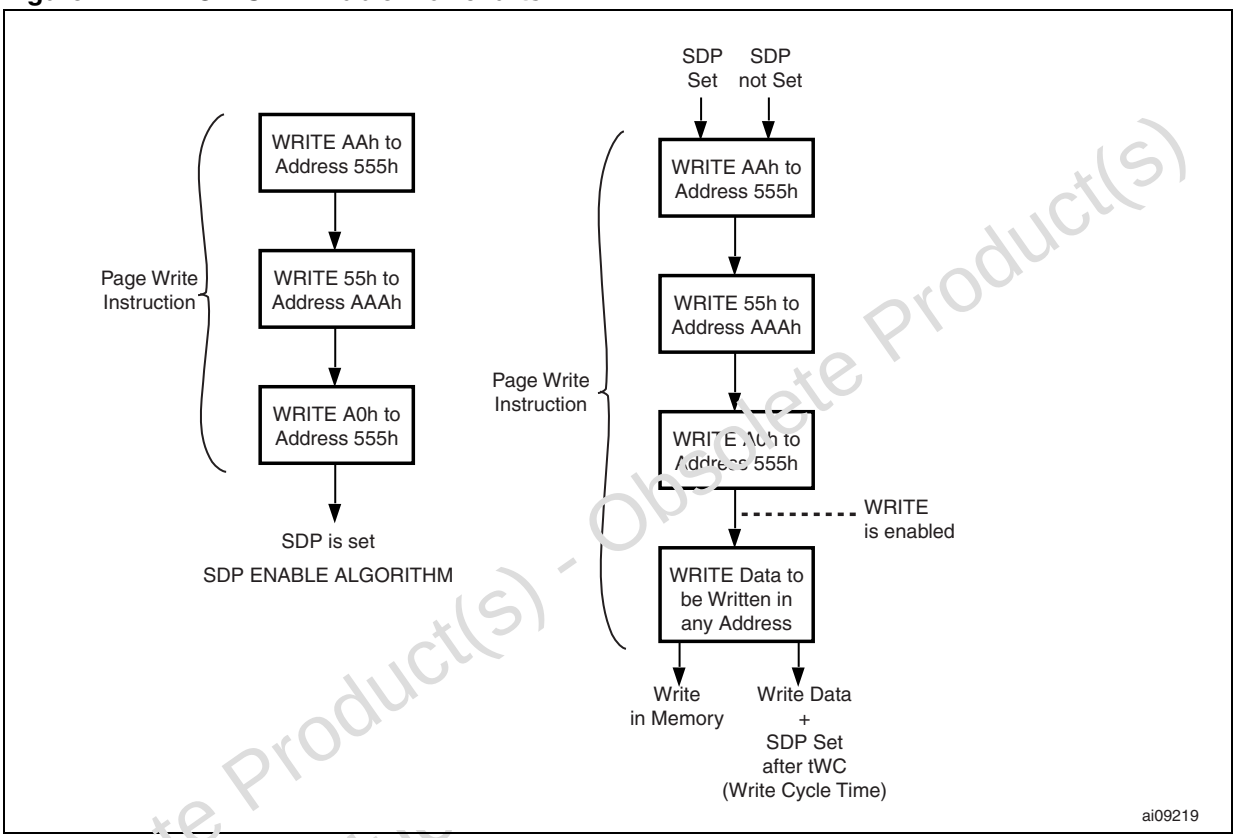
Obsolete Product(s) - Obsolete Product(s)

### Writing the OTP Row

Writing to the OTP row (64 bytes) can only be done once per byte, and is enabled by an instruction. This instruction is composed of three specific WRITE operations of data bytes at three specific memory locations followed by the data to be stored in the OTP row (refer to Table 8., page 20).

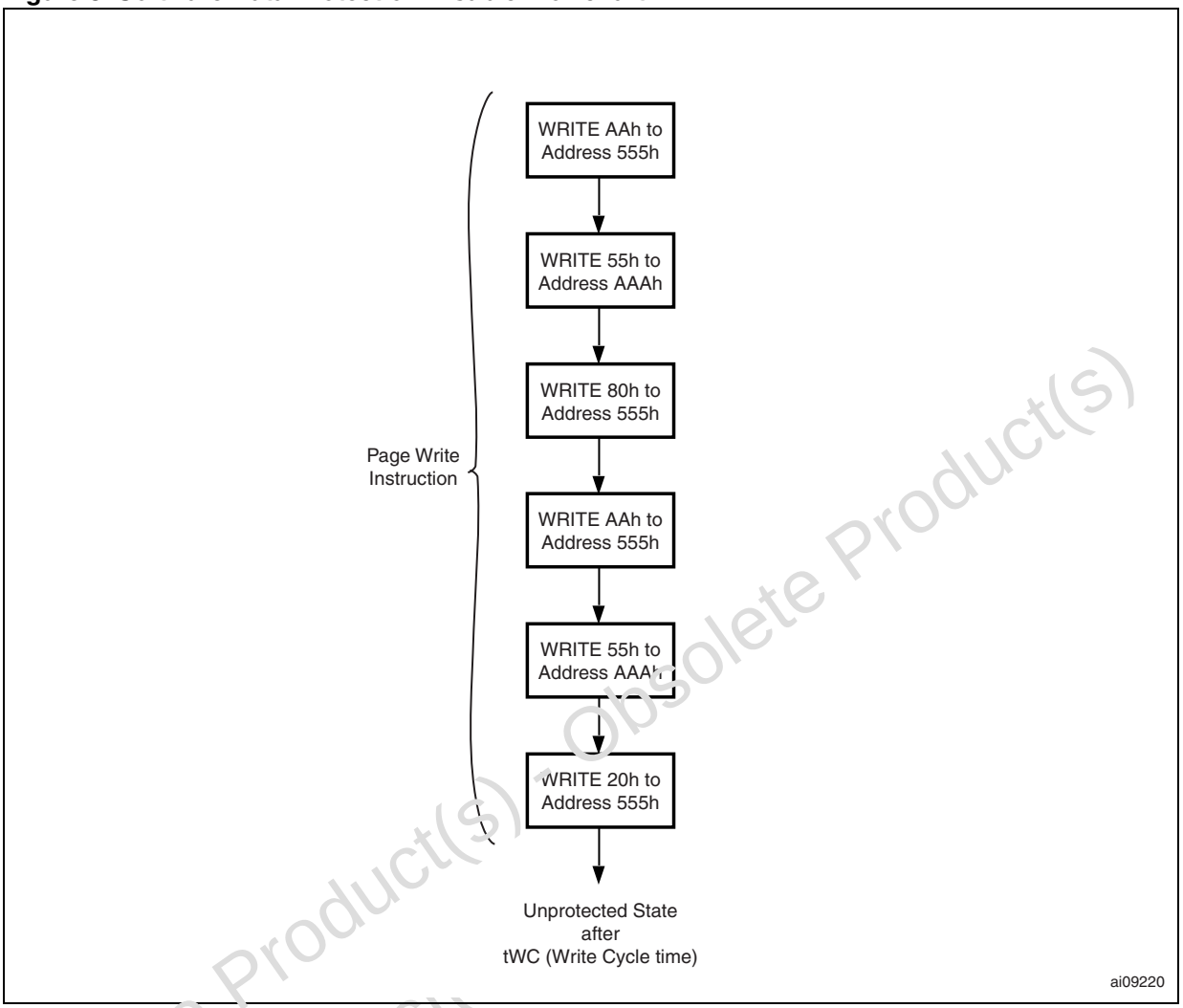
During the WRITE operations, address bit A6 must be zero, while address bits A5-A0 define the OTP Row byte to be written while any EEPROM Sector Select signal (EESi) is active. Writing the OTP Row is allowed only when SDP mode is not enabled.

**Figure 7. EEPROM SDP Enable Flowcharts**




Obsolete Product(s) - Obsolete Product(s)

**Figure 8. Software Data Protection Disable Flowchart**




Obsolete Product(s) - Obsolete Product(s)

## PROGRAMMING FLASH MEMORY

Flash memory must be erased prior to being programmed. The MCU may erase Flash memory all at once or by-sector, but not byte-by-byte. A byte of Flash memory erases to all logic ones (FF hex), and its bits are programmed to logic zeros. Although erasing Flash memory occurs on a sector basis, programming Flash memory occurs on a byte basis.

The PSD main Flash and optional boot Flash require the MCU to send an instruction to program a byte or perform an erase function (see Table 8., page 20). This differs from EEPROM, which can be programmed with simple MCU bus write operations (unless EEPROM SDP mode is enabled).

Once the MCU issues a Flash memory program or erase instruction, it must check for the status of completion. The embedded algorithms that are invoked inside the PSD support several means to provide status to the MCU. Status may be checked using any of three methods: Data Polling, Data Toggle, or the Ready/Busy output pin.

### Data Polling

Polling on DQ7 is a method of checking whether a Program or Erase instruction is in progress or has completed. Figure 9 shows the Data Polling algorithm.

When the MCU issues a programming instruction, the embedded algorithm within the PSD begins. The MCU then reads the location of the byte to be programmed in Flash to check status. Data bit DQ7 of this location becomes the compliment of data bit 7of the original data byte to be programmed. The MCU continues to poll this location, comparing DQ7 and monitoring the Error bit on DQ5. When the DQ7 matches data bit 7 of the original data and the Error bit at DQ5 remains '0', then the embedded algorithm is complete. If the Error bit at DQ5 is '1', the MCU should test DQ7 again since DQ7 may have changed simultaneously with DQ5 (see Figure 9).

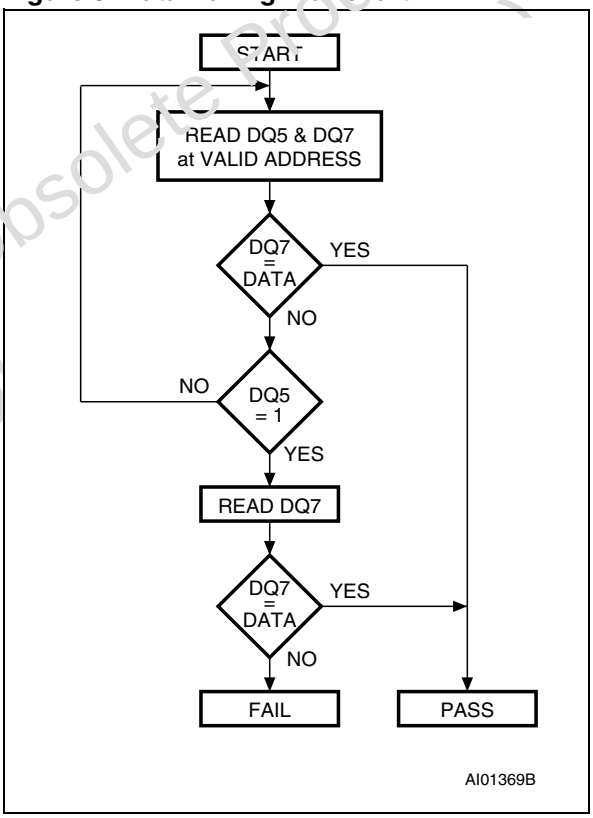
The Error bit at DQ5 will be set if either an internal timeout occurred while the embedded algorithm attempted to program the byte or if the MCU attempted to program a '1' to a bit that was not erased (not erased is logic '0').

It is suggested (as with all Flash memories) to read the location again after the embedded programming algorithm has completed to compare the byte that was written to Flash with the byte that was intended to be written.

When using the Data Polling method after an erase instruction, Figure 9 still applies. However, DQ7 will be '0' until the erase operation is complete. A '1' on DQ5 will indicate a timeout failure of the erase operation, a '0' indicates no error. The MCU can read any location within the sector being erased to get DQ7 and DQ5.

PSDsoft Express will generate ANSI C code functions which implement these Data Polling algorithms.

**Figure 9. Data Polling Flowchart**




Obsolete Product(s) - Obsolete Product(s)

### Data Toggle

Checking the Data Toggle bit on DQ6 is a method of determining whether a Program or Erase instruction is in progress or has completed. Figure 10 shows the Data Toggle algorithm.

When the MCU issues a programming instruction, the embedded algorithm within the PSD begins. The MCU then reads the location of the byte to be programmed in Flash to check status. Data bit DQ6 of this location will toggle each time the MCU reads this location until the embedded algorithm is complete. The MCU continues to read this location, checking DQ6 and monitoring the Error bit on DQ5. When DQ6 stops toggling (two consecutive reads yield the same value), and the Error bit on DQ5 remains '0', then the embedded algorithm is complete. If the Error bit on DQ5 is '1', the MCU should test DQ6 again, since DQ6 may have changed simultaneously with DQ5 (see Figure 10).

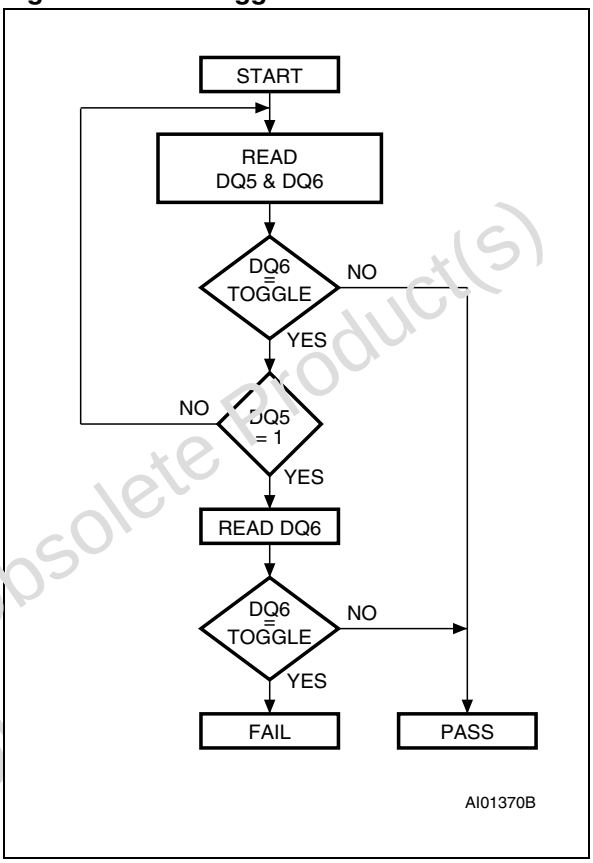
The Error bit at DQ5 will be set if either an internal timeout occurred while the embedded algorithm attempted to program the byte, or if the MCU attempted to program a '1' to a bit that was not erased (not erased is logic '0').

It is suggested (as with all Flash memories) to read the location again after the embedded programming algorithm has completed to compare the byte that was written to Flash with the byte that was intended to be written.

When using the Data Toggle method after an erase instruction, Figure 10 still applies. DQ6 will toggle until the erase operation is complete. A '1' on DQ5 will indicate a timeout failure of the erase operation, a '0' indicates no error. The MCU can read any location within the sector being erased to get DQ6 and DQ5.

PSDsoft Express will generate ANSI C code functions which implement these Data Toggling algorithms.

**Figure 10. Data Toggle Flowchart**




Obsolete Product(s)

## 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 8., page 20. 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 status bits DQ5, DQ6, and DQ7, as detailed in section entitled PROGRAMMING FLASH MEMORY, page 27. The Error 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 array with 00h because the PSD will automatically do this before erasing to 0FFh.

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

**Flash Sector Erase.** The Sector Erase instruction uses six write operations, as described in Table 8., page 20. Additional Flash Sector Erase confirm commands and Flash sector addresses can be written subsequently to erase other Flash sectors in parallel, without further coded cycles, if the additional instruction is transmitted in a shorter time than the timeout period of about 100 µs. The input of a new Sector Erase instruction will restart the time-out period.

The status of the internal timer can be monitored through the level of DQ3 (Erase time-out bit). If DQ3 is '0', the Sector Erase instruction has been received and the timeout is counting. If DQ3 is '1', the timeout has expired and the PSD is busy erasing the Flash sector(s). Before and during Erase timeout, any instruction other than Erase suspend and Erase Resume will abort the instruction and reset the device to READ mode. It is not necessary to program the Flash sector with 00h as the PSD will do this automatically before erasing (byte=FFh).

During a Sector Erase, the memory status may be checked by reading status bits DQ5, DQ6, and DQ7, as detailed in section entitled PROGRAMMING FLASH MEMORY, page 27.

During execution of the erase instruction, the Flash block logic accepts only Reset and Erase Suspend instructions. Erasure of one Flash sector may be suspended, in order to read data from another Flash sector, and then resumed.

### Flash Erase Suspend

When a Flash Sector Erase operation is in progress, the Erase Suspend instruction will suspend the operation by writing 0B0h to any address when an appropriate Chip Select (FSi) is true. (See Table 8., page 20). This allows reading of data from another Flash sector after the Erase operation has been suspended. Erase suspend is accepted only during the Flash Sector Erase instruction execution and defaults to READ mode. An Erase Suspend instruction executed during an Erase timeout will, in addition to suspending the erase, terminate the time out.

The Toggle Bit DQ6 stops toggling when the PSD internal logic is suspended. The toggle Bit status must be monitored at an address within the Flash sector being erased. The Toggle Bit will stop toggling between 0.1 µs and 15 µs after the Erase Suspend instruction has been executed. The PSD will then automatically be set to Read Flash Block Memory Array mode.

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

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

### Flash Erase Resume

If an Erase Suspend instruction was previously executed, the erase operation may be resumed by this instruction. The Erase Resume instruction consists of writing 030h to any address while an appropriate Chip Select (FSi) is true. (See Table 8., page 20.)

Obsolete Product(s) - Obsolete Product(s)

## FLASH AND EEPROM MEMORY SPECIFIC FEATURES

### Flash Memory and EEPROM Sector Protect

Each Flash and EEPROM sector can be separately protected against Program and Erase functions. Sector Protection provides additional data security because it disables all program or erase operations. This mode can be activated through the JTAG Port or a Device Programmer.

Sector protection can be selected for each sector using the PSDsoft Configuration program. This will automatically protect selected sectors when the device is programmed through the JTAG Port or a Device Programmer. Flash and EEPROM sectors can be unprotected to allow updating of their contents using the JTAG Port or a Device Programmer. The microcontroller can read (but cannot change) the sector protection bits.

Any attempt to program or erase a protected Flash or EEPROM sector will be ignored by the device. The Verify operation will result 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 protection and PSD/EE protection registers (CSIOP). See Table 10.

### Reset

The Reset instruction resets the internal memory logic state machine in a few milliseconds. Reset is an instruction of either one write operation or three write operations (refer to Table 8., page 20).

**Table 10. 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: 1. Bit Definitions:
**Sec<i>_Prot** 1 = Flash <i> is write protected.
**Sec<i>_Prot** 0 = Flash <i> is not write protected.

**Table 11. Sector Protection/Security Bit Definition – PSD/EE 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: 1. Bit Definitions:
**Sec<i>_Prot** 1 = EEPROM Boot Sector <i> is write protected.
**Sec<i>_Prot** 0 = EEPROM Boot 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.

## SRAM

The SRAM is a 16 Kbit (2K x 8) memory. The SRAM is enabled when RS0—the SRAM chip select output from the DPLD—is high. RS0 can contain up to two product terms, allowing flexible memory mapping.

The chip select signal (RS0) for the SRAM is configured using PSDsoft Express Configuration.

Obsolete Product(s) - Obsolete Product(s)

## MEMORY SELECT SIGNALS

The main Flash (FSi), EEPROM (EESi), and SRAM (RS0) memory select signals are all outputs of the DPLD. They are setup by entering equations for them in PSDsoft Express. The following rules apply to the equations for the internal chip select signals:

1. Flash memory and EEPROM sector select signals must not be larger than the physical sector size.
2. Any main Flash memory sector must not be mapped in the same memory space as another Flash sector.
3. An EEPROM sector must not be mapped in the same memory space as another EEPROM sector.
4. SRAM, I/O, and Peripheral I/O spaces must not overlap.
5. An EEPROM sector may overlap a main Flash memory sector. In case of overlap, priority will be given to the EEPROM.
6. SRAM, I/O, and Peripheral I/O spaces may overlap any other memory sector. Priority will be 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, EES0 is valid from 8000h to 9FFFh, and RS0 is valid from 8000h to 87FFh. Any address in the range of RS0 will always access the SRAM. Any address in the range of EES0 greater than 87FFh (and less than 9FFFh) will automatically address EEPROM segment 0. Any address greater than 9FFFh will access the Flash memory segment 0. You can see that half of the Flash memory segment 0 and one-fourth of EEPROM segment 0 can not be accessed in this example. Also note that an equation that defined FS1 to anywhere in the range of 8000h to BFFFh would not be valid.

Figure 11 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 for MCUs with Separate Program and Data Spaces**

The 8031 and compatible family of microcontrollers, which includes the 80C51, 80C151, 80C251, 80C51XA, and the C500 family, have separate address spaces for code memory (selected using $\overline{PSEN}$) and data memory (selected using $\overline{RD}$). Any of the memories within the PSD can reside in either space or both spaces. This is controlled through manipulation of the VM register that resides in the PSD's CSIOP space.

The VM register is set using PSDsoft Express to have an initial value. It can subsequently be changed by the microcontroller so that memory mapping can be changed on-the-fly. For example, I may wish to have SRAM and Flash in Data Space at boot, and EEPROM in Program Space at boot, and later swap EEPROM and Flash. This is easily done with the VM register by using PSDsoft Express to configure it for boot up and having the microcontroller change it when desired. Table 12 describes the VM Register.

**Figure 11. Priority Level of Memory and I/O Components**

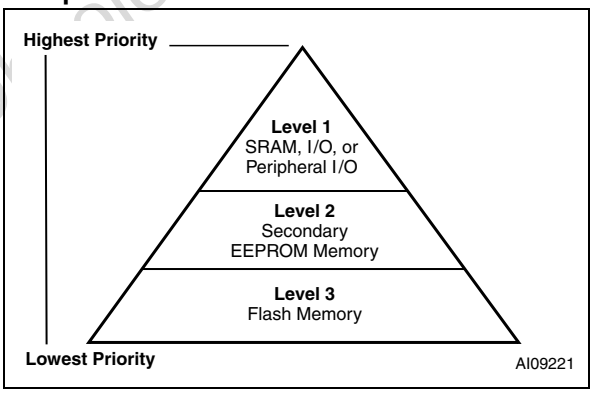



**Table 12. VM Register**

| Bit 7<br>PIO_EN | Bit 6 | Bit 5 | Bit 4<br>FL_Data | Bit 3<br>EE_Data | Bit 2<br>FL_Code | Bit 1<br>EE_Code | Bit 0<br>SRAM_Code |
|---|---|---|---|---|---|---|---|
| 0 = disable PIO mode | not used | not used | 0 = $\overline{RD}$ can't access Flash memory | 0 = $\overline{RD}$ can't access EEPROM | 0 = $\overline{PSEN}$ can't access Flash memory | 0 = $\overline{PSEN}$ can't access EEPROM | 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 EEPROM | 1 = $\overline{PSEN}$ access Flash memory | 1 = $\overline{PSEN}$ access EEPROM | 1 = $\overline{PSEN}$ access SRAM |

Obsolete Product(s) - Obsolete Product(s)

### Separate Space Modes

Code memory space is separated from data memory space. For example, the PSEN signal is used to access the program code from the Flash Memory, while the RD signal is used to access data from the EEPROM, SRAM and I/O Ports. This configuration requires the VM register to be set to 0Ch. See Figure 12.

### Combined Space Modes

The program and data memory spaces are combined into one space that allows the main Flash Memory, EEPROM, and SRAM to be accessed by either PSEN or RD. For example, to configure the main Flash memory in combined space mode, bits 2 and 4 of the VM register are set to "1" (see Figure 13).

**Figure 12. 80C31 Memory Modes - Separate Space**

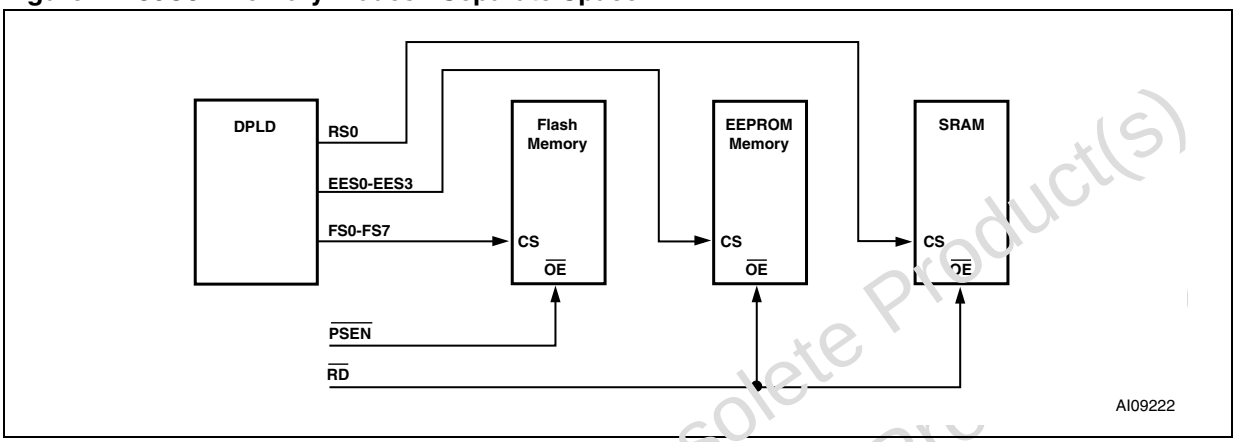
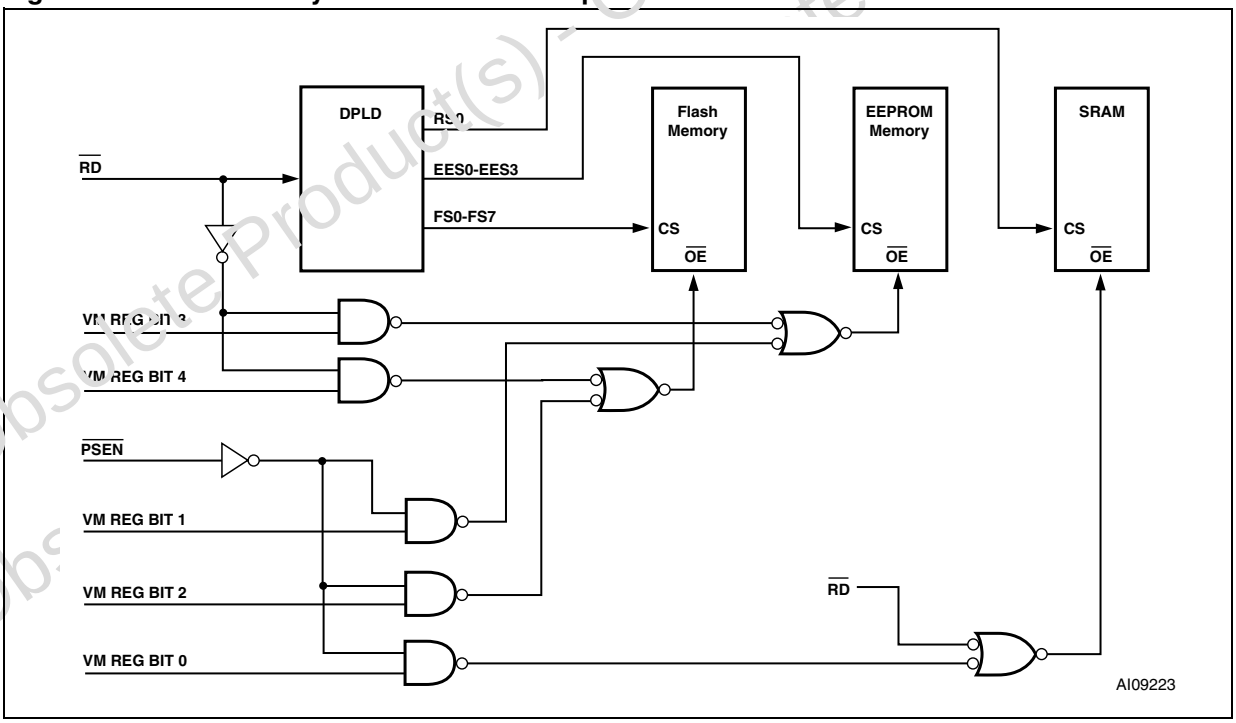



**Figure 13. 80C31 Memory Mode - Combined Space**




Obsolete Product(s) - Obsolete Product(s)

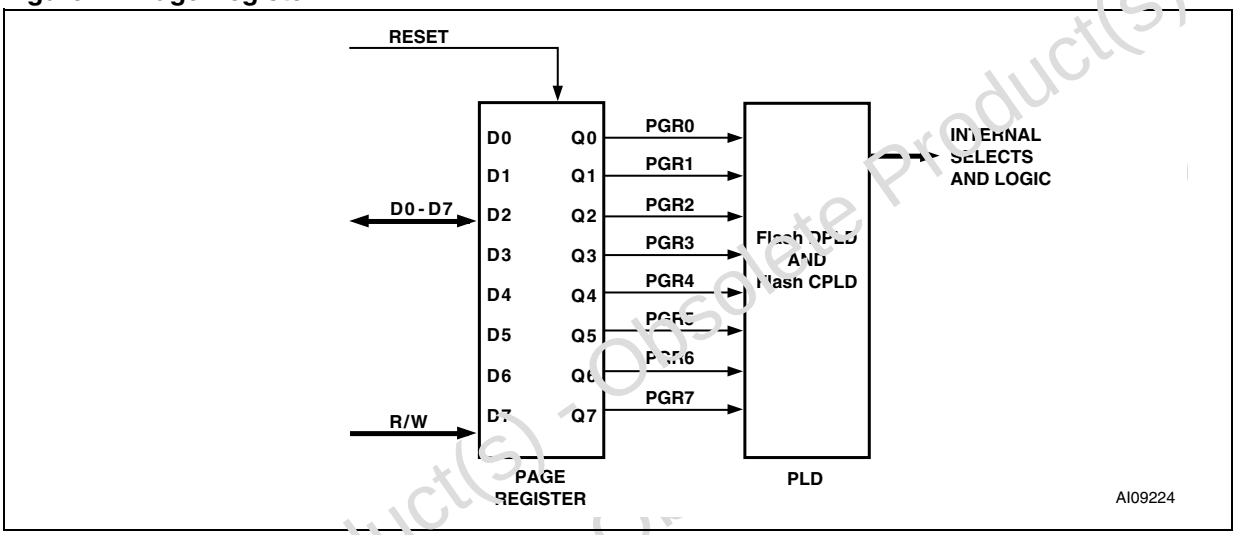
## PAGE REGISTER

The 8-bit Page Register increases the addressing capability of the microcontroller by a factor of up to 256. The contents of the register can also be read by the microcontroller. The outputs of the Page Register (PGR0-PGR7) are inputs to the DPLD decoder and can be included in the Flash Memory, EEPROM, and SRAM chip select 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 14 shows the Page Register. The eight flip flops in the register are connected to the internal data bus D0-D7. The microcontroller can write to or read from the Page Register. The Page Register can be accessed at address location CSIOP + E0h.

**Figure 14. Page Register**




Obsolete Product(s) - Obsolete Product(s)

## PLD'S

The PLDs bring programmable logic functionality to the PSD. After specifying the logic for the PLDs using the PSDabel tool in PSDsoft Express, the logic is programmed into the device and available upon power-up.

The PSD contains two PLDs: the Decode PLD (DPLD), and the Complex PLD (CPLD). The PLDs are briefly discussed in the next few paragraphs, and in more detail in the sections entitled DECODE PLD (DPLD) and COMPLEX PLD (CPLD). Figure 15., page 35 shows the configuration of the PLDs.

The DPLD performs address decoding for internal and external components, such as memory, registers, and I/O port selects.

The CPLD can be used for logic functions, such as loadable counters and shift registers, state machines, and encoding and decoding logic. These logic functions can be constructed using the 16 Output macrocells (OMCs), 24 Input macrocells (IMCs), and the AND array. The CPLD can also be used to generate external chip selects.

The AND array is used to form product terms. These product terms are specified using PSDabel. An Input Bus consisting of 73 signals is connected to the PLDs. The signals are shown in Table 13.

### The Turbo Bit in PSD

The PLDs in the PSD can minimize power consumption by switching off when inputs remain unchanged for an extended time of about 70ns. Setting the Turbo mode bit to off (Bit 3 of the PMMR0 register) automatically places the PLDs into standby if no inputs are changing. Turbo-off mode increases propagation delays while reducing power consumption. See the section entitled POWER MANAGEMENT, page 64, on how to set the Turbo Bit.

Additionally, five bits are available in the PMMR2 register to block MCU control signals from entering the PLDs. This reduces power consumption and can be used only when these MCU control signals are not used in PLD logic equations.

The PLDs in the PSD can minimize power consumption by switching off when inputs remain unchanged for an extended time of about 70ns. Each of the two PLDs has unique characteristics suited for its applications. They are described in the following sections.

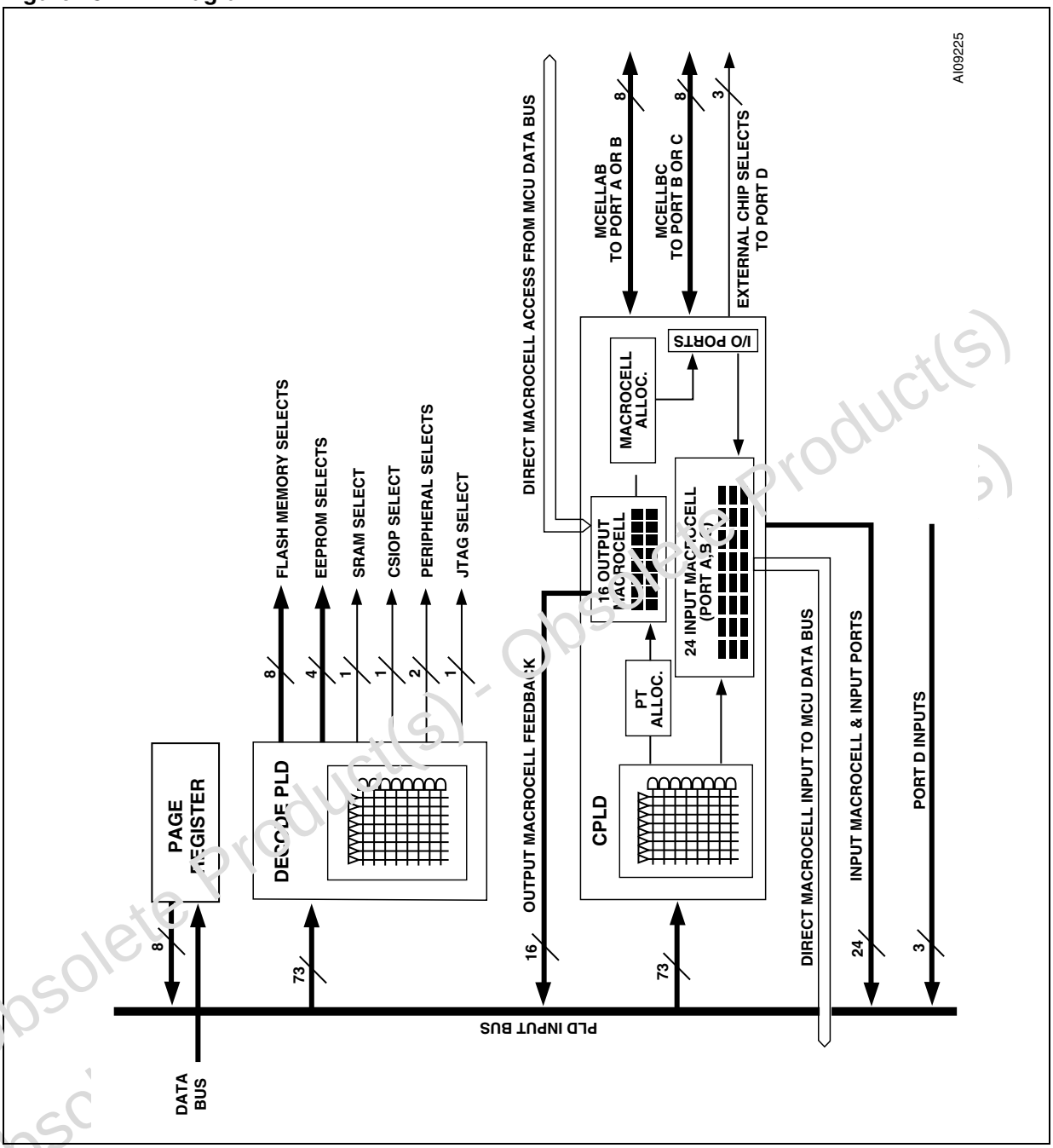
**Table 13. DPLD and CPLD Inputs**

| Input Source | Input Name | Number of Signals |
|---|---|---|
| MCU Address Bus[1] | A15-A0 | 16 |
| MCU Control Signals | CNTL2-CNTL0 | 3 |
| Reset | RST | 1 |
| Power-down | PDN | 1 |
| Port A Input Macrocells | PA7-PA0 | 8 |
| Port B Input Macrocells | PB7-PB0 | 8 |
| Port C Input Macrocells | PC7-PC0 | 8 |
| Port D Inputs | PD2-PD0 | 3 |
| Page Register | PGR7-PGR0 | 8 |
| Macrocell AB Feedback | MCELLAB.FB7-FB0 | 8 |
| Macrocell BC Feedback | MCELLBC.FB7-FB0 | 8 |
| EEPROM Program Status Bit | Ready/$\overline{\text{Busy}}$ | 1 |

Note: 1. The address inputs are A19-A4 in 80C51XA mode.

Obsolete Product(s) - Obsolete Product(s)

**Figure 15. PLD Diagram**



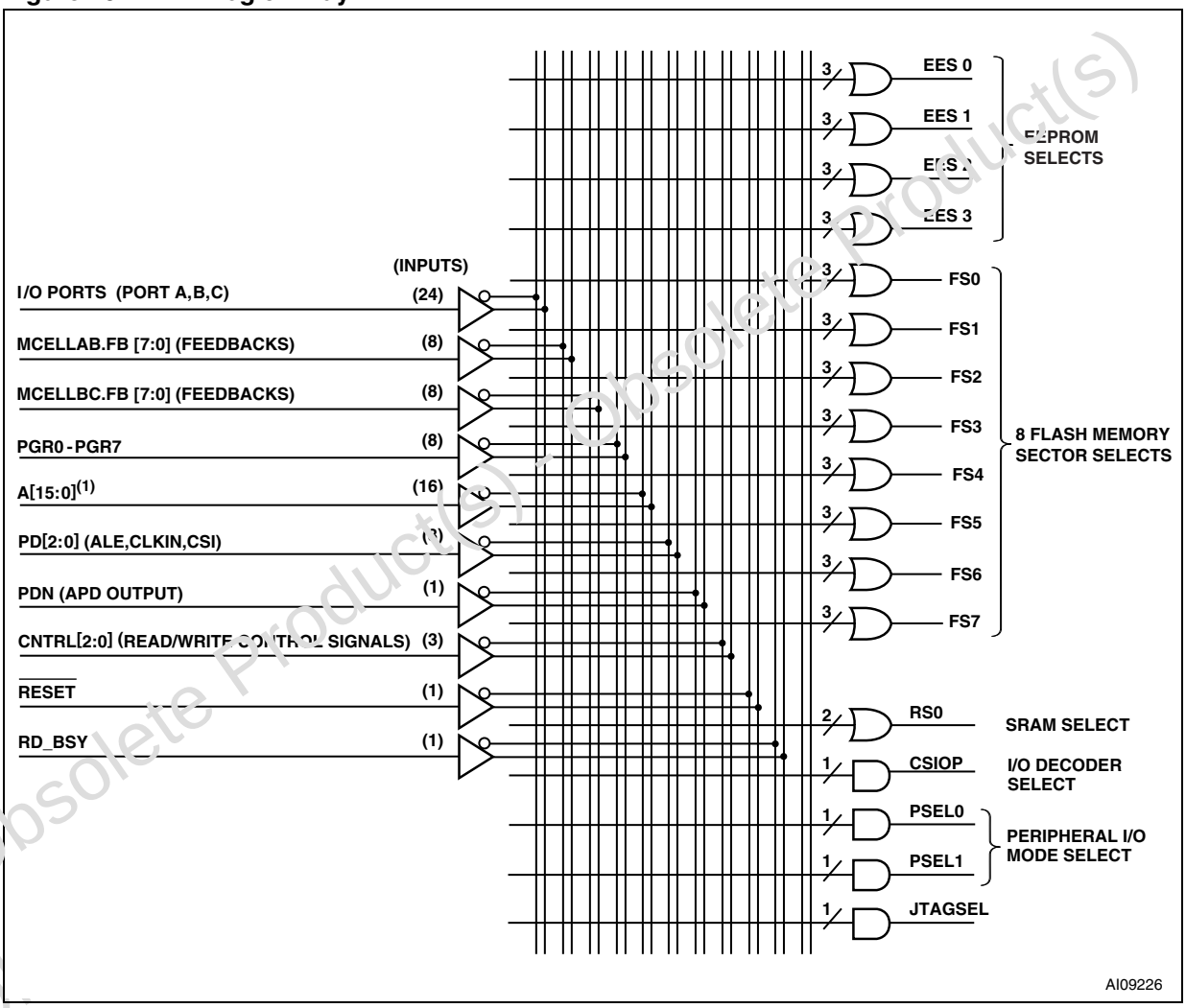

Obsolete Product(s) - Obsolete Product(s)

## DECODE PLD (DPLD)

The DPLD, shown in Figure 16, is used for decoding the address for internal and external components. The DPLD can be used to generate the following decode signals:

- 8 sector selects for the main Flash memory (three product terms each)
- 4 sector selects for the EEPROM (three product terms each)
- 1 internal SRAM select signal (two product terms)
- 1 internal CSIOP (PSD configuration register) select signal
- 1 JTAG select signal (enables JTAG on Port C)
- 2 internal peripheral select signals (peripheral I/O mode).

**Figure 16. DPLD Logic Array**




Obsolete Product(s) - Obsolete Product(s)

Note: 1. The address inputs are A19-A4 in 80C51XA mode.

## 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 3 external chip selects, routed to Port D.

Although external chip selects can be produced by any Output Macrocell, these three external chip selects on Port D do not consume any Output macrocells.

As shown in Figure 15., page 35, the CPLD has the following blocks:

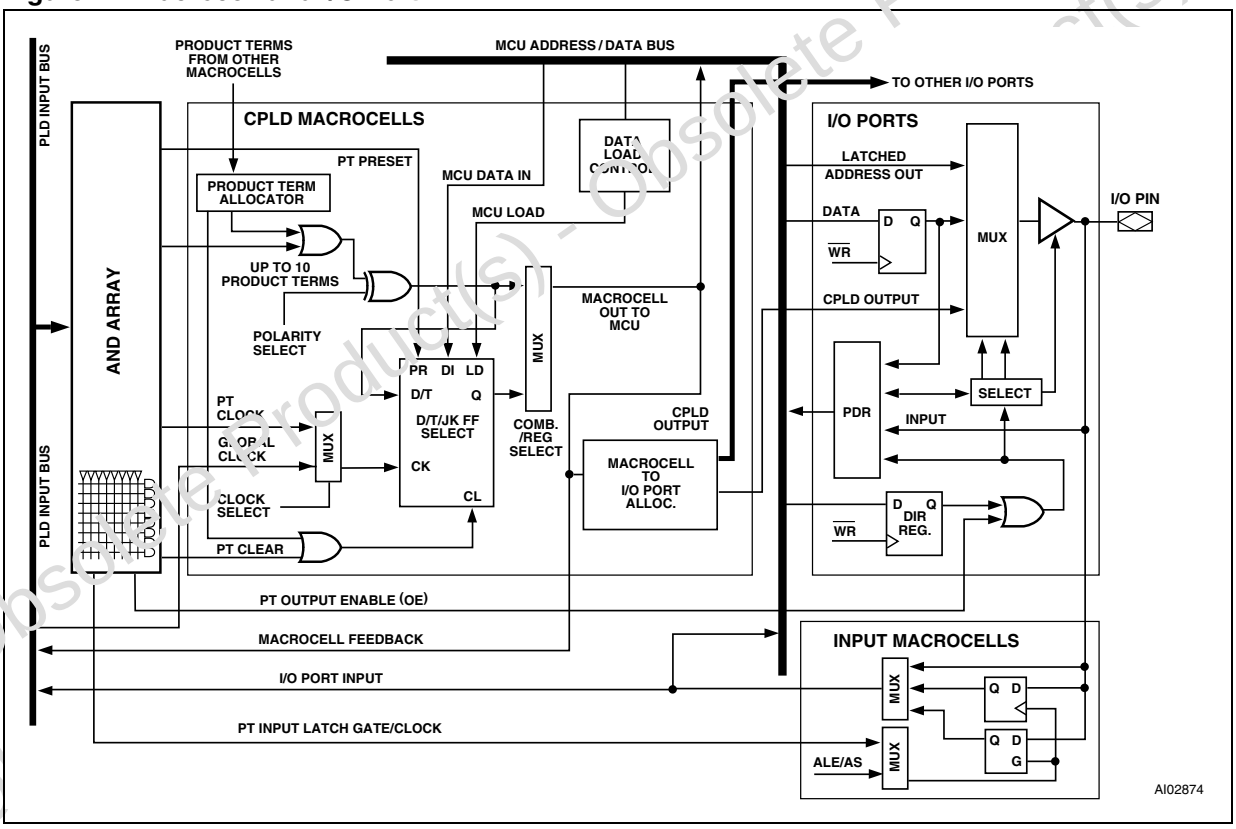
- 24 Input macrocells (IMCs)
- 16 Output macrocells (OMCs)
- Macrocell Allocator
- Product Term Allocator
- AND array capable of generating up to 137 product terms
- Four I/O ports.

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

The Input Macrocells (IMC) and Output Macrocells (OMC) are connected to the PSD internal data bus and can be directly accessed by the microcontroller. 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 logic array as required in most standard PLD macrocell architectures.

**Figure 17. Macrocell and I/O Port**




Obsolete Product(s) - Obsolete Product(s)

### 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 PSDabel, the Macrocell Allocator will assign it to either Port A or B. The same is true for a McellBC output on Port B or C. Table 14 shows the macrocells and Port assignment.

The Output Macrocell (OMC) architecture is shown in Figure 18., page 40. As shown in the figure, there are native product terms available from the AND array, and borrowed product terms available (if unused) from other OMCs. The polarity of the product term is controlled by the XOR gate.

The OMC can implement either sequential logic, using the flip-flop element, or combinatorial logic. The multiplexer selects between the sequential or combinatorial logic outputs. The multiplexer output can drive a Port pin and has a feedback path to the AND array inputs.

The flip-flop in the OMC can be configured as a D, T, JK, or SR type in the PSDabel program. The flip-flop's clock, preset, and clear inputs may be driven from a product term of the AND array. Alternatively, the external CLKIN signal can be used for the clock input to the flip-flop. The flip-flop is clocked on the rising edge of the clock input. The preset and clear are active-high inputs. Each clear input can use up to two product terms.

**Table 14. Output Macrocell Port and Data Bit Assignments**

| Output Macrocell | Port Assignment | 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, C0 | 4 | 5 | D0 |
| McellBC1 | Port B1, C1 | 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, C5 | 4 | 6 | D5 |
| McellBC6 | Port B6, C6 | 4 | 6 | D6 |
| McellBC7 | Port B7, C7 | 4 | 6 | D7 |

Obsolete Product(s) - Obsolete Product(s)

### Product Term Allocator

The CPLD has a Product Term Allocator. The PSDabel compiler uses the Product Term Allocator to borrow and place product terms from one macrocell to another. The following list summarizes how product terms are allocated:

- McellAB0-McellAB7 all have three native product terms and may borrow up to six more
- McellBC0-McellBC3 all have four native product terms and may borrow up to five more
- McellBC4-McellBC7 all have four native product terms and may borrow up to six more.

Each macrocell may only borrow product terms from certain other macrocells. Product terms already in use by one macrocell are not available for another macrocell.

If an equation requires more product terms than are available to it, then “external” product terms are required, which will consume other Output Macrocells (OMC). If external product terms are used, extra delay will be added for the equation that required the extra product terms.

This is called product term expansion. PSDsoft Express will perform this expansion as needed.

**Loading and Reading the Output Macrocells (OMC).** The OMCs occupy a memory location in the MCU address space, as defined by the CSIOP (refer to the I/O section). The flip-flops in each of the 16 OMCs can be loaded from the data bus by a microcontroller. Loading the OMCs with data from the MCU takes priority over internal functions. As such, the preset, clear, and clock inputs to the flip-flop can be overridden by the MCU. The ability to load the flip-flops and read them back is useful in such applications as loadable counters and shift registers, mailboxes, and handshaking protocols.

Data can be loaded to the OMCs on the trailing edge of the $\overline{WR}$ signal (edge loading) or during the time that the $\overline{WR}$ signal is active (level loading). The method of loading is specified in PSDsoft Express Configuration.

### The OMC Mask Register

There is one Mask Register for each of the two groups of eight OMCs. The Mask Registers can be used to block the loading of data to individual OMCs. The default value for the Mask Registers is 00h, which allows loading of the OMCs. When a given bit in a Mask Register is set to a ‘1’, the MCU will be blocked from writing to the associated OMC. For example, suppose McellAB0-3 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.

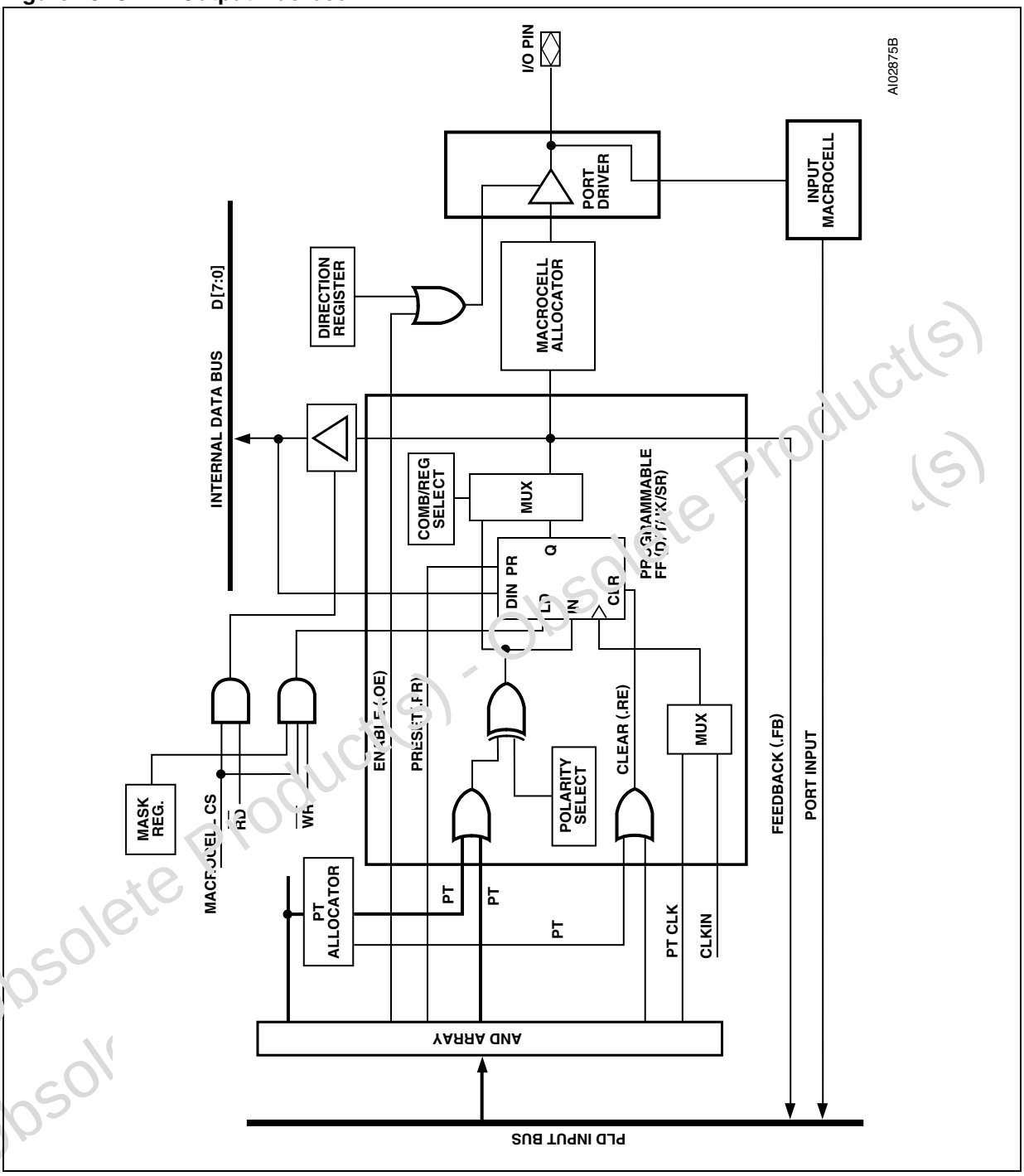
### The Output Enable of the OMC

The OMC 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 OMC output is declared as an internal node and not as a Port pin output in the PSDabel file, then 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.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Figure 18. CPLD Output Macrocell**




Obsolete Product(s) - Obsolete Product(s)

### Input Macrocells (IMC)

The CPLD has 24 IMCs, one for each pin on Ports A, B, and C. The architecture of the IMC is shown in Figure 19., page 42. The IMCs 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 IMCs can be read by the microcontroller 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/AS). Each product term output is used to latch or clock four IMCs. Port inputs 3-0 can be controlled by one product term and 7-4 by another.

Configurations for the IMCs are specified by equations written in PSDabel (see Application Note 55). Outputs of the IMCs can be read by the MCU via the IMC buffer. See the I/O Port section on how to read the IMCs.

IMCs can use the address strobe to latch address bits higher than A15. Any latched addresses are routed to the PLDs as inputs.

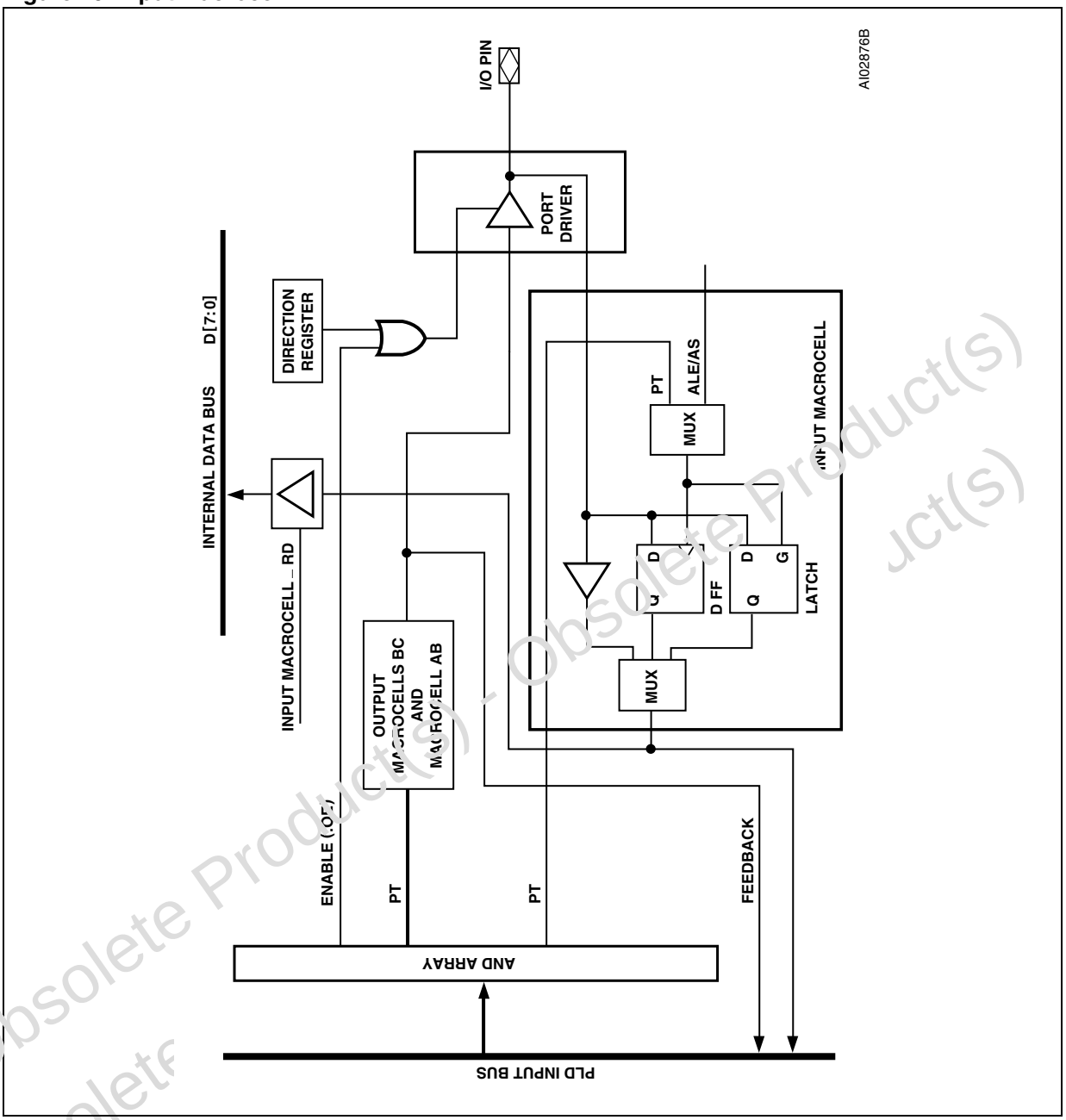
IMCs are particularly useful with handshaking communication applications where two processors pass data back and forth through a common mailbox. Figure 20., page 43 shows a typical configuration where the Master MCU writes to the Port A Data Out Register. This, in turn, can be read by the Slave MCU via the activation of the “Slave-Read” output enable product term.

The Slave can also write to the Port A IMCs and the Master can then read the IMCs directly.

Note that the “Slave-Read” and “Slave-wr” signals are product terms that are derived from the Slave MCU inputs $\overline{RD}$, $\overline{WR}$, and Slave_CS.

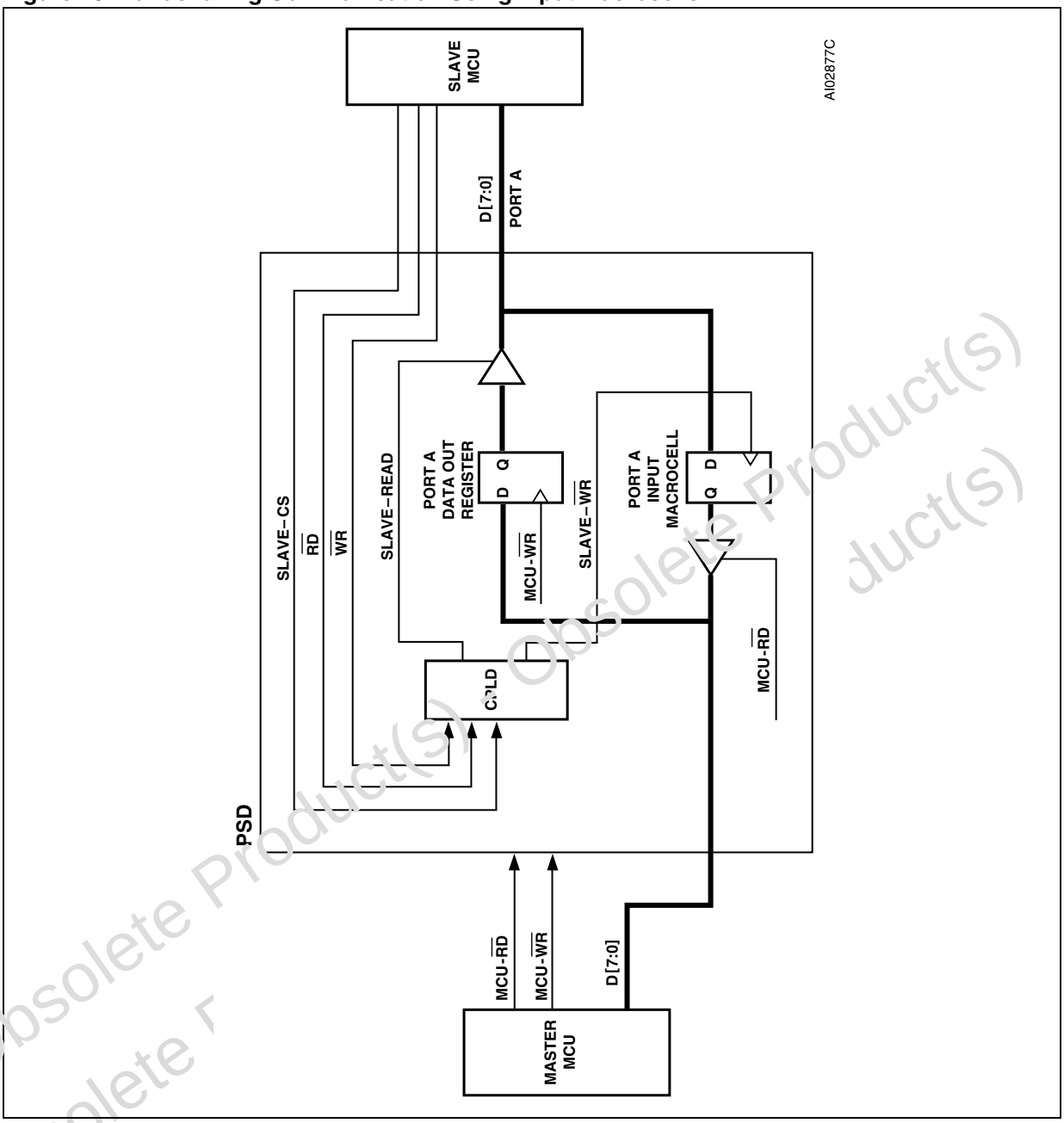
Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Figure 19. Input Macrocell**




Obsolete Product(s) - Obsolete Product(s)

**Figure 20. Handshaking Communication Using Input Macrocells**




Obsolete Product(s) Obsolete Product(s)

## MCU BUS INTERFACE

The “no-glue logic” PSD MCU Bus Interface block can be directly connected to most popular MCUs and their control signals.

Key 8-bit MCUs, with their bus types and control signals, are shown in Table 15. The interface type is specified using the PSDsoft Express Configuration.

**Table 15. MCUs and their Control Signals**

| MCU | Data Bus Width | CNTL0 | CNTL1 | CNTL2 | PC7 | PD0[2] | ADIO0 | PA3-PA0 | PA7-PA3 |
|---|---|---|---|---|---|---|---|---|---|
| 8031 | 8 | $\overline{WR}$ | $\overline{RD}$ | $\overline{PSEN}$ | (Note 1) | ALE | A0 | (Note 1) | (Note 1) |
| 80C51XA | 8 | $\overline{WR}$ | $\overline{RD}$ | $\overline{PSEN}$ | (Note 1) | ALE | A4 | A3-A0 | (Note 1) |
| 80C251 | 8 | $\overline{WR}$ | $\overline{PSEN}$ | (Note 1) | (Note 1) | ALE | A0 | (Note 1) | (Note 1) |
| 80C251 | 8 | $\overline{WR}$ | $\overline{RD}$ | $\overline{PSEN}$ | (Note 1) | ALE | A0 | (Note 1) | (Note 1) |
| 80198 | 8 | $\overline{WR}$ | $\overline{RD}$ | (Note 1) | (Note 1) | ALE | A0 | (Note 1) | (Note 1) |
| 68HC11 | 8 | R/$\overline{W}$ | E | (Note 1) | (Note 1) | AS | A0 | (Note 1) | (Note 1) |
| 68HC912 | 8 | R/$\overline{W}$ | E | (Note 1) | $\overline{DBE}$ | AS | A0 | (Note 1) | (Note 1) |
| Z80 | 8 | $\overline{WR}$ | $\overline{RD}$ | (Note 1) | (Note 1) | (Note 1) | A0 | D3-D0 | D7-D4 |
| Z8 | 8 | R/$\overline{W}$ | $\overline{DS}$ | (Note 1) | (Note 1) | AS | A0 | (Note 1) | (Note 1) |
| 68330 | 8 | R/$\overline{W}$ | $\overline{DS}$ | (Note 1) | (Note 1) | AS | A0 | (Note 1) | (Note 1) |
| M37702M2 | 8 | R/$\overline{W}$ | $\overline{E}$ | (Note 1) | (Note 1) | ALE | A0 | D3-D0 | D7-D4 |

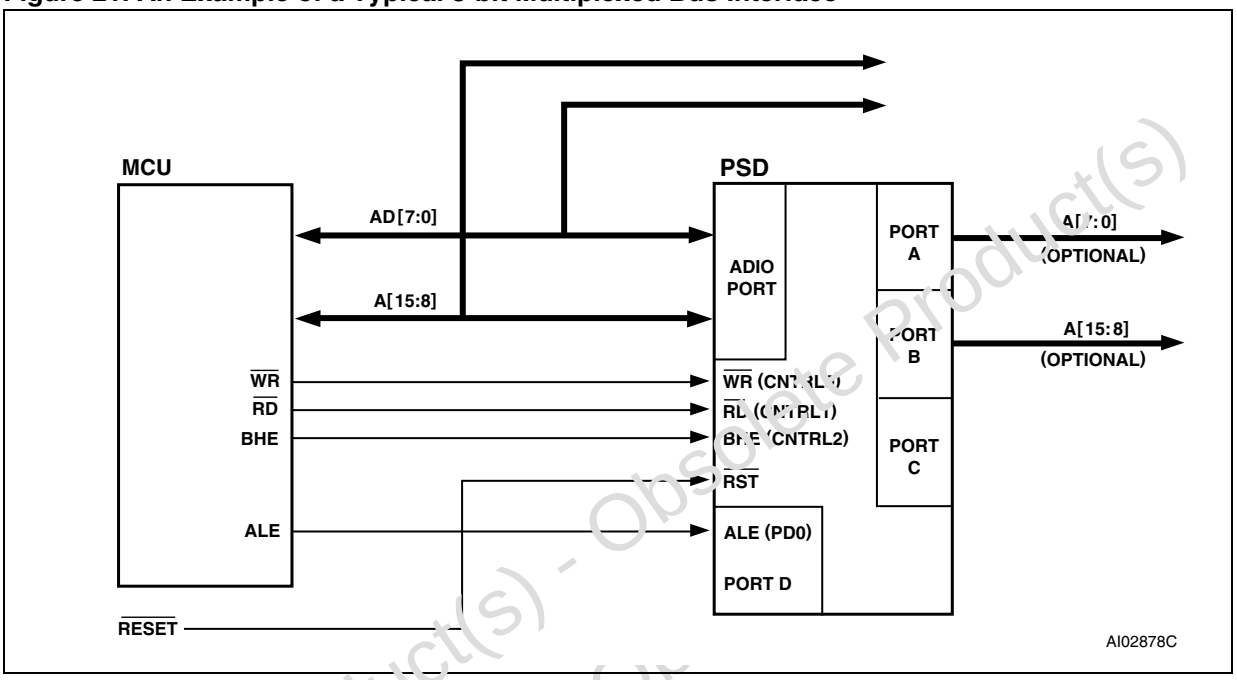
Note: 1. Unused CNTL2 pin can be configured as CPLD input. Other unused pins (PC7, PD0, PA3-0) can be configured for other I/O functions.
2. ALE/AS input is optional for MCUs with a non-multiplexed bus

Obsolete Product(s) - Obsolete Product(s)

### PSD Interface to a Multiplexed 8-Bit Bus

Figure 21 shows an example of a system using a MCU with an 8-bit multiplexed bus and a PSD. The ADIO port on the PSD is connected directly to the MCU address/data bus. Address Strobe (ALE/AS, PD0) latches the address signals internally. Latched addresses can be brought out to Port A or B. The PSD drives the ADIO data bus only when one of its internal resources is accessed and Read Strobe ($\overline{RD}$, CNTL1) is active. Should the system address bus exceed sixteen bits, Ports A, B, C, or D may be used as additional address inputs.

**Figure 21. An Example of a Typical 8-bit Multiplexed Bus Interface**

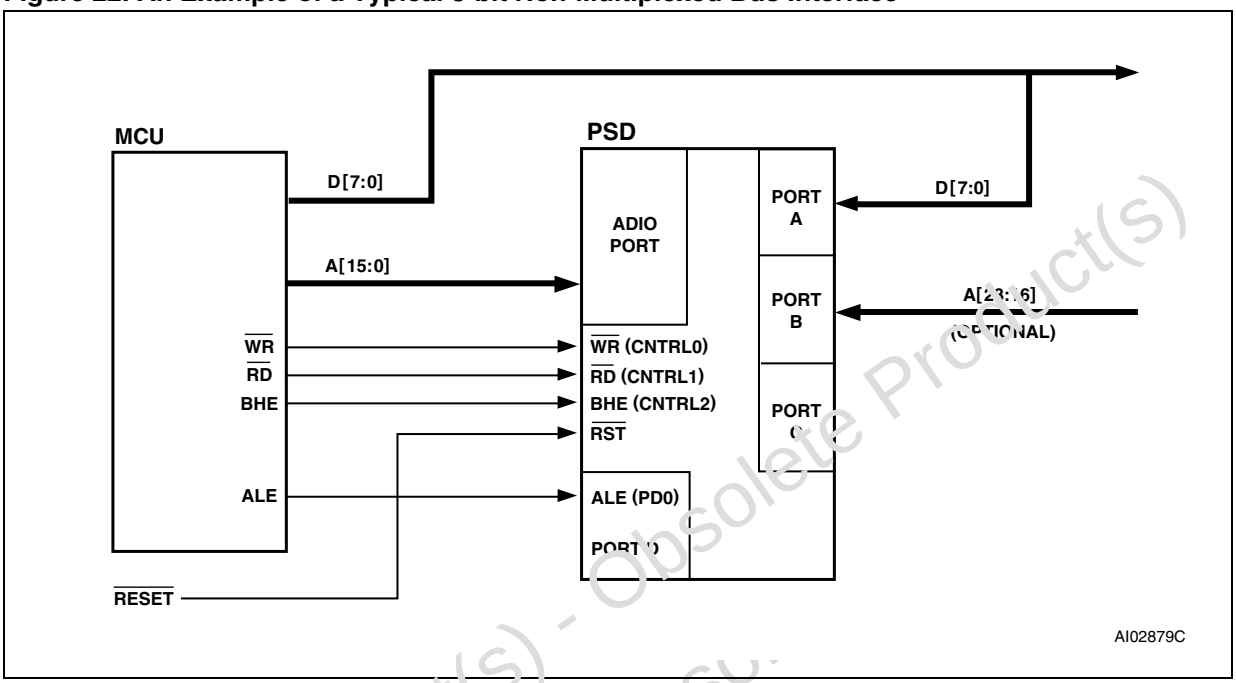



Obsolete Product(s) - Obsolete Product(s)

### PSD Interface to a Non-Multiplexed 8-Bit Bus

Figure 22 shows an example of a system using a microcontroller with an 8-bit non-multiplexed bus and a PSD. The address bus is connected to the ADIO Port, and the data bus is connected to Port A. Port A is in tri-state mode when the PSD is not accessed by the microcontroller. Should the system address bus exceed sixteen bits, Ports B, C, or D may be used for additional address inputs.

**Figure 22. An Example of a Typical 8-bit Non-Multiplexed Bus Interface**




Obsolete Product(s) - Obsolete Product(s)

### Data Byte Enable Reference

Microcontrollers have different data byte orientations. The following table shows how the PSD interprets byte/word operations in different bus WRITE configurations. Even-byte refers to locations with address A0 equal to zero and odd byte as locations with A0 equal to one.

**Table 16. Eight-Bit Data Bus**

| BHE | A0 | D7-D0 |
|---|---|---|
| X | 0 | Even Byte |
| X | 1 | Odd Byte |

### MCU Bus Interface Examples

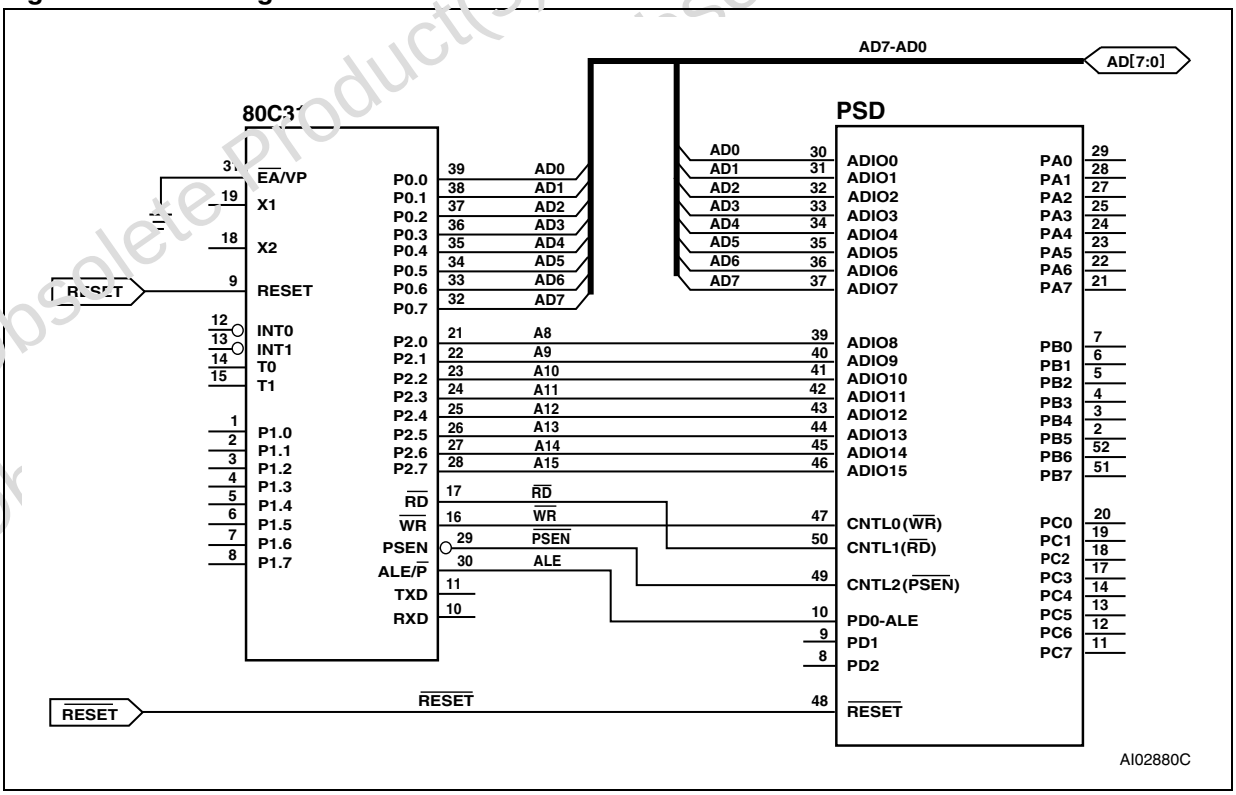
Figure 23 to 26 show examples of the basic connections between the PSD and some popular MCUs. The PSD Control input pins are labeled as to the MCU function for which they are configured. The MCU bus interface is specified using the PSDsoft Express Configuration.

The first configuration is 80C31-compatible, and the bus interface to the PSD is identical to that shown in Figure 23. The second and third configurations have the same bus connection as shown in Table 17., page 48. There is only one READ input ($\overline{PSEN}$) connected to the CNTL1 pin on the PSD. The A16 connection to the PA0 pin allows for a larger address input to the PSD. Configuration 4 is shown in Figure 24., page 49. The $\overline{RD}$ signal is connected to Cntl1 and the $\overline{PSEN}$ signal is connected to the CNTL2.

#### 80C31

Figure 23 shows the bus interface for the 80C31, which has an 8-bit multiplexed address/data bus. The lower address byte is multiplexed with the data bus. The MCU control signals Program Select Enable ($\overline{PSEN}$, CNTL2), Read Strobe ($\overline{RD}$, CNTL1), and Write Strobe ($\overline{WR}$, CNTL0) may be used for accessing the internal memory and I/O Ports. The ALE input (pin PD0) latches the address.

**Figure 23. Interfacing the PSD with an 80C31**




Obsolete Product(s) - Obsolete Product(s)

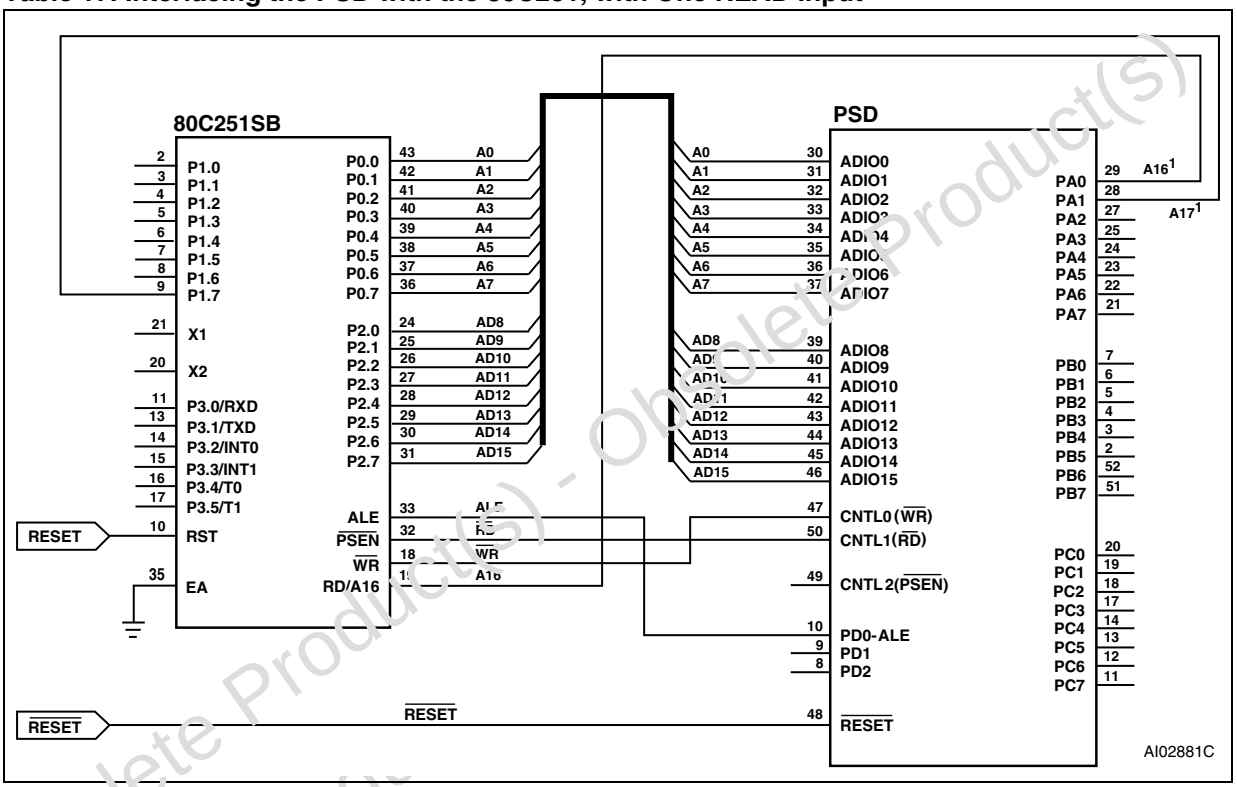
### 80C251

The Intel 80C251 MCU features a user-configurable bus interface with four possible bus configurations, as shown in Table 18., page 49.

The 80C251 has two major operating modes: Page Mode and Non-Page Mode. In Non-Page Mode, the data is multiplexed with the lower address byte, and ALE is active in every bus cycle.

In Page Mode, data D[7:0] is multiplexed with address A[15:8]. In a bus cycle where there is a Page hit, the ALE signal is not active and only addresses A[7:0] are changing. The PSD supports both modes. In Page Mode, the PSD bus timing is identical to Non-Page Mode except the address hold time and setup time with respect to ALE is not required. The PSD access time is measured from address A[7:0] valid to data in valid.

**Table 17. Interfacing the PSD with the 80C251, with One READ Input**




Note: 1. The A16 and A17 connections are optional.
2. In non-Page-Mode, AD7-AD0 connects to ADIO7-ADIO0.

Obsolete Product(s) - Obsolete Product(s)

**Figure 24. Interfacing the PSD with the 80C251, with $\overline{RD}$ and $\overline{PSEN}$ Inputs**

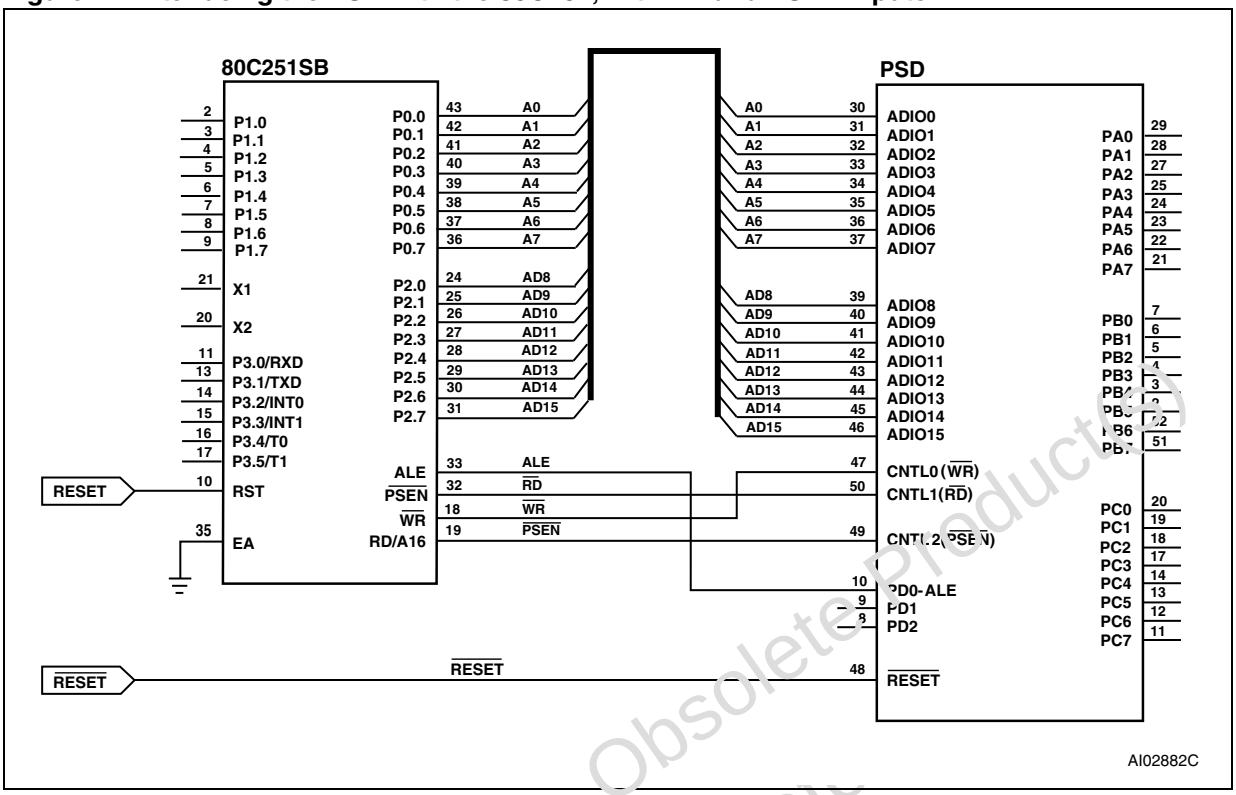



**Table 18. 80C251 Configurations**

| Configuration | 80C251 READ/WRITE Pins | Connecting to PSD Pins | Page Mode |
|---|---|---|---|
| 1 | $\overline{WR}$<br>$\overline{RD}$<br>$\overline{PSEN}$ | CNTL0<br>CNTL1<br>CNTL2 | Non-Page Mode, 80C31 compatible A7-A0 multiplex with D7-D0 |
| 2 | $\overline{WR}$<br>$\overline{PSEN}$ only | CNTL0<br>CNTL1 | Non-Page Mode<br>A7-A0 multiplex with D7-D0 |
| 3 | $\overline{WR}$<br>$\overline{PSEN}$ only | CNTL0<br>CNTL1 | Page Mode<br>A15-A8 multiplex with D7-D0 |
| 4 | $\overline{WR}$<br>$\overline{RD}$<br>$\overline{PSEN}$ | CNTL0<br>CNTL1<br>CNTL2 | Page Mode<br>A15-A8 multiplex with D7-D0 |

Obsolete Product(s) - Obsolete Product(s)

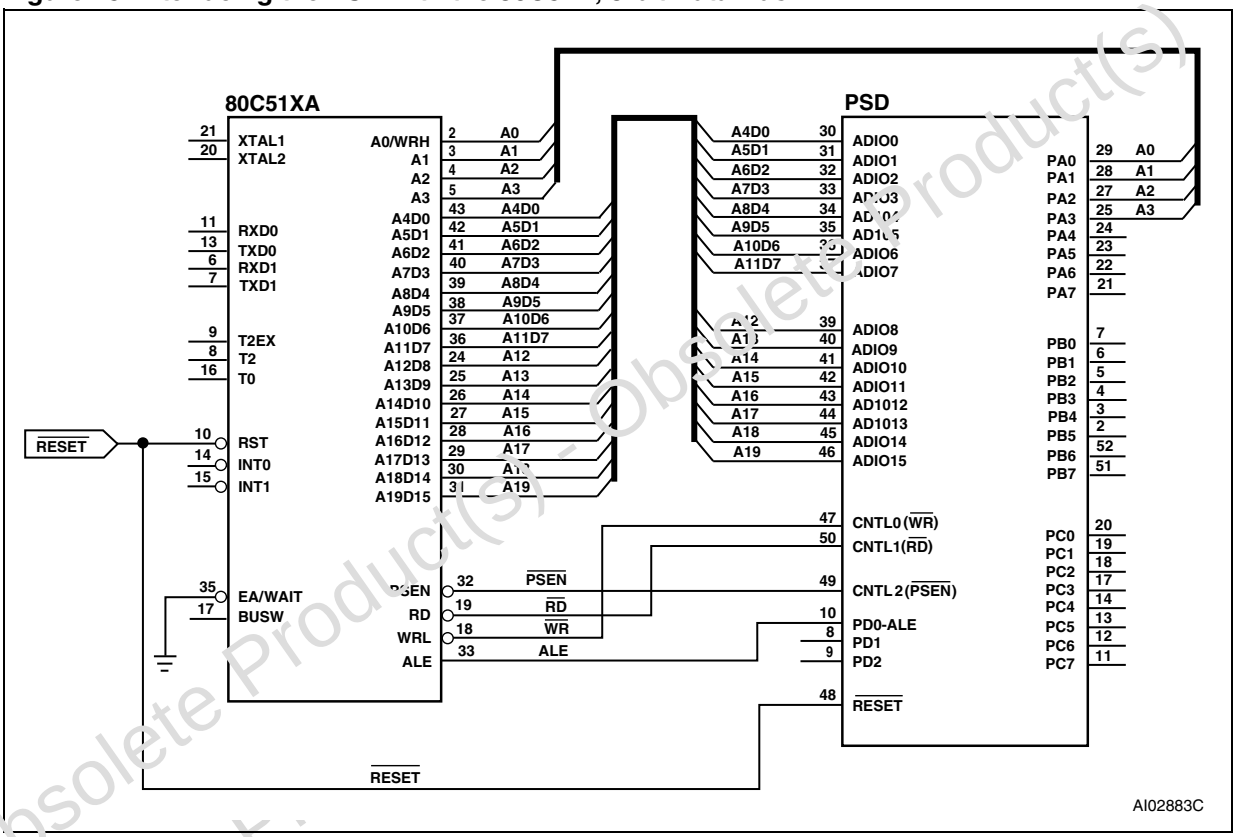
### 80C51XA

The Philips 80C51XA microcontroller family supports an 8- or 16-bit multiplexed bus that can have burst cycles. Address bits (A3-A0) are not multiplexed, while (A19-A4) are multiplexed with data bits (D15-D0) in 16-bit mode. In 8-bit mode, (A11-A4) are multiplexed with data bits (D7-D0).

The 80C51XA can be configured to operate in eight-bit data mode. (shown in Figure 25).

The 80C51XA improves bus throughput and performance by executing Burst cycles for code fetches. In Burst Mode, address A19-A4 are latched internally by the PSD, while the 80C51XA changes the A3-A0 lines to fetch up to 16 bytes of code. The PSD access time is then measured from address A3-A0 valid to data in valid. The PSD bus timing requirement in Burst Mode is identical to the normal bus cycle, except the address setup and hold time with respect to ALE does not apply.

**Figure 25. Interfacing the PSD with the 80C51X, 8-bit Data Bus**

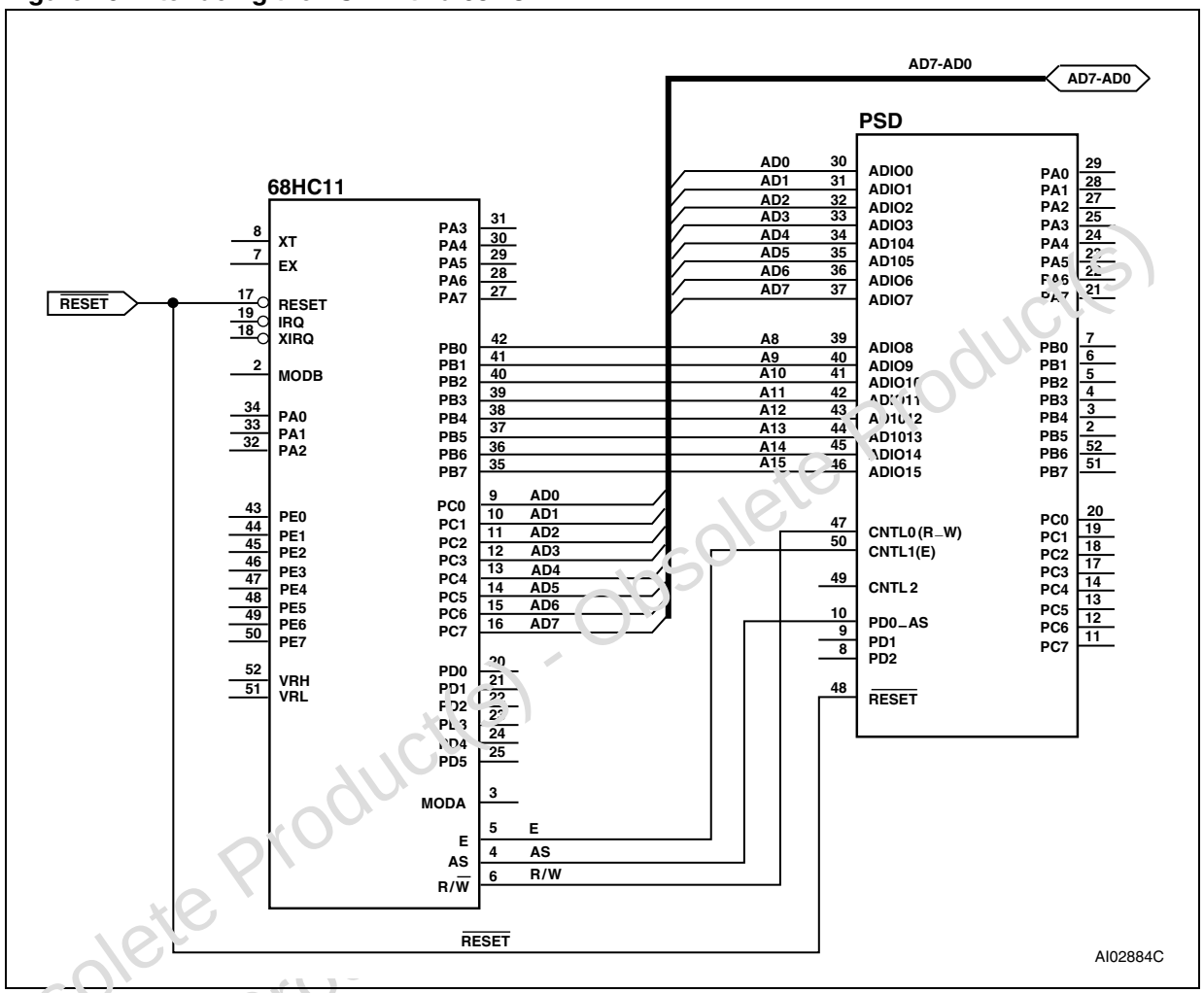



Obsolete Product(s) - Obsolete Product(s)

### 68HC11

Figure 26 shows an interface to a 68HC11 where the PSD is configured in 8-bit multiplexed mode with E and R/$\overline{W}$ settings. The DPLD can generate the READ and $\overline{WR}$ signals for external devices.

**Figure 26. Interfacing the PSD with a 68HC11**




Obsolete Product(s) - Obsolete Product(s)

## I/O PORTS

There are four programmable I/O ports: Ports A, B, C, and D. Each of the ports is eight bits except Port D, which is 3 bits. 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 address space.

The topics discussed in this section are:

- General Port architecture
- Port Operating Modes
- Port Configuration Registers (PCR)
- Port Data Registers
- Individual Port Functionality.

### General Port Architecture

The general architecture of the I/O Port is shown in Figure 27., page 53. Individual Port architectures are shown in Figure 29., page 60 to Figure 32., page 63. In general, once the purpose for a port pin has been defined, that pin will no longer be available for other purposes. Exceptions will be noted.

As shown in Figure 27., page 53, the ports contain an output multiplexer whose selects 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 from CPLD.

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

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 PSDabel file, 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 microcontroller. The PDB feedback path allows the microcontroller to check the contents of the registers.

Ports A, B, and C have embedded Input Macrocells (IMCs). The IMCs can be configured as latches, registers, or direct inputs to the PLDs. The latches and registers are clocked by the address strobe (AS/ALE) or a product term from the PLD AND array. The outputs from the IMCs drive the PLD input bus and can be read by the microcontroller. See the section entitled Input Macrocell, page 42.

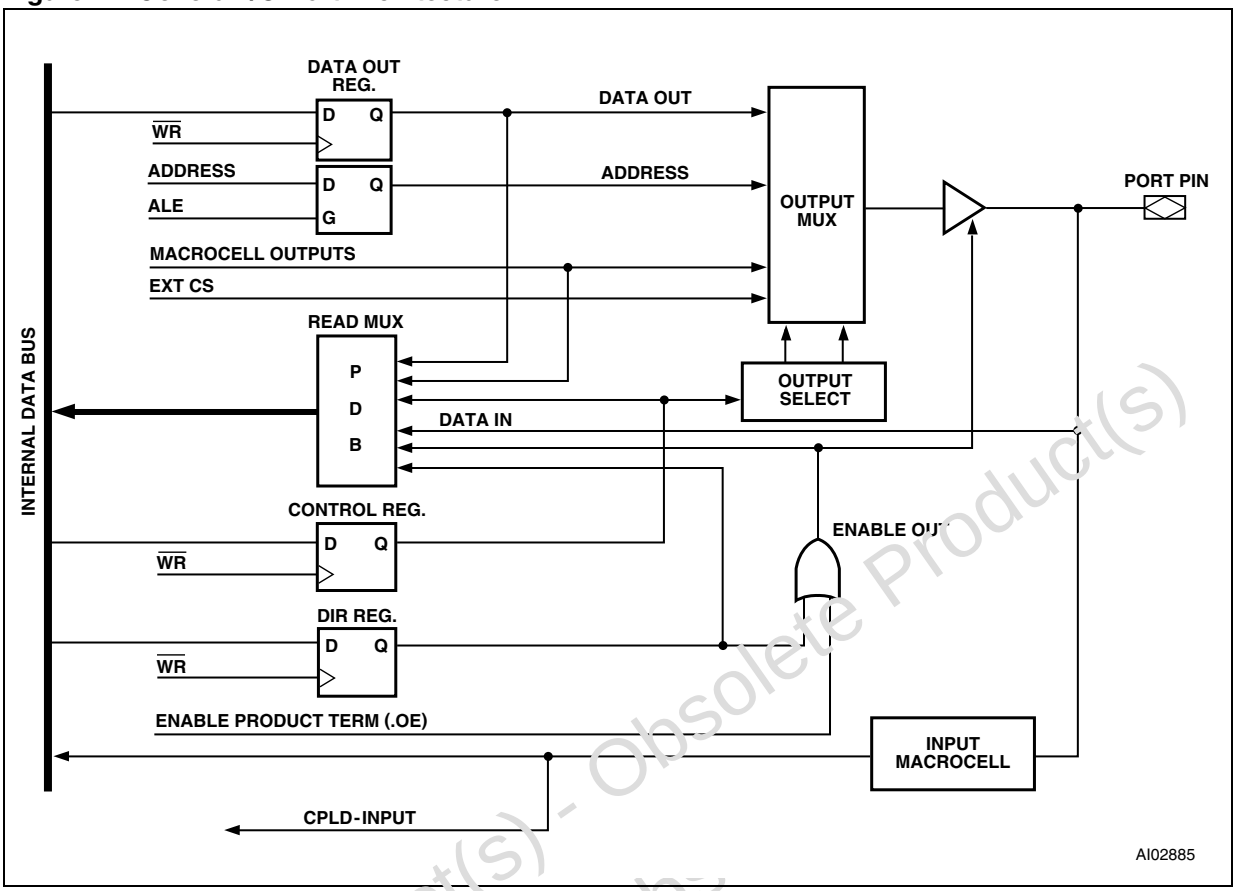
### Port Operating Modes

The I/O Ports have several modes of operation. Some modes can be defined using PSDabel, some by the microcontroller writing to the Control Registers in CSIOP space, and some by both. The modes that can only be defined using PSDsoft Express must be programmed into the device and cannot be changed unless the device is reprogrammed. The modes that can be changed by the microcontroller 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 microcontroller at run-time.

Table 19., page 54 summarizes which modes are available on each port. Table 22., page 57 shows how and where the different modes are configured. Each of the port operating modes are described in the following subsections.

Obsolete Product(s) - Obsolete Product(s)

**Figure 27. General I/O Port Architecture**




Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obs

### MCU I/O Mode

In the MCU I/O Mode, the microcontroller uses the PSD ports to expand its own I/O ports. By setting up the CSIOP space, the ports on the PSD are mapped into the microcontroller address space. The addresses of the ports are listed in Table 6., page 17.

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 the section entitled Peripheral I/O Mode, page 56. When the pin is configured as an output, the content of the Data Out Register drives the pin. When configured as an input, the microcontroller can read the port input through the Data In buffer. See Figure 27., page 53.

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 equation are written for them in PSDabel.

### PLD I/O Mode

The PLD I/O Mode uses a port as an input to the CPLD's Input Macrocells, and/or as an output from the CPLD's Output Macrocells. 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 setting 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 as a PLD input pin in PSDabel. The PLD I/O Mode is specified in PSDabel by declaring the port pins, and then writing an equation assigning the PLD I/O to a port.

### Address Out Mode

For microcontrollers with a multiplexed address/data bus, Address Out Mode can be used to drive latched addresses onto 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 21., page 55 for the address output pin assignments on Ports A and B for various MCUs.

For non-multiplexed 8 bit bus mode, address lines A7-A0 are available to Port B in Address Out Mode.

**Note:** do not drive address lines with Address Out Mode to an external memory device if it is intended for the MCU to boot from the external device. The MCU must first boot from PSD memory so the Direction and Control register bits can be set.

**Table 19. Port Operating Modes**

| Port Mode | Port A | Port B | Port C | Port D |
|---|---|---|---|---|
| MCU I/O | Yes | Yes | Yes | Yes |
| PLD I/O<br>McellAB Outputs<br>McellBC Outputs<br>Additional Ext. CS Outputs<br>PLD Inputs | <br>Yes<br>No<br>No<br>Yes | <br>Yes<br>Yes<br>No<br>Yes | <br>No<br>Yes<br>No<br>Yes | <br>No<br>No<br>Yes<br>Yes |
| Address Out | Yes (A7-A0 | Yes (A7-A0)<br>or (A15-A8) | No | No |
| Address In | Yes | Yes | Yes | Yes |
| Data Port | Yes (D7-D0) | No | No | No |
| Peripheral I/O | Yes | No | No | No |
| JTAG ISP | No | No | Yes[1] | No |

Note: 1. Can be multiplexed with other I/O functions.

Obsolete Product(s) - Obsolete Product(s)

**Table 20. Port Operating Mode Settings**

| Mode | Defined in PSDabel | Defined in PSD Configuration | Control Register Setting | Direction Register Setting | VM Register Setting | JTAG Enable |
|---|---|---|---|---|---|---|
| MCU I/O | Declare pins only | N/A[1] | 0 | 1 = output, 0 = input (Note [2]) | N/A | N/A |
| PLD I/O | Logic equations | N/A | N/A | (Note [2]) | N/A | N/A |
| Data Port (Port A) | N/A | Specify bus type | N/A | N/A | N/A | N/A |
| Address Out (Port A,B) | Declare pins only | N/A | 1 | 1 (Note [2]) | N/A | N/A |
| Address In (Port A,B,C,D) | Logic equation for Input Macrocells | N/A | N/A | N/A | N/A | N/A |
| Peripheral I/O (Port A) | Logic equations (PSEL0 & 1) | N/A | N/A | N/A | PIO bit = 1 | N/A |
| JTAG ISP (Note [3]) | JTAGSEL | JTAG Configuration | N/A | N/A | N/A | JTAG_Enable |

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.
3. Any of these three methods enables the JTAG pins on Port C.

**Table 21. I/O Port Latched Address Output Assignments**

| MCU | Port A (PA3-PA0) | Port A (PA7-PA4) | Port B (PB3-PB0) | Port B (PB7-PB4) |
|---|---|---|---|---|
| 8051XA (8-Bit) | N/A[1] | Address a7-a4 | Address A11-A8 | N/A |
| 80C251 (Page Mode) | N/A | N/A | Address A11-A8 | Address A15-A12 |
| All Other 8-Bit Multiplexed | Address A3-A0 | Address A7-A4 | Address A3-A0 | Address A7-A4 |
| 8-Bit Non-Multiplexed Bus | N/A | N/A | Address A3-A0 | Address A7-A4 |

Note: 1. N/A = Not Applicable.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

### Address In Mode

For microcontrollers that have more than 16 address lines, the higher addresses can be connected to Port A, B, C, and D. The address input can be latched in the Input Macrocell by the address strobe (ALE/AS). Any input that is included in the DPLD equations for the PLD's Flash, EEPROM, or SRAM is considered to be an address input.

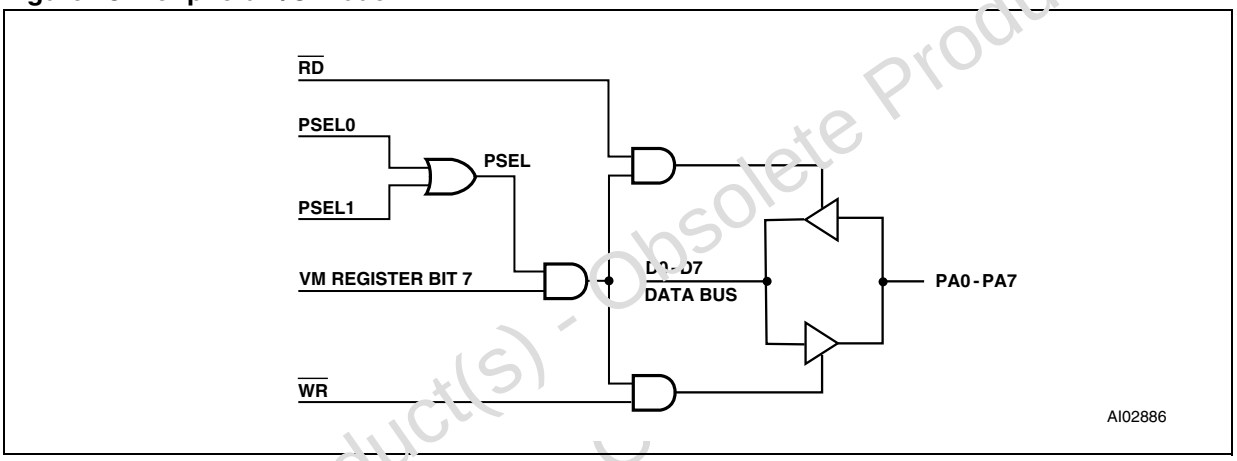
### Data Port Mode

Port A can be used as a data bus port for a microcontroller with a non-multiplexed address/data bus. The Data Port is connected to the data bus of the microcontroller. The general I/O functions are disabled in Port A if the port is configured as a Data Port.

### 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-stateable, bi-directional data buffer for the microcontroller. Peripheral I/O Mode is enabled by setting Bit 7 of the VM Register to a '1.' Figure 28 shows how Port A acts as a bi-directional buffer for the microcontroller data bus if Peripheral I/O Mode is enabled. An equation for PSEL0 and/or PSEL1 must be written in PSDabel. The buffer is tri-stated when PSEL 0 or PSEL1 is not active.

**Figure 28. Peripheral I/O Mode**




Obsolete Product(s) - Obsolete Product(s)

### JTAG In-System Programming (ISP)

Port C is JTAG compliant, and can be used for In-System Programming (ISP). You can multiplex JTAG operations with other functions on Port C because ISP is not performed during normal system operation. For more information on the JTAG Port, see the section entitled PROGRAMMING IN-CIRCUIT USING THE JTAG SERIAL INTERFACE, page 71.

### 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 6., page 17. The addresses in Table 6., page 17 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 22, are used for setting the Port configurations. The default Power-up state for each register in Table 22 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.

**Table 22. Port Configuration Registers (PCR)**

| Register Name | Port | MCU Access |
|---|---|---|
| Control | A,B | WRITE/READ |
| Direction | A,B,C,D | WRITE/READ |
| Drive Select[1] | A,B,C,D | WRITE/READ |

Note: 1. See Table 26., page 58 for Drive Register bit definition.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

### 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 will cause the corresponding pin to be an output, and any bit set to '0' will cause it to be an input. The default mode for all port pins is input.

Figure 29., page 60 and Figure 30., page 61 show 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 25. Since Port D only contains three pins (shown in Figure 32., page 63), the Direction Register for Port D has only the three least significant 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.

Aside: 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 26 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 23. Port Pin Direction Control, Output Enable P.T. Not Defined**

| Direction Register Bit | Port Pin Mode |
|---|---|
| 0 | Input |
| 1 | Output |

**Table 24. 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 25. 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 |

**Table 26. 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[1] | NA[1] | NA[1] | NA[1] | NA[1] | Slew Rate | Slew Rate | Slew Rate |

Note: 1. NA = Not Applicable.

Obsolete Product(s) - Obsolete Product(s)

### Port Data Registers

The Port Data Registers, shown in Table 27, are used by the MCU to write data to or read data from the ports. Table 27 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 microcontroller's address space. The microcontroller can read the output of the OMCs. If the Mask Macrocell Register bits are not set, writing to the Macrocell loads data to the Macrocell flip flops. See the section entitled PLD'S, page 34.

### Mask Macrocell Register

Each Mask Register bit corresponds to an OMC flip flop. When the Mask Register bit is set to a "1", loading data into the OMC flip flop is blocked. The default value is "0" or unblocked.

### Input Macrocells (IMC)

The IMCs can be used to latch or store external inputs. The outputs of the IMCs are routed to the PLD input bus, and can be read by the microcontroller. Refer to the section entitled PLD'S, page 34 for a detailed description.

**Table 27. 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<br>WRITE – loading macrocell 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 |

Obsolete Product(s) - Obsolete Product(s)

### Enable Out

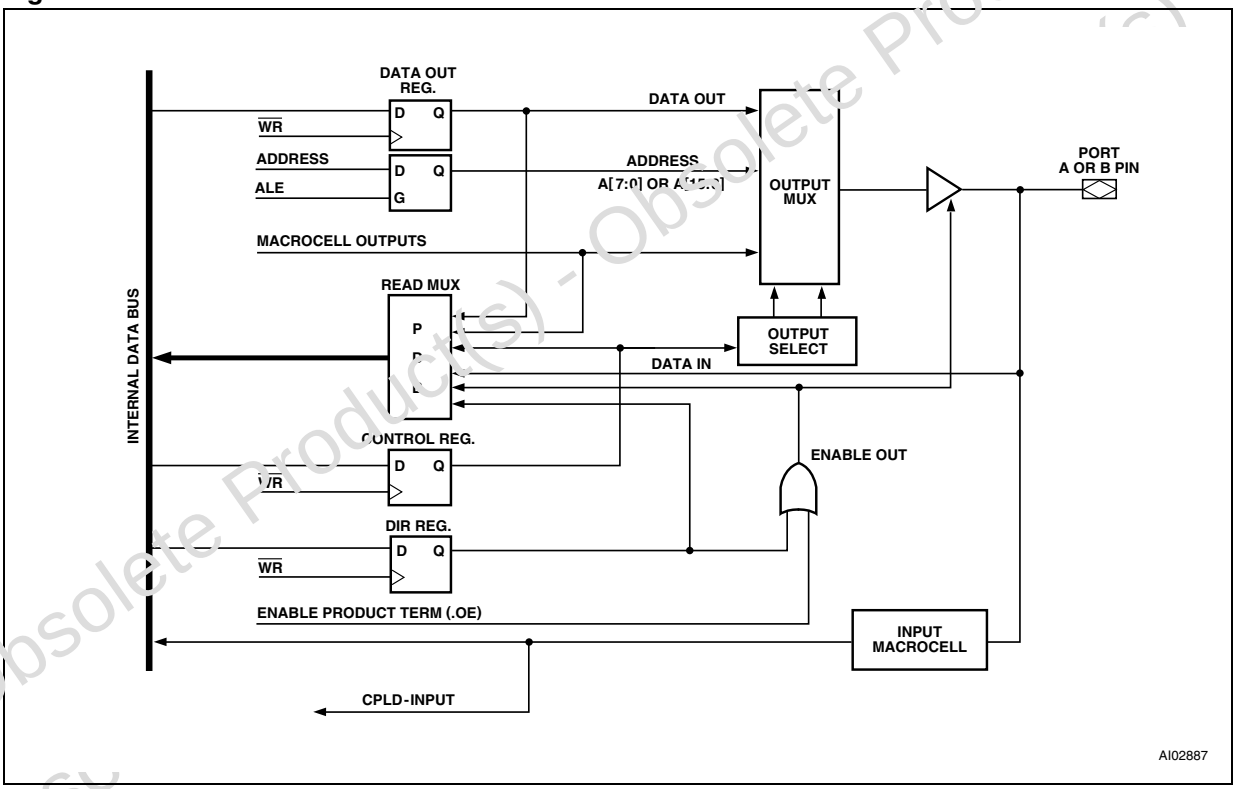
The Enable Out register can be read by the microcontroller. It contains the output enable values for a given port. A '1' indicates the driver is in output mode. A '0' indicates the driver is in tri-state and the pin is in input mode.

### Ports A and B – Functionality and Structure

Ports A and B have similar functionality and structure, as shown in Figure 29. 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 21., page 55.
- Address In – Additional high address inputs using the Input Macrocells (IMC).
- 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.
- Data Port – Port A to D7-D0 for 8 bit non-multiplexed bus
- Multiplexed Address/Data port for certain types of MCU bus interfaces.
- Peripheral Mode – Port A only

**Figure 29. Port A and Port B Structure**




Obsolete Product(s) - Obsolete Product(s)

### Port C – Functionality and Structure

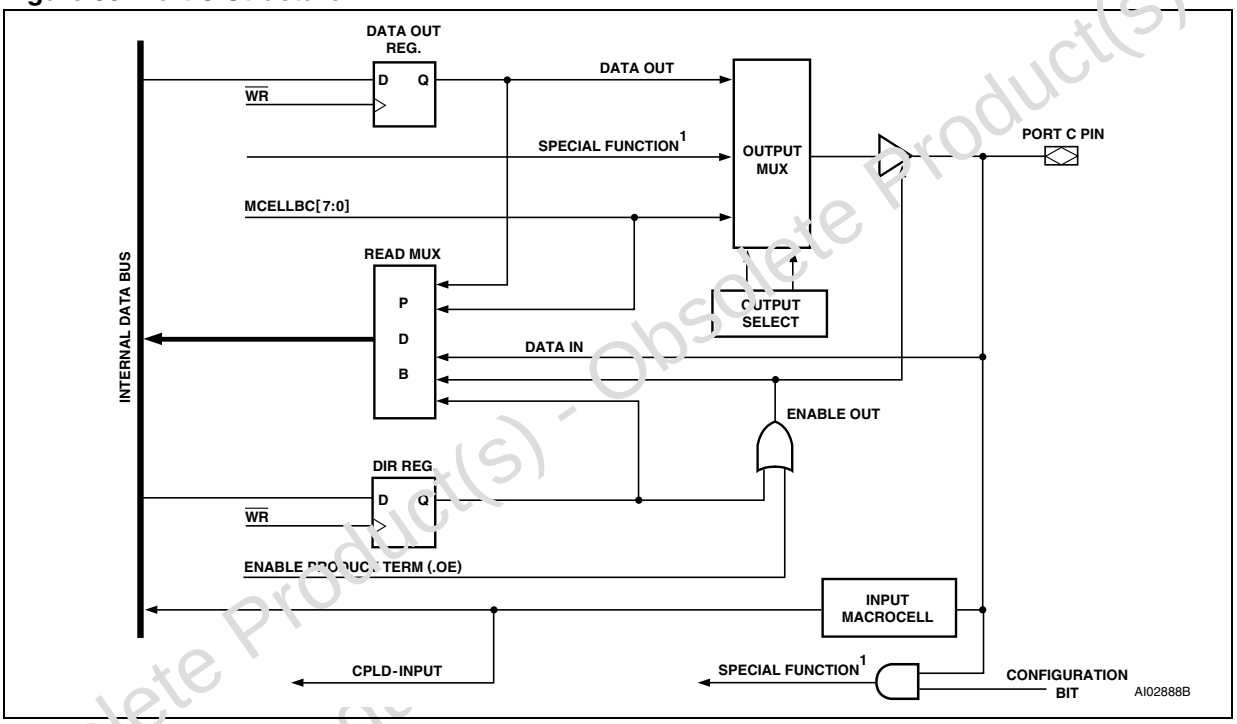
Port C can be configured to perform one or more of the following functions (see Figure 30):

- MCU I/O Mode
- CPLD Output – McellBC7-McellBC0 outputs can be connected to Port B or Port C.
- CPLD Input – via the Input Macrocells (IMC)
- Address In – Additional high address inputs using the Input Macrocells (IMC).
- In-System Programming (ISP) – JTAG port can be enabled for programming/erase of the PSD device. (See the section entitled PROGRAMMING IN-CIRCUIT USING THE JTAG SERIAL INTERFACE, page 71, for more information on JTAG programming.)
- Open Drain – Port C pins can be configured in Open Drain Mode

Port C does not support Address Out mode, and therefore no Control Register is required.

Pin PC7 may be configured as the DBE input in certain MCU interfaces.

**Figure 30. Port C Structure**




Note: 1. ISP.

Obsolete Product(s) - Obsolete Product(s)

### Port D – Functionality and Structure

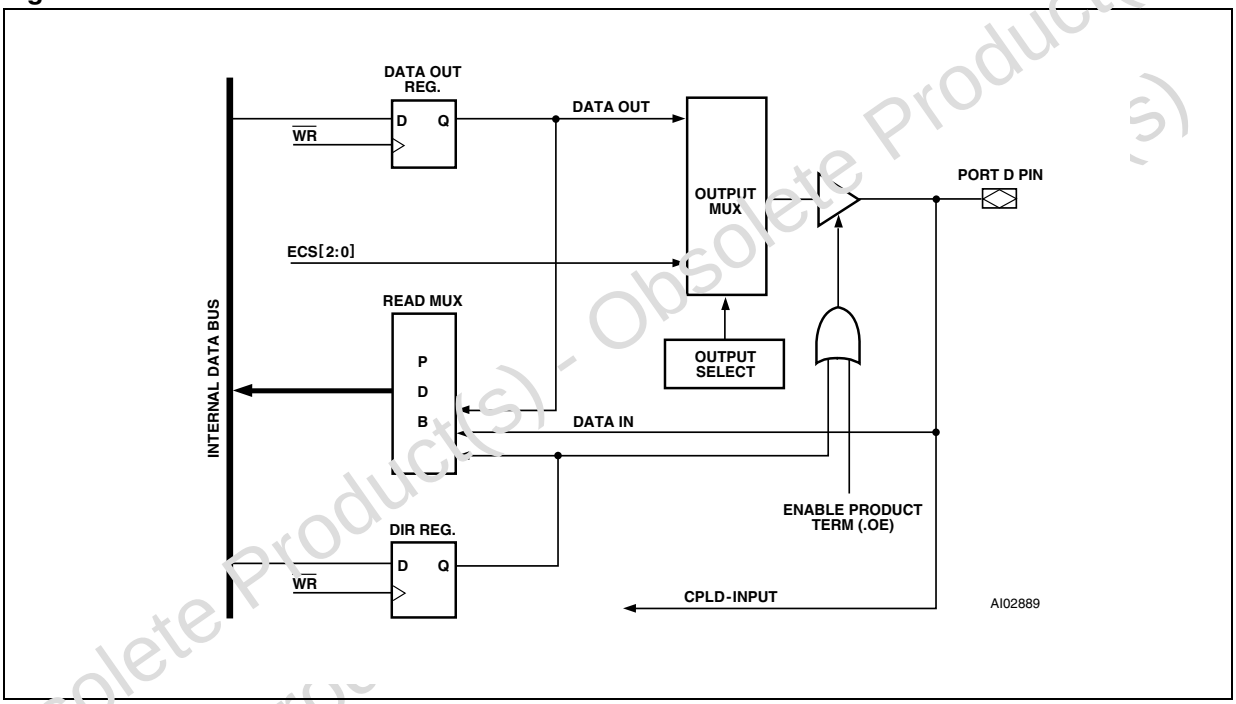
Port D has three I/O pins. See Figure 31 and Figure 32., page 63. This port does not support Address Out mode, and therefore no Control Register is required. Port D can be configured to perform one or more of the following functions:

- MCU I/O Mode
- CPLD Output – External Chip Select (ECS0-ECS2)
- CPLD Input – direct input to the CPLD, no Input Macrocells (IMC)
- 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:

- PD0 – ALE, as address strobe input
- PD1 – CLKIN, as clock input to the macrocells flip-flops and APD counter
- PD2 – $\overline{CSI}$, as active Low chip select input. A High input will disable the Flash memory, EEPROM, SRAM and CSIOP.

**Figure 31. Port D Structure**

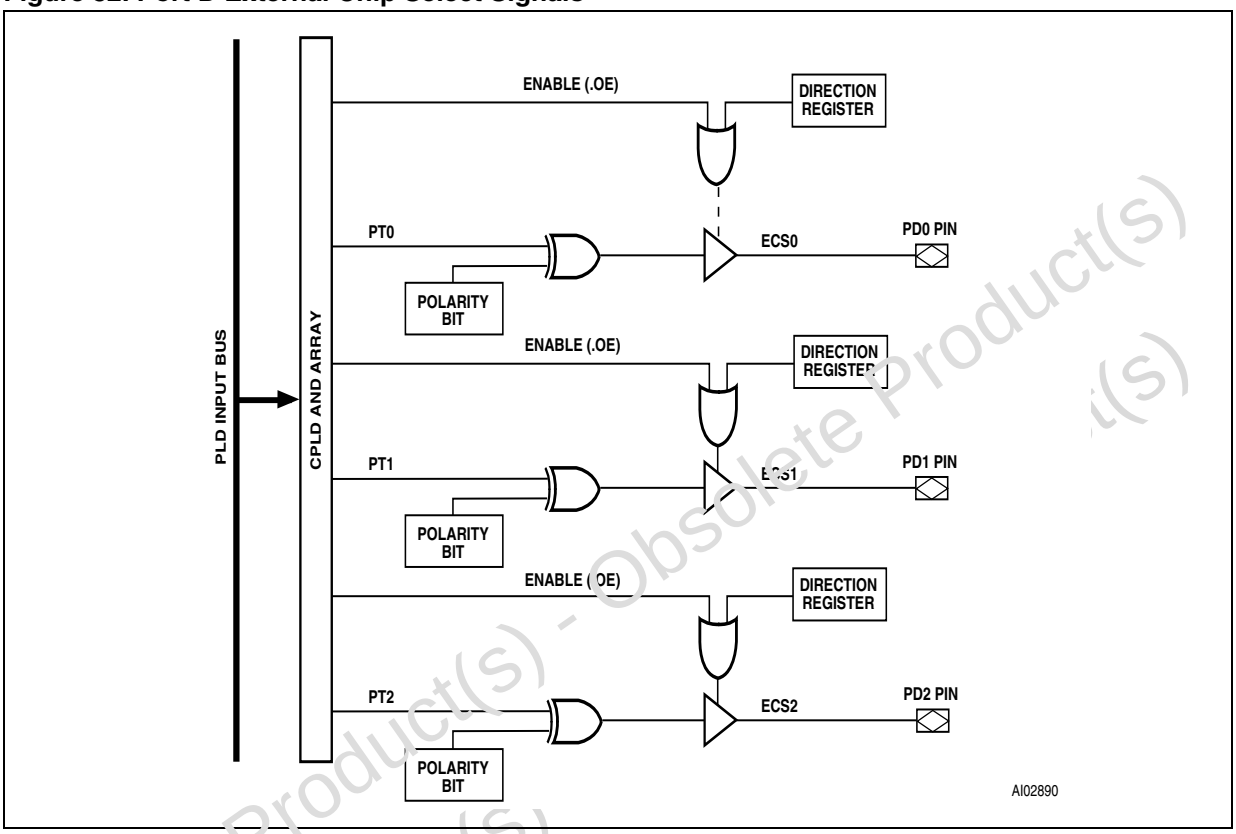



Obsolete Product(s) - Obsolete Product(s)

### External Chip Select

The CPLD also provides three External Chip Select (ECS0-ECS2) outputs on Port D pins that can be used to select external devices. Each External Chip Select (ECS0-ECS2) consists of one product term that can be configured active High or Low. The output enable of the pin is controlled by either the output enable product term or the Direction Register. (See Figure 32.)

**Figure 32. Port D External Chip Select Signals**




Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s)

## POWER MANAGEMENT

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

– All memory types in a PSD (Flash, EEPROM, and SRAM) are built with Zero-Power technology. In addition to using special silicon design methodology, Zero-Power 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 section entitled PLD Power Management, page 66.

– Like the Zero-Power feature, the Automatic Power Down (APD) logic allows the PSD to reduce to standby current automatically. The APD will block MCU address/data signals from reaching the memories and PLDs. This feature is available on all PSD devices. The APD Unit is described in more detail in the sections entitled Automatic Power-down (APD) Unit and Power-down Mode, page 65.

  Built in logic will monitor the address strobe of the MCU for activity. If there is no activity for a certain time period (MCU is asleep), the APD logic initiates Power Down Mode (if enabled). Once in Power Down Mode, all address/data signals are blocked from reaching PSD memories and PLDs, and the memories are deselected internally. This allows the memories and PLDs to remain in standby mode even if the address/data lines 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 standby mode, but not the memories.

– The PSD Chip Select Input ($\overline{CSI}$) on all families 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 logic, especially if your MCU has a chip select output. There is a slight penalty in memory access time when the $\overline{CSI}$ signal makes its initial transition from deselected to selected.

– The PMMR registers can be written by the MCU at run-time to manage power. PSD 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 36., page 73 and Figure 37., page 73). Significant power savings can be achieved by blocking signals that are not used in DPLD or CPLD logic equations.

  The PSD has a Turbo Bit in the PMMR0 register. This bit can be set to disable the Turbo Mode feature (default is Turbo Mode on). While Turbo Mode is disabled, 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 enabled. When the Turbo Mode is enabled, there is a significant DC current component and the AC component is higher.

Obsolete Product(s) - Obsolete Product(s)

### Automatic Power-down (APD) Unit and Power-down Mode

The APD Unit, shown in Figure 33, puts the PSD into Power-down mode by monitoring the activity of Address Strobe (ALE/AS, PD0). If the APD Unit is enabled, as soon as activity on Address Strobe (ALE/AS, PD0) stops, a four bit counter starts counting. If Address Strobe (ALE/AS, PD0) remains inactive for fifteen clock periods of CLKIN (PD1), the Power-down (PDN) signal becomes active, and the PSD enters Power-down mode, as discussed next.

### Power-down Mode

By default, if you enable the PSD APD unit, Power Down Mode is automatically enabled. The device will enter Power Down Mode if the address strobe (ALE/AS) remains inactive for fifteen CLKIN (pin PD1) clock periods.

The following should be kept in mind when the PSD is in Power Down Mode:

– If the address strobe starts pulsing again, the PSD will return to normal operation. The PSD will also return to normal operation if either the CSI input returns low or the Reset input returns high.
– The MCU address/data bus is blocked from all memories 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 clock (CLKIN). Note that blocking CLKIN from the PLDs will not block CLKIN from the APD unit.
– All PSD memories enter Standby Mode and are drawing standby current. However, the PLDs and I/O ports 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 28 for Power Down Mode effects on PSD ports.
– Typical standby current are of the order of the microampere (see Table 29). These standby current values assume that there are no transitions on any PLD input.

**Table 28. Power-down Mode's Effect on Ports**

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

**Figure 33. APD Unit**

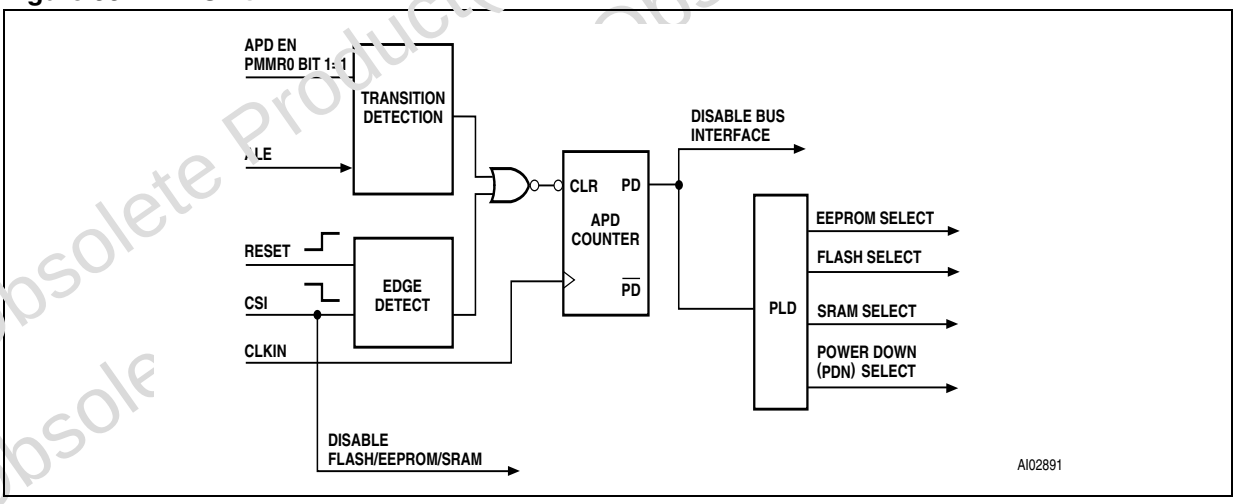



**Table 29. PSD Timing and Stand-by Current during Power-down Mode**

| Mode | PLD Propagation Delay | Memory Access Time | Access Recovery Time to Normal Access | Typical Stand-by Current | |
|---|---|---|---|---|---|
| | | | | 5V $V_{CC}$ | 3V $V_{CC}$ |
| Power-down | Normal $t_{PD}$ (1) | No Access | $t_{LVDV}$ | 50µA (2) | 25µA (2) |

Note: 1. Power-down does not affect the operation of the PLD. The PLD operation in this mode is based only on the Turbo bit.
2. Typical current consumption assuming no PLD inputs are changing state and the PLD Turbo bit is 0.

Obsolete Product(s) - Obsolete Product(s)

### For Users of the HC11 (or compatible)

The HC11 turns off its E clock when it sleeps. Therefore, if you are using an HC11 (or compatible) in your design, and you wish to use the Power-down mode, you must not connect the E clock to CLKIN (PD1). You should instead connect an independent clock signal to the CLKIN input (PD1). The clock frequency must be *less than* 15 times the frequency of AS. The reason for this is that if the frequency is greater than 15 times the frequency of AS, the PSD will keep going into Power-down mode.

### Other Power Saving Options

The PSD offers other reduced power saving options that are independent of the Power-down mode. Except for the Chip Select Input ($\overline{CSI}$, PD2) feature, they are enabled by setting bits in the PMMR0 and PMMR2 registers.

### PLD Power Management

The power and speed of the PLDs are controlled by the Turbo bit (bit 3) in the PMMR0. By setting the bit to '1', the Turbo mode is disabled and the PLDs consume Zero Power current when the inputs are not switching for an extended time of 70ns. The propagation delay time will be increased by 10ns after the Turbo bit is set to '1' (turned off) when the inputs change at a composite frequency of less than 15 MHz. When the Turbo bit is set to a '0' (turned on), the PLDs run at full power and speed. The Turbo bit affects the PLD's D.C. power, AC power, and propagation delay.

**Note:** Blocking MCU control signals with PMMR2 bits can further reduce PLD AC power consumption.

### PSD Chip Select Input ($\overline{CSI}$, PD2)

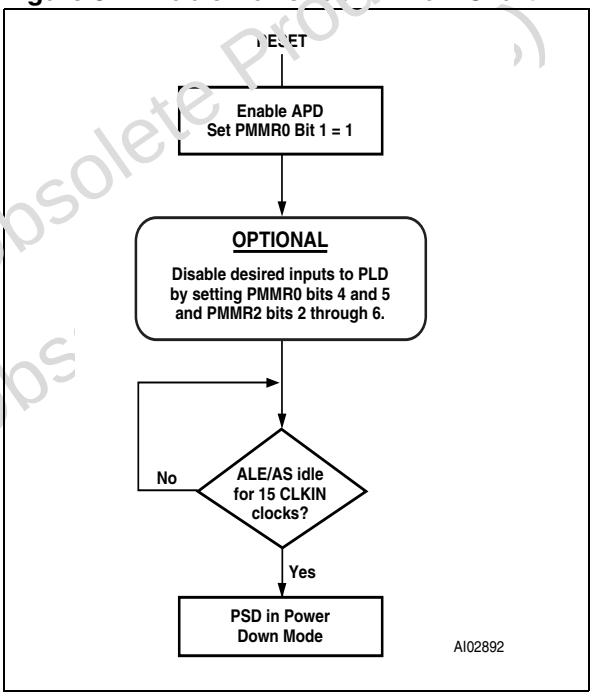
Pin PD2 of Port D can be configured in PSDsoft Express as the $\overline{CSI}$ input. When low, the signal selects and enables the internal Flash, EEPROM, SRAM, and I/O for READ or WRITE operations involving the PSD. A high on the $\overline{CSI}$ pin will disable the Flash memory, EEPROM, and SRAM, and reduce the PSD power consumption. However, the PLD and I/O pins remain operational when $\overline{CSI}$ is High.

**Note:** There may be a timing penalty when using the $\overline{CSI}$ pin depending on the speed grade of the PSD that you are using. See the timing parameter $t_{SLQV}$ in Table 63., page 95 or Table 64., page 95.

### Input Clock

The PSD provides the option to turn off the CLKIN input to the PLD to save AC power consumption. The CLKIN is an input to the PLD AND array and the Output Macrocells. During Power Down Mode, or, if the CLKIN input is not being used as part of the PLD logic equation, the clock should be disabled to save AC power. The CLKIN will be disconnected from the PLD AND array or the Macrocells by setting bits 4 or 5 to a '1' in PMMR0.

**Figure 34. Enable Power-down Flow Chart**




Obsolete Product(s) - Obsolete Product(s)

**Table 30. Power Management Mode Registers PMMR0 (Note 1)**

| | | | |
|---|---|---|---|
| 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. |
| 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. |

Note: 1. The bits of this register are cleared to zero following Power-up. Subsequent Reset ($\overline{RESET}$) pulses do not clear the registers.

**Table 31. Power Management Mode Registers PMMR2 (Note 1)**

| | | | |
|---|---|---|---|
| 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 CNTL0 | 0 = on | Cntl0 input to the PLD AND Array is connected. |
| | | 1 = off | Cntl0 input to PLD AND Array is disconnected, saving power. |
| Bit 3 | PLD Array CNTL1 | 0 = on | Cntl1 input to the PLD AND Array is connected. |
| | | 1 = off | Cntl1 input to PLD AND Array is disconnected, saving power. |
| Bit 4 | PLD Array CNTL2 | 0 = on | Cntl2 input to the PLD AND Array is connected. |
| | | 1 = off | Cntl2 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 | PLD Array DBE | 0 = on | DBE input to the PLD AND Array is connected. |
| | | 1 = off | DBE input to PLD AND Array is disconnected, saving power. |
| Bit 7 | X | 0 | Not used, and should be set to zero. |

Note: 1. The bits of this register are cleared to zero following Power-up. Subsequent Reset ($\overline{RESET}$) pulses do not clear the registers.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

### Input Control Signals

The PSD provides the option to turn off the input control signals (CNTL0, CNTL1, CNTL2, ALE, and DBE) 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 will be disconnected from the PLD AND array by setting bits 2, 3, 4, 5, and 6 to a '1' in the PMMR2.

**Table 32. APD Counter Operation**

| APD Enable Bit | ALE PD Polarity | ALE Level | APD Counter |
|---|---|---|---|
| 0 | X | X | Not Counting |
| 1 | X | Pulsing | Not Counting |
| 1 | 1 | 1 | Counting (Generates PDN after 15 Clocks) |
| 1 | 0 | 0 | Counting (Generates PDN after 15 Clocks) |

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

## RESET TIMING AND DEVICE STATUS AT RESET

### Power-On Reset

Upon Power-up, the PSD requires a Reset ($\overline{RESET}$) pulse of duration $t_{NLNH-PO}$ (See Tables 67 and 68 for values) after $V_{CC}$ is steady. During this period, the device loads internal configurations, clears some of the registers and sets the Flash memory or EEPROM into Operating mode. After the rising edge of Reset ($\overline{RESET}$), the PSD remains in the Reset mode for an additional period, $t_{OPR}$ (See Tables 67 and 68 for values), before the first memory access is allowed.

The PSD Flash or EEPROM memory is reset to the READ mode upon power up. The FSi and EESi select signals along with the write strobe signal must be in the false state during power-up reset for maximum security of the data contents and to remove the possibility of a byte being written on the first edge of a write strobe signal. The PSD automatically prevents write strobes from reaching the EEPROM memory array for about 5ms ($t_{EEH-WL}$). Any Flash memory WRITE cycle initiation is prevented automatically when $V_{CC}$ is below $V_{LKO}$.

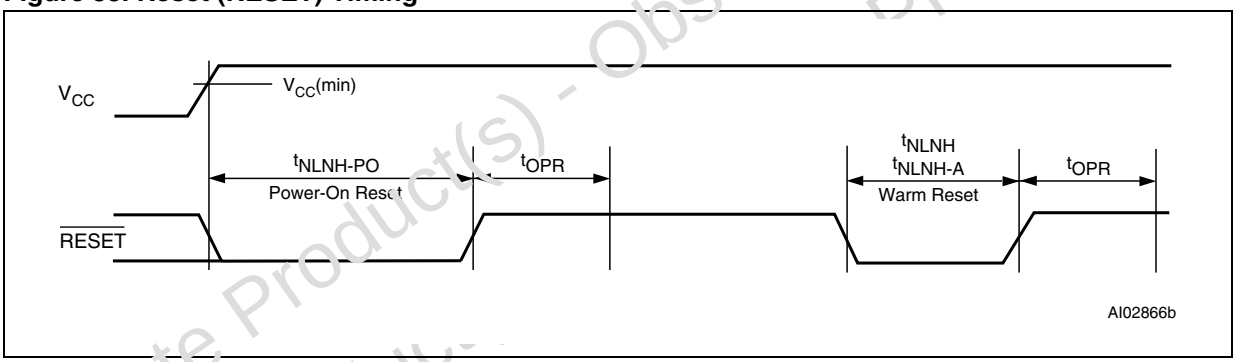
### Warm Reset

Once the device is up and running, the device can be reset with a much shorter pulse of $t_{NLNH}$ (See Tables 67 and 68 for values). The same $t_{OPR}$ time is needed before the device is operational after warm reset. Figure 35 shows the timing of the power on and warm reset.

### I/O Pin, Register and PLD Status at Reset

Table 33., page 70 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 PSD Configuration bits are loaded. This loading of PSD 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 PSDabel equations.

**Figure 35. Reset ($\overline{RESET}$) Timing**




Obsolete Product(s) - Obsolete Product(s)

**Table 33. 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 |
| Data Port | Tri-stated | Tri-stated | Tri-stated |
| 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[1] | Initialized, based on the selection in PSDsoft Express Configuration menu | Initialized, based on the selection in PSDsoft Express 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.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

## PROGRAMMING IN-CIRCUIT USING THE JTAG SERIAL INTERFACE

The JTAG interface on the PSD can be enabled on Port C (see Table 34., page 72). All memory (Flash and EEPROM), PLD logic, and PSD configuration bits may be programmed through the JTAG interface. A blank part 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 operations.

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

### Standard JTAG Signals

The standard JTAG signals (TMS, TCK, TDI, and TDO) can be enabled by any of three different conditions that are logically ORed. When enabled, TDI, TDO, TCK, and TMS are inputs, waiting for a serial command from an external JTAG controller device (such as FlashLink or Automated Test Equipment). When the enabling command is received from the external JTAG controller, TDO becomes an output and the JTAG channel is fully functional inside the PSD. The same command that enables the JTAG channel may optionally enable the two additional JTAG pins, TSTAT and TERR.

The following symbolic logic equation specifies the conditions enabling the four basic JTAG pins (TMS, TCK, TDI, and TDO) on their respective Port C pins. For purposes of discussion, the logic label JTAG_ON will be used. When JTAG_ON is true, the four pins are enabled for JTAG. When JTAG_ON is false, the four pins can be used for general PSD I/O.

```
JTAG_ON = PSDsoft_enabled +
/* An NVM configuration bit inside the PSD
    is set by the designer in the PSDsoft
    Express Configuration utility. This
    dedicates the pins for JTAG at all
    times (compliant with IEEE 1149.1) */
Microcontroller_enabled +
/* The microcontroller can set a bit at run-
    time by writing to the PSD register,
    JTAG Enable. This register is located
    at address CSIOP + offset C7h. Setting
    the JTAG_ENABLE bit in this register
    will enable the pins for JTAG use. This
    bit is cleared by a PSD reset or the
    microcontroller. See Table
    35., page 72 for bit definition. */
PSD_product_term_enabled:
/* A dedicated product term (PT) inside the
    PSD can be used to enable the JTAG pins.
    This PT has the reserved name JTAGSEL.
    Once defined as a node in PSDabel, the
    designer can write an equation for
    JTAGSEL. This method is used when the
    Port C JTAG pins are multiplexed with
    other I/O signals. It is recommended to
    logically tie the node JTAGSEL to the
    JEN\ signal on the Flashlink cable when
    multiplexing JTAG signals. (AN1153)
```

The PSD supports JTAG In-System-Configuration (ISC) commands, but not Boundary Scan. A definition of these JTAG-ISC commands and sequences are defined in a supplemental document available from ST. ST's PSDsoft Express software tool and FlashLink JTAG programming cable implement these JTAG-ISC commands. This document is needed only as a reference for designers who use a FlashLink to program their PSD.

Obsolete Product(s) - Obsolete Product(s)

### JTAG Extensions

TSTAT and TERR are two JTAG extension signals enabled by an “ISC_ENABLE” command received over the four standard JTAG pins (TMS, TCK, TDI, and TDO). They are used to speed programming and erase functions by indicating status on PSD pins instead of having to scan the status out serially using the standard JTAG channel.

TERR will indicate if an error has occurred when erasing a sector or programming a byte in Flash memory. This signal will go Low (active) when an error condition occurs, and stay Low until an “ISC_CLEAR” command is executed or a chip reset pulse is received after an “ISC-DISABLE” command. TERR does not apply to EEPROM.

TSTAT behaves the same as the Ready/Busy signal described in the section entitled Ready/Busy Pin (PC3), page 18. TSTAT will be High when the PSD device is in READ mode (Flash memory and EEPROM contents can be read). TSTAT will be Low when Flash memory programming or erase cycles are in progress, and also when data is being written to EEPROM.

TSTAT and TERR can be configured as open-drain type signals during an “ISC_ENABLE” command. This facilitates a wired-OR connection of TSTAT signals from several PSD devices and a wired-OR connection of TERR signals from those same devices. This is useful when several PSD devices are “chained” together in a JTAG environment.

### Security, Flash memory and EEPROM 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/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 Flash Memory and EEPROM sectors can individually be sector protected against erasures. The sector protect bits can be set in PSDsoft Express Configuration.

**Table 34. JTAG Port Signals**

| Port C Pin | JTAG Signals | Description |
|---|---|---|
| PC0 | TMS | Mode Select |
| PC1 | TCK | Clock |
| PC3 | TSTAT | Status |
| PC4 | TERR | Error Flag |
| PC5 | TDI | Serial Data In |
| PC6 | TDO | Serial Data Out |

## INITIAL DELIVERY STATE

When delivered from ST, the PSD device has all bits in the memory and PLDs set to '1.' The PSD Configuration Register bits are set to '0.' 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.

**Table 35. JTAG Enable Register**

| | | | |
|---|---|---|---|
| Bit 0 | JTAG_Enable | 0 = off | JTAG port is disabled. |
| | | 1 = on | JTAG port is enabled. |
| Bit 1 | X | 0 | Not used, and should be set to zero. |
| Bit 2 | X | 0 | Not used, and should be set to zero. |
| Bit 3 | X | 0 | Not used, and should be set to zero. |
| Bit 4 | X | 0 | Not used, and should be set to zero. |
| Bit 5 | X | 0 | Not used, and should be set to zero. |
| 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 state of Reset (RESET) does not interrupt (or prevent) JTAG operations if the JTAG signals are dedicated by an NVM Configuration bit (via PSDsoft Express). However, Reset (RESET) prevents or interrupts JTAG operations if the JTAG enable register is used to enable the JTAG signals.

Obsolete Product(s) - Obsolete Product(s)

## AC/DC PARAMETERS

The following tables describe the AD and DC parameters of the PSD:

- DC Electrical Specification
- AC Timing Specification

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

  MCU Timing
  - READ Timing
  - WRITE Timing
  - Peripheral Mode Timing
  - Power-down and Reset Timing

The following are issues concerning the parameters presented:

- In the DC specification the supply current is given for different modes of operation. Before calculating the total power consumption, determine the percentage of time that the PSD is in each mode. Also, the supply power is considerably different if the Turbo bit is '0.'
- The AC power component gives the PLD, EEPROM and SRAM mA/MHz specification. Figures 36 and 37 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 36. PLD $I_{CC}$ /Frequency Consumption (5V range)**

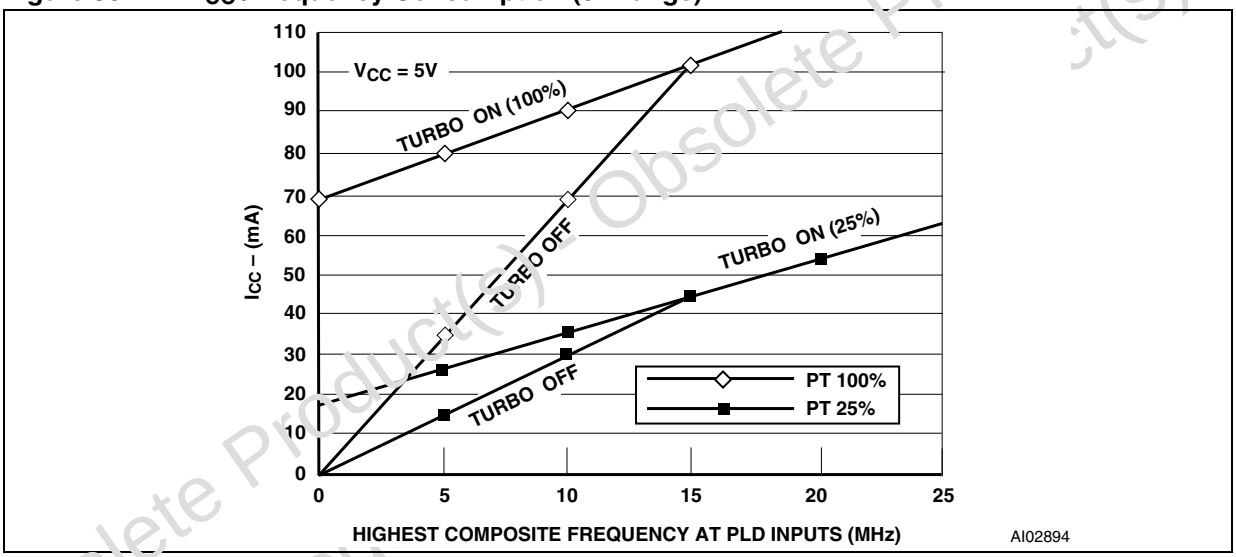
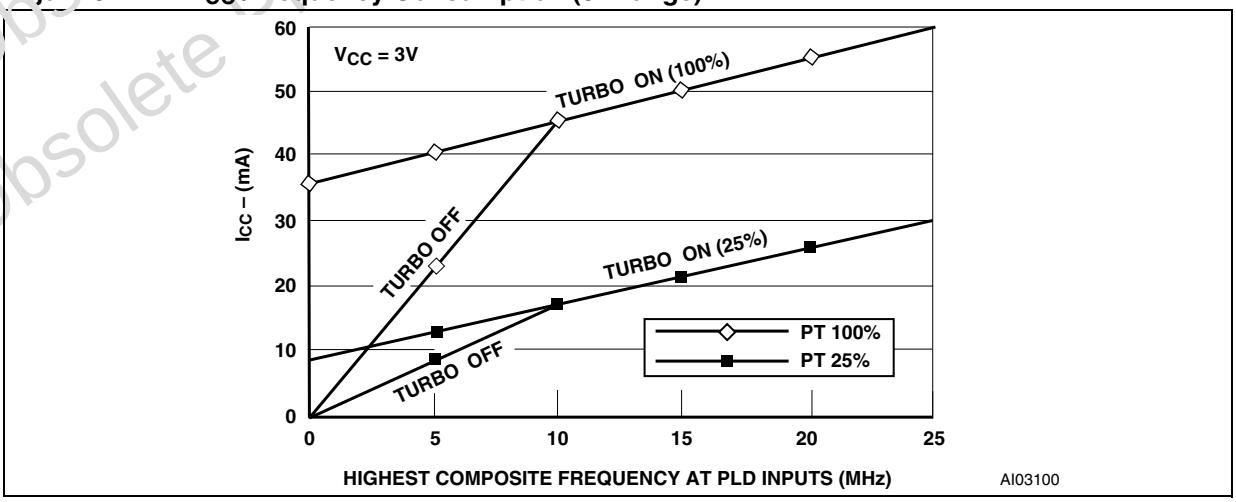



**Figure 37. PLD $I_{CC}$ /Frequency Consumption (3V range)**




Obsolete Product(s) - Obsolete Product(s)

**Table 36. Example of PSD Typical Power Calculation at $V_{CC}$ = 5.0V (Turbo Mode On)**

| Conditions | | | |
|---|---|---|---|
| Highest Composite PLD input frequency | | | |
| | (Freq PLD) | = 8MHz | |
| MCU ALE frequency (Freq ALE) | | = 4MHz | |
| | % Flash memory Access | = 80% | |
| | % SRAM access | = 15% | |
| | % I/O access | = 5% (no additional power above base) | |
| Operational Modes | | | |
| | % Normal | = 10% | |
| | % Power-down Mode | = 90% | |
| Number of product terms used | | | |
| | (from fitter report) | = 45 PT | |
| | % of total product terms | = 45/182 = 24.7% | |
| | Turbo Mode | = ON | |
| **Calculation (using typical values)** | | | |
| $I_{CC}$ total | | = Ipwrdown x %pwrdown + %normal x ($I_{CC}$ (ac) + $I_{CC}$ (dc)) | |
| | | = Ipwrdown x %pwrdown + % normal x (%flash x 2.5mA/MHz x Freq ALE | |
| | | | + %SRAM x 1.5mA/MHz x Freq ALE |
| | | | + % PLD x 2mA/MHz x Freq PLD |
| | | | + #PT x 400µA/PT) |
| | | = 50µA x 0.90 + 0.1 x (0.8 x 2.5mA/MHz x 4MHz | |
| | | | + 0.15 x 1.5mA/MHz x 4MHz |
| | | | + 2mA/MHz x 8MHz |
| | | | + 45 x 0.4mA/PT) |
| | | = 45µA + 0.1 x (8 + 0.9 + 16 + 18mA) | |
| | | = 45µA + 0.1 x 42.9 | |
| | | = 45µA + 4.29mA | |
| | | = 4.34mA | |
| This is the operating power with no EEPROM WRITE or Flash memory Erase cycles. Calculation is based on $I_{OUT}$ = 0mA. | | | |

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Table 37. Example of PSD Typical Power Calculation at $V_{CC}$ = 5.0V (Turbo Mode Off)**

| Conditions | | |
|---|---|---|
| Highest Composite PLD input frequency | | |
| | (Freq PLD) | = 8MHz |
| MCU ALE frequency (Freq ALE) | | = 4MHz |
| | % Flash memory Access | = 80% |
| | % SRAM access | = 15% |
| | % I/O access | = 5% (no additional power above base) |
| Operational Modes | | |
| | % Normal | = 10% |
| | % Power-down Mode | = 90% |
| 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)** | | |
| $I_{CC}$ total | | = Ipwrdown x %pwrdown + %normal x ($I_{CC}$ (ac) + $I_{CC}$ (dc)) |
| | | = Ipwrdown x %pwrdown + % normal x (%flash x 2.5mA/MHz x Freq ALE |
| | | + %SRAM x 1.5mA/MHz x Freq ALE |
| | | + % PLD x (from graph using Freq PLD)) |
| | | = 50µA x 0.90 + 0.1 x (0.8 x 2.5mA/MHz x 4MHz |
| | | + 0.15 x 1.5mA/MHz x 4MHz |
| | | + 24mA) |
| | | = 45µA + 0.1 x (8 + 0.9 + 24) |
| | | = 45µA + 0.1 x 32.9 |
| | | = 45µA + 3.29mA |
| | | = 3.34mA |
| This is the operating power with no EEPROM WRITE or Flash memory Erase cycles. Calculation is based on $I_{OUT}$ = 0mA. | | |

Obsolete Product(s) - Obsolete Product(s)

## 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 implied. 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 38. Absolute Maximum Ratings**

| Symbol | Parameter | Min. | Max. | Unit |
|---|---|---|---|---|
| $T_{STG}$ | Storage Temperature | –65 | 125 | °C |
| $T_{LEAD}$ | Lead Temperature during Soldering (20 seconds max.)[1] | | 235 | °C |
| $V_{IO}$ | Input and Output Voltage (Q = $V_{OH}$ or Hi-Z) | –0.6 | 7.0 | V |
| $V_{CC}$ | Supply Voltage | –0.6 | 7.0 | V |
| $V_{PP}$ | Device Programmer Supply Voltage | –0.6 | 14.0 | V |
| $V_{ESD}$ | Electrostatic Discharge Voltage (Human Body model) [2] | –2000 | 2000 | V |

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

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

## 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 Measurement 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 39. Operating Conditions (5V devices)**

| Symbol | Parameter | Min. | Max. | Unit |
|---|---|---|---|---|
| $V_{CC}$ | Supply Voltage | 4.5 | 5.5 | V |
| $T_A$ | Ambient Operating Temperature (Industrial) | –40 | 85 | °C |
| | Ambient Operating Temperature (Commercial) | 0 | 70 | °C |

**Table 40. Operating Conditions (3V devices)**

| Symbol | Parameter | Min. | Max. | Unit |
|---|---|---|---|---|
| $V_{CC}$ | Supply Voltage | 3.0 | 3.6 | V |
| $T_A$ | Ambient Operating Temperature (Industrial) | –40 | 85 | °C |
| | Ambient Operating Temperature (Commercial) | 0 | 70 | °C |

**Table 41. AC Signal Letters for PLD Timings**

| | |
|---|---|
| A | Address Input |
| C | CEout Output |
| D | Input Data |
| E | E Input |
| G | Internal WDOG_ON signal |
| I | Interrupt Input |
| L | ALE Input |
| N | Reset Input or Output |
| P | Port Signal Output |
| Q | Output Data |
| R | $\overline{WR}$, $\overline{UDS}$, $\overline{LDS}$, $\overline{DS}$, IORD, $\overline{PSEN}$ Inputs |
| S | Chip Select Input |
| T | R/$\overline{W}$ Input |
| W | Internal PDN Signal |
| M | Output Macrocell |

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

**Table 42. AC Signal Behavior Symbols for PLD Timings**

| | |
|---|---|
| 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 |

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

**Table 43. AC Measurement Conditions**

| Symbol | Parameter | Min. | Max. | Unit |
|---|---|---|---|---|
| $C_L$ | Load Capacitance | 30 | | pF |

Note: 1. Output Hi-Z is defined as the point where data out is no longer driven.

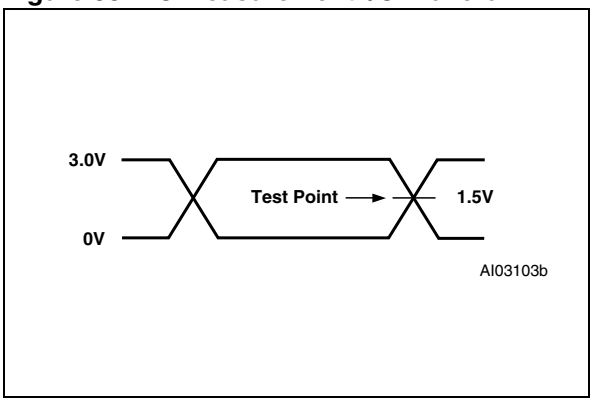
Obsolete Product(s) - Obsolete Product(s)

### Table 44. Capacitance

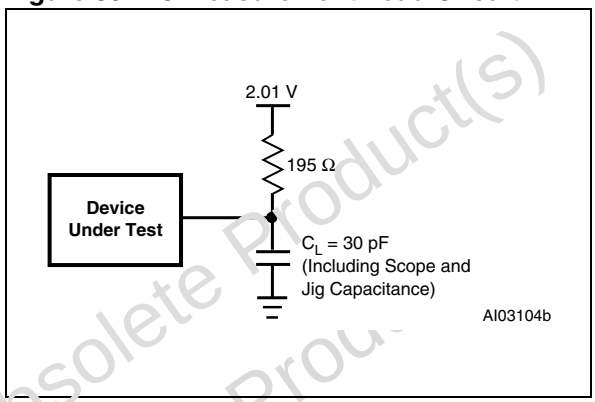
| Symbol | Parameter | Test Condition | Typ.[2] | Max. | Unit |
|---|---|---|---|---|---|
| $C_{IN}$ | Input Capacitance (for input pins) | $V_{IN}$ = 0V | 4 | 6 | pF |
| $C_{OUT}$ | Output Capacitance (for input/ output pins) | $V_{OUT}$ = 0V | 8 | 12 | pF |
| $C_{VPP}$ | Capacitance (for CNTL2/$V_{PP}$) | $V_{PP}$ = 0V | 18 | 25 | pF |

Note: 1. Sampled only, not 100% tested.
2. Typical values are for $T_A$ = 25°C and nominal supply voltages.

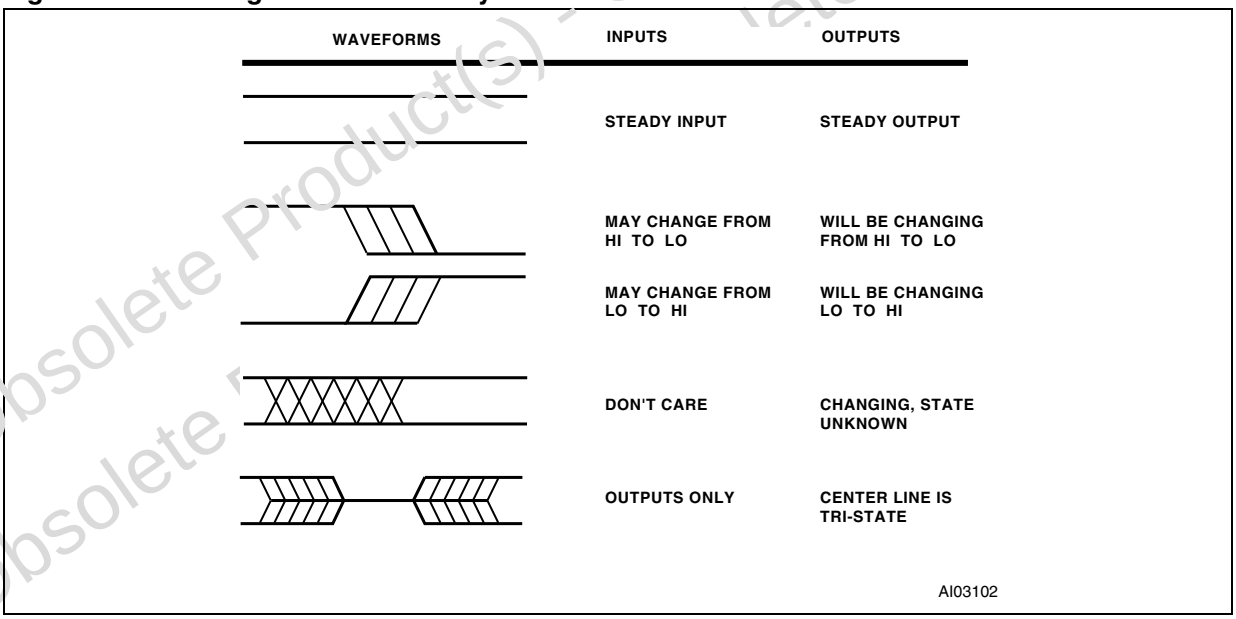
### Figure 38. AC Measurement I/O Waveform




### Figure 39. AC Measurement Load Circuit




### Figure 40. Switching Waveforms – Key




### Table 45. DC Characteristics (5V devices)

| Symbol | Parameter | Test Condition (in addition to those in Table 39) | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| $V_{IH}$ | Input High Voltage | 4.5V < $V_{CC}$ < 5.5V | 2 | | $V_{CC}$ +0.5 | V |
| $V_{IL}$ | Input Low Voltage | 4.5V < $V_{CC}$ < 5.5V | –0.5 | | 0.8 | V |

Obsolete Product(s) - Obsolete Product(s)

| Symbol | Parameter | | Test Condition (in addition to those in Table 39) | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|---|
| $V_{IH1}$ | Reset High Level Input Voltage | | (Note 1) | $0.8V_{CC}$ | | $V_{CC}$ +0.5 | V |
| $V_{IL1}$ | Reset Low Level Input Voltage | | (Note 1) | –0.5 | | $0.2V_{CC}$ –0.1 | V |
| $V_{HYS}$ | Reset Pin Hysteresis | | | 0.3 | | | V |
| $V_{LKO}$ | $V_{CC}$ (min) for Flash Erase and Program | | | 2.5 | | 4.2 | V |
| $V_{OL}$ | Output Low Voltage | | $I_{OL}$ = 20µA, $V_{CC}$ = 4.5V | | 0.01 | 0.1 | V |
| | | | $I_{OL}$ = 8mA, $V_{CC}$ = 4.5V | | 0.25 | 0.45 | V |
| $V_{OH}$ | Output High Voltage | | $I_{OH}$ = –20µA, $V_{CC}$ = 4.5V | 4.4 | 4.49 | | V |
| | | | $I_{OH}$ = –2mA, $V_{CC}$ = 4.5V | 2.4 | 3.9 | | V |
| $I_{SB}$ | Stand-by Supply Current for Power-down Mode | | $\overline{CSI}$ >$V_{CC}$ –0.3V (Notes 2,3) | | 50 | 200 | µA |
| $I_{LI}$ | Input Leakage Current | | $V_{SS}$ < $V_{IN}$ < $V_{CC}$ | –1 | ±0.1 | 1 | µA |
| $I_{LO}$ | Output Leakage Current | | 0.45 < $V_{OUT}$ < $V_{CC}$ | –10 | ±5 | 10 | µA |
| $I_{CC}$ (DC) (Note 5) | Operating Supply Current | ZPLD Only | ZPLD_TURBO = Off, f = 0MHz (Note 5) | | 0 | | mA |
| | | | ZPLD_TURBO = On, f = 0MHz | | 400 | 700 | µA/PT |
| | | Flash memory or EEPROM | During Flash memory or EEPROM WRITE/Erase Only | | 15 | 30 | mA |
| | | | Read only, f = 0MHz | | 0 | 0 | mA |
| | | SRAM | f = 0MHz | | 0 | 0 | mA |
| $I_{CC}$ (AC) (Note 5) | ZPLD AC Adder | | | See Figure 36, note 4 | | | |
| | Flash memory or EEPROM AC Adder | | | | 2.5 | 3.5 | mA/ MHz |
| | SRAM AC Adder | | | | 1.5 | 3.0 | mA/ MHz |

Note: 1. Reset ($\overline{RESET}$) has hysteresis. $V_{IL1}$ is valid at or below $0.2V_{CC}$ –0.1. $V_{IH1}$ is valid at or above $0.8V_{CC}$.
2. $\overline{CSI}$ deselected or internal Power-down mode is active.
3. PLD is in non-Turbo mode, and none of the inputs are switching.
4. Please see Figure 36., page 73 for the PLD current calculation.
5. $I_{OUT}$ = 0mA

### Table 46. DC Characteristics (3V devices)

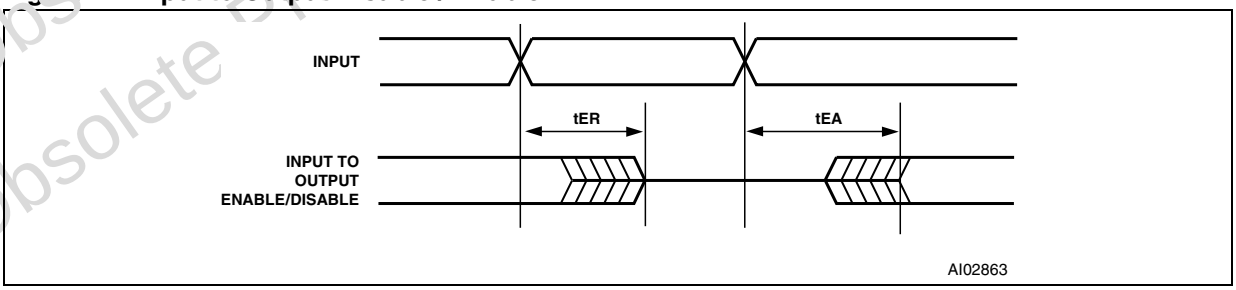
| Symbol | Parameter | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|
| $V_{IH}$ | High Level Input Voltage | 3.0V < $V_{CC}$ < 3.6V | $0.7V_{CC}$ | | $V_{CC}$ +0.5 | V |
| $V_{IL}$ | Low Level Input Voltage | 3.0V < $V_{CC}$ < 3.6V | –0.5 | | 0.8 | V |
| $V_{IH1}$ | Reset High Level Input Voltage | (Note 1) | $0.8V_{CC}$ | | $V_{CC}$ +0.5 | V |
| $V_{IL1}$ | Reset Low Level Input Voltage | (Note 1) | –0.5 | | $0.2V_{CC}$ –0.1 | V |

Obsolete Product(s) - Obsolete Product(s)

| Symbol | Parameter | | Conditions | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|---|
| $V_{HYS}$ | Reset Pin Hysteresis | | | 0.3 | | | V |
| $V_{LKO}$ | $V_{CC}$ (min) for Flash Erase and Program | | | 1.5 | | 2.2 | V |
| $V_{OL}$ | Output Low Voltage | | $I_{OL}$ = 20µA, $V_{CC}$ = 3.0V | | 0.01 | 0.1 | V |
| | | | $I_{OL}$ = 4mA, $V_{CC}$ = 3.0V | | 0.15 | 0.45 | V |
| $V_{OH}$ | Output High Voltage | | $I_{OH}$ = –20µA, $V_{CC}$ = 3.0V | 2.9 | 2.99 | | V |
| | | | $I_{OH}$ = –1mA, $V_{CC}$ = 3.0V | 2.7 | 2.8 | | V |
| $I_{SB}$ | Stand-by Supply Current for Power-down Mode | | $\overline{CSI}$ >$V_{CC}$ –0.3V (Notes 2) | | 25 | 100 | µA |
| $I_{LI}$ | Input Leakage Current | | $V_{SS}$ < $V_{IN}$ < $V_{CC}$ | –1 | ±0.1 | 1 | µA |
| $I_{LO}$ | Output Leakage Current | | 0.45 < $V_{IN}$ < $V_{CC}$ | –10 | ±5 | 10 | µA |
| $I_{CC}$ (DC) (Note 3) | Operating Supply Current | ZPLD Only | ZPLD_TURBO = Off, f = 0MHz (Note 3) | | 0 | | µA/PT |
| | | | ZPLD_TURBO = On, f = 0MHz | | 200 | 400 | µA/PT |
| | | Flash memory or EEPROM | During Flash memory or EEPROM WRITE/Erase Only | | 10 | 25 | mA |
| | | | Read only, f = 0MHz | | 0 | 0 | mA |
| | | SRAM | f = 0MHz | | 0 | 0 | mA |
| $I_{CC}$ (AC) (Note 3) | ZPLD AC Adder | | See Figure 37., page 73 | | | | |
| | Flash memory or EEPROM AC Adder | | | | 1.5 | 2.0 | mA/ MHz |
| | SRAM AC Adder | | | | 0.8 | 1.5 | mA/ MHz |

Note: 1. Reset ($\overline{RESET}$) has hysteresis. $V_{IL1}$ is valid at or below 0.2$V_{CC}$ –0.1. $V_{IH1}$ is valid at or above 0.8$V_{CC}$.
2. $\overline{CSI}$ deselected or internal PD is active.
3. $I_{OUT}$ = 0mA

**Figure 41. Input to Output Disable / Enable**




Obsolete Product(s) - Obsolete Product(s)

**Figure 42. Combinatorial Timing PLD**

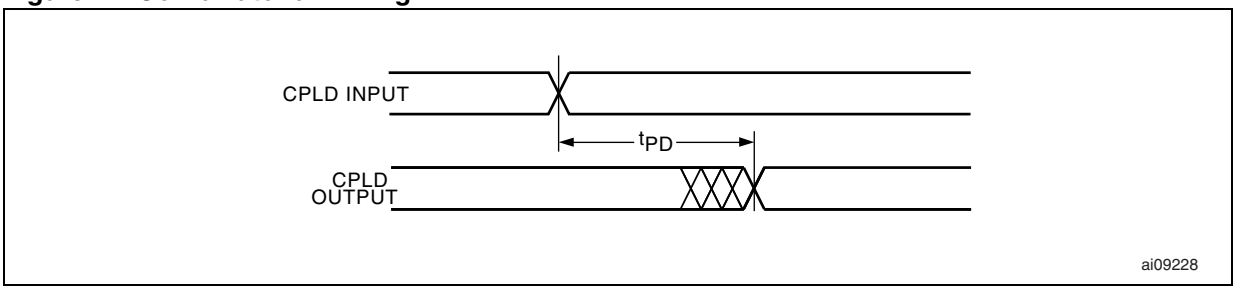



**Table 47. CPLD Combinatorial Timing (5V devices)**

| Symbol | Parameter | Conditions | -90 | | -12 | | -15 | | Fast PT Aloc | Turbo Off[2] | Slew rate[1] | Unit |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | Min | Max | | | | |
| $t_{PD}$ | CPLD Input Pin/ Feedback to CPLD Combinatorial Output | | | 25 | | 30 | | 32 | + 2 | + 10 | – 2 | ns |
| $t_{EA}$ | CPLD Input to CPLD Output Enable | | | 26 | | 30 | | 32 | | + 10 | – 2 | ns |
| $t_{ER}$ | CPLD Input to CPLD Output Disable | | | 26 | | 30 | | 32 | | + 10 | – 2 | ns |
| $t_{ARP}$ | CPLD Register Clear or Preset Delay | | | 26 | | 30 | | 33 | | + 10 | – 2 | ns |
| $t_{ARPW}$ | CPLD Register Clear or Preset Pulse Width | | 20 | | 24 | | 29 | | | + 10 | | ns |
| $t_{ARD}$ | CPLD Array Delay | Any macrocell | | 16 | | 18 | | 22 | + 2 | | | ns |

Note: 1. Fast Slew Rate output available on PA3-PA0, PB3-PB0, and PD2-PD0. Decrement times by given amount.
2. ZPSD versions only.

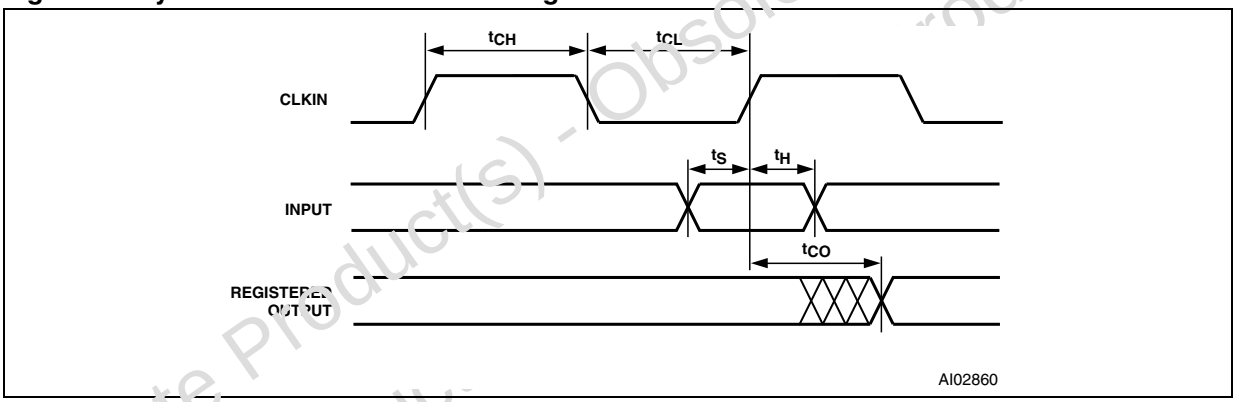
Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Table 48. CPLD Combinatorial Timing (3V devices)**

| Symbol | Parameter | Conditions | -15 | | -20 | | PT Aloc | Turbo Off[2] | Slew rate[1] | Unit |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | | | | |
| $t_{PD}$ | CPLD Input Pin/Feedback to CPLD Combinatorial Output | | | 48 | | 55 | + 4 | + 20 | – 6 | ns |
| $t_{EA}$ | CPLD Input to CPLD Output Enable | | | 43 | | 50 | | + 20 | – 6 | ns |
| $t_{ER}$ | CPLD Input to CPLD Output Disable | | | 43 | | 50 | | + 20 | – 6 | ns |
| $t_{ARP}$ | CPLD Register Clear or Preset Delay | | | 48 | | 55 | | + 20 | – 6 | ns |
| $t_{ARPW}$ | CPLD Register Clear or Preset Pulse Width | | 30 | | 35 | | | + 20 | | ns |
| $t_{ARD}$ | CPLD Array Delay | Any macrocell | | 29 | | 33 | + 4 | | | ns |

Note: 1. Fast Slew Rate output available on PA3-PA0, PB3-PB0, and PD2-PD0. Decrement times by given amount.
2. ZPSD versions only.

**Figure 43. Synchronous Clock Mode Timing – PLD**




Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s)

**Table 49. CPLD Macrocell Synchronous Clock Mode Timing (5V devices)**

| Symbol | Parameter | Conditions | -90 | | -12 | | -15 | | Fast PT Aloc | Turbo Off | Slew rate[1] | Unit |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | Min | Max | | | | |
| $f_{MAX}$ | Maximum Frequency External Feedback | $1/(t_S+t_{CO})$ | | 30.30 | | 26.3 | | 23.8 | | | | MHz |
| | Maximum Frequency Internal Feedback ($f_{CNT}$) | $1/(t_S+t_{CO}-10)$ | | 43.48 | | 35.7 | | 31.25 | | | | MHz |
| | Maximum Frequency Pipelined Data | $1/(t_{CH}+t_{CL})$ | | 50.00 | | 41.67 | | 33.3 | | | | MHz |
| $t_S$ | Input Setup Time | | 15 | | 18 | | 20 | | + 2 | + 10 | | ns |
| $t_H$ | Input Hold Time | | 0 | | 0 | | 0 | | | | | ns |
| $t_{CH}$ | Clock High Time | Clock Input | 10 | | 12 | | 15 | | | | | ns |
| $t_{CL}$ | Clock Low Time | Clock Input | 10 | | 12 | | 15 | | | | | ns |
| $t_{CO}$ | Clock to Output Delay | Clock Input | | 18 | | 20 | | 22 | | | – 2 | ns |
| $t_{ARD}$ | CPLD Array Delay | Any macrocell | | 16 | | 18 | | 22 | + 2 | | | ns |
| $t_{MIN}$ | Minimum Clock Period [2] | $t_{CH}+t_{CL}$ | 20 | | 24 | | 30 | | | | | ns |

Note: 1. Fast Slew Rate output available on PA3-PA0, PB3-PB0, and PD2-PD0. Decrement times by given amount.
2. CLKIN (PD1) $t_{CLCL} = t_{CH} + t_{CL}$.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s)

**Table 50. CPLD Macrocell Synchronous Clock Mode Timing (3V devices)**

| Symbol | Parameter | Conditions | -15 | | -20 | | PT Aloc | Turbo Off | Slew rate[1] | Unit |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | | | | |
| $f_{MAX}$ | Maximum Frequency External Feedback | $1/(t_S+t_{CO})$ | | 17.8 | | 14.7 | | | | MHz |
| | Maximum Frequency Internal Feedback ($f_{CNT}$) | $1/(t_S+t_{CO}-10)$ | | 19.6 | | 17.2 | | | | MHz |
| | Maximum Frequency Pipelined Data | $1/(t_{CH}+t_{CL})$ | | 33.3 | | 31.2 | | | | MHz |
| $t_S$ | Input Setup Time | | 27 | | 35 | | + 4 | + 20 | | ns |
| $t_H$ | Input Hold Time | | 0 | | 0 | | | | | ns |
| $t_{CH}$ | Clock High Time | Clock Input | 15 | | 16 | | | | | ns |
| $t_{CL}$ | Clock Low Time | Clock Input | 15 | | 16 | | | | | ns |
| $t_{CO}$ | Clock to Output Delay | Clock Input | | 35 | | 39 | | | – 6 | ns |
| $t_{ARD}$ | CPLD Array Delay | Any macrocell | | 29 | | 33 | + 4 | | | ns |
| $t_{MIN}$ | Minimum Clock Period[2] | $t_{CH}+t_{CL}$ | 29 | | 32 | | | | | ns |

Note: 1. Fast Slew Rate output available on PA3-PA0, PB3-PB0, and PD2-PD0.
2. CLKIN (PD1) $t_{CLCL} = t_{CH} + t_{CL}$.

**Figure 44. Asynchronous Reset / Preset**

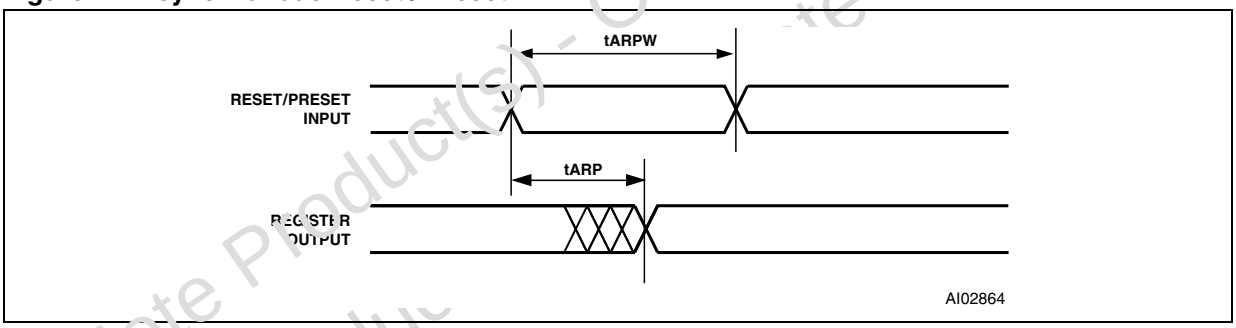
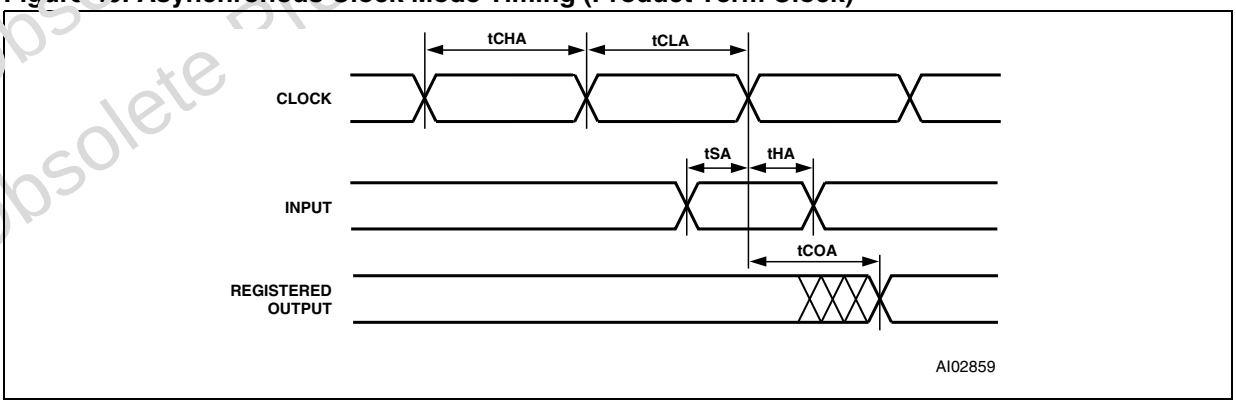



**Figure 45. Asynchronous Clock Mode Timing (Product Term Clock)**




Obsolete Product(s) - Obsolete Product(s)

**Table 51. CPLD Macrocell Asynchronous Clock Mode Timing (5V devices)**

| Symbol | Parameter | Conditions | -90 | | -12 | | -15 | | PT Aloc | Turbo Off[1] | Slew Rate | Unit |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | Min | Max | | | | |
| $f_{MAXA}$ | Maximum Frequency External Feedback | $1/(t_{SA}+t_{COA})$ | | 26.32 | | 23.25 | | 20.4 | | | | MHz |
| | Maximum Frequency Internal Feedback ($f_{CNTA}$) | $1/(t_{SA}+t_{COA}–10)$ | | 35.71 | | 30.30 | | 25.64 | | | | MHz |
| | Maximum Frequency Pipelined Data | $1/(t_{CHA}+t_{CLA})$ | | 41.67 | | 35.71 | | 33.3 | | | | MHz |
| $t_{SA}$ | Input Setup Time | | 8 | | 10 | | 12 | | + 2 | + 10 | | ns |
| $t_{HA}$ | Input Hold Time | | 12 | | 14 | | 14 | | | | | ns |
| $t_{CHA}$ | Clock Input High Time | | 12 | | 14 | | 15 | | | + 10 | | ns |
| $t_{CLA}$ | Clock Input Low Time | | 12 | | 14 | | 15 | | | + 10 | | ns |
| $t_{COA}$ | Clock to Output Delay | | | 30 | | 33 | | 37 | | + 10 | – 2 | ns |
| $t_{ARDA}$ | CPLD Array Delay | Any macrocell | | 16 | | 18 | | 22 | + 2 | | | ns |
| $t_{MINA}$ | Minimum Clock Period | $1/f_{CNTA}$ | 28 | | 33 | | 39 | | | | | ns |

Note: 1. ZPSD versions only.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Table 52. CPLD Macrocell Asynchronous Clock Mode Timing (3V devices)**

| Symbol | Parameter | Conditions | -15 | | -20 | | PT Aloc | Turbo Off[1] | Slew Rate | Unit |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | | | | |
| $f_{MAXA}$ | Maximum Frequency External Feedback | $1/(t_{SA}+t_{COA})$ | | 19.2 | | 16.9 | | | | MHz |
| | Maximum Frequency Internal Feedback ($f_{CNTA}$) | $1/(t_{SA}+t_{COA}–10)$ | | 23.8 | | 20.4 | | | | MHz |
| | Maximum Frequency Pipelined Data | $1/(t_{CHA}+t_{CLA})$ | | 27 | | 24.4 | | | | MHz |
| $t_{SA}$ | Input Setup Time | | 12 | | 13 | | + 4 | + 20 | | ns |
| $t_{HA}$ | Input Hold Time | | 15 | | 17 | | | | | ns |
| $t_{CHA}$ | Clock High Time | | 22 | | 25 | | | + 20 | | ns |
| $t_{CLA}$ | Clock Low Time | | 15 | | 16 | | | + 20 | | ns |
| $t_{COA}$ | Clock to Output Delay | | | 40 | | 45 | | + 20 | – 6 | ns |
| $t_{ARD}$ | CPLD Array Delay | Any macrocell | | 29 | | 33 | + 4 | | | ns |
| $t_{MINA}$ | Minimum Clock Period | $1/f_{CNTA}$ | 42 | | 45 | | | | | ns |

Note: 1. ZPSD Versions only.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Figure 46. Input Macrocell Timing (product term clock)**

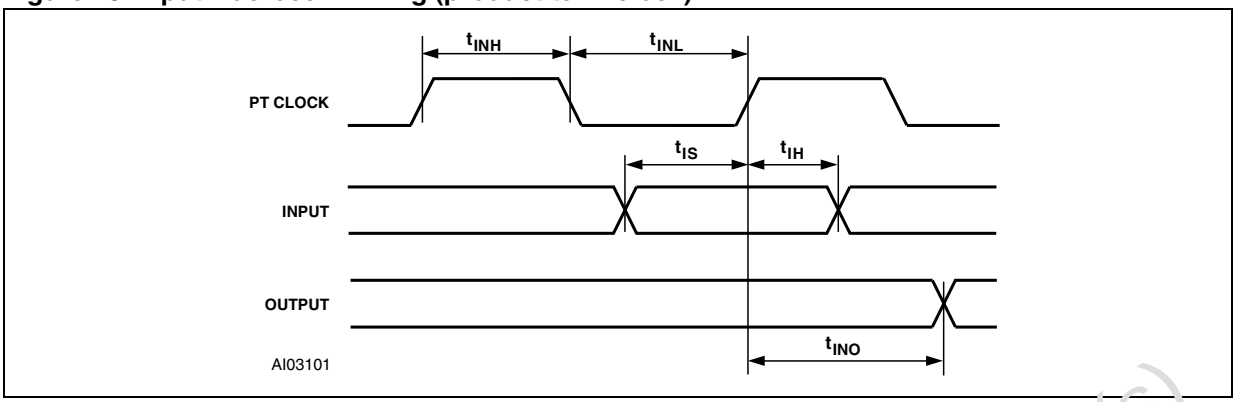



**Table 53. Input Macrocell Timing (5V devices)**

| Symbol | Parameter | Conditions | -90 | | -12 | | -15 | | PT Aloc | Turbo Off[2] | Unit |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | Min | Max | | | |
| $t_{IS}$ | Input Setup Time | (Note [1]) | 0 | | 0 | | 0 | | | | ns |
| $t_{IH}$ | Input Hold Time | (Note [1]) | 20 | | 22 | | 26 | | | + 10 | ns |
| $t_{INH}$ | NIB Input High Time | (Note [1]) | 12 | | 15 | | 18 | | | | ns |
| $t_{INL}$ | NIB Input Low Time | (Note [1]) | 12 | | 15 | | 18 | | | | ns |
| $t_{INO}$ | NIB Input to Combinatorial Delay | (Note [1]) | | 46 | | 50 | | 59 | + 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 $t_{AVLX}$ and $t_{LXAX}$.
2. ZPSD versions only.

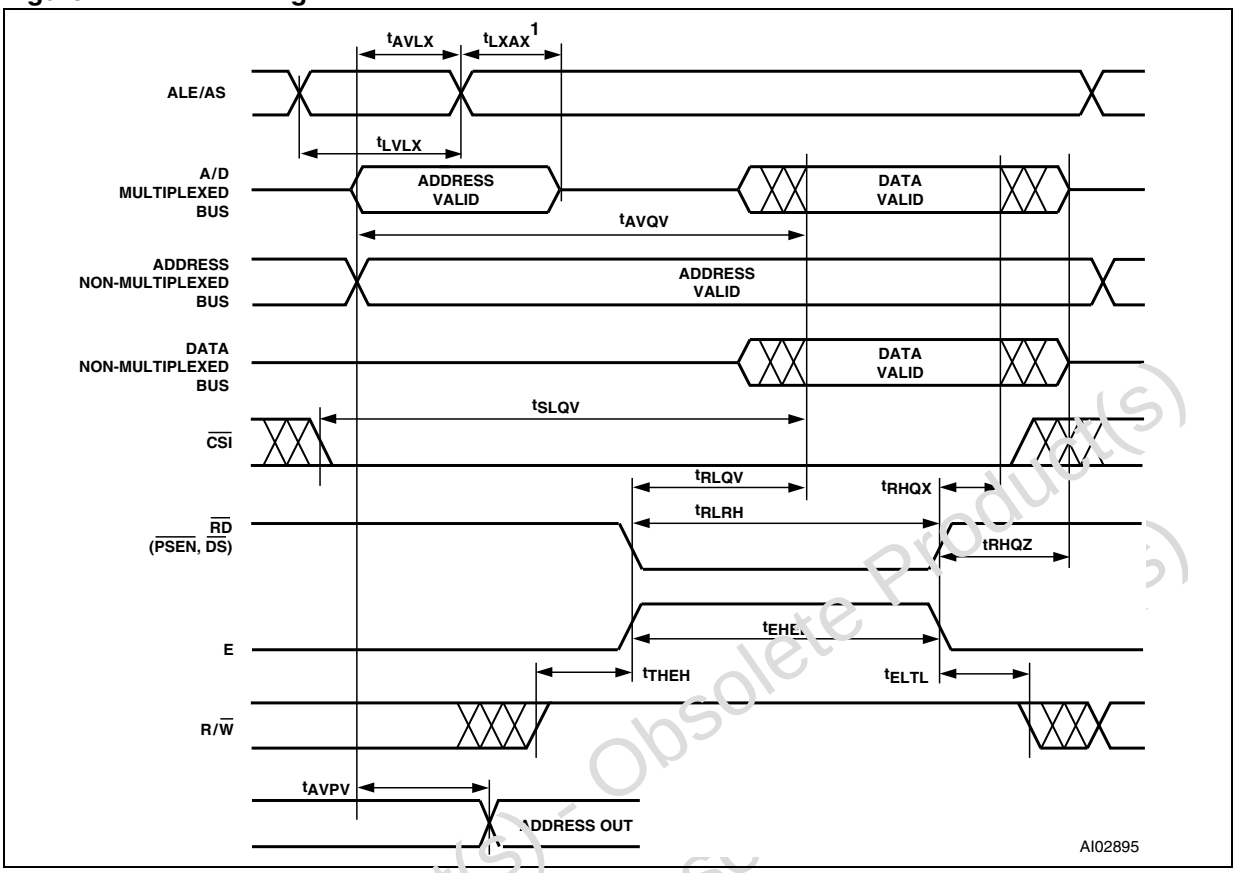
**Table 54. Input Macrocell Timing (3V Devices)**

| Symbol | Parameter | Conditions | -15 | | -20 | | PT Aloc | Turbo Off[2] | Unit |
|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | | | |
| $t_{IS}$ | Input Setup Time | (Note [1]) | 0 | | 0 | | | | ns |
| $t_{IH}$ | Input Hold Time | (Note [1]) | 25 | | 30 | | | + 20 | ns |
| $t_{INH}$ | NIB Input High Time | (Note [1]) | 13 | | 15 | | | | ns |
| $t_{INL}$ | NIB Input Low Time | (Note [1]) | 13 | | 15 | | | | ns |
| $t_{INO}$ | NIB Input to Combinatorial Delay | (Note [1]) | | 62 | | 70 | + 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 $t_{AVLX}$ and $t_{LXAX}$.
2. ZPSD Versions only.

Obsolete Product(s) - Obsolete Product(s)

**Figure 47. READ Timing**




Note: 1. $t_{AVLX}$ and $t_{LXAX}$ are not required for 80C251 in Page Mode or 80C51XA in Burst Mode.

Obsolete Product(s) - Obsolete Product(s)

**Table 55. READ Timing (5V devices)**

| Symbol | Parameter | Conditions | -90 | | -12 | | -15 | | Turbo Off | Unit |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | Min | Max | | |
| $t_{LVLX}$ | ALE or AS Pulse Width | | 20 | | 22 | | 28 | | | ns |
| $t_{AVLX}$ | Address Setup Time | (Note 3) | 6 | | 8 | | 10 | | | ns |
| $t_{LXAX}$ | Address Hold Time | (Note 3) | 8 | | 9 | | 11 | | | ns |
| $t_{AVQV}$ | Address Valid to Data Valid | (Notes 3,6) | | 90 | | 120 | | 150 | + 10 | ns |
| $t_{SLQV}$ | CS Valid to Data Valid | | | 100 | | 135 | | 150 | | ns |
| $t_{RLQV}$ | $\overline{RD}$ to Data Valid 8-Bit Bus | (Note 5) | | 32 | | 35 | | 40 | | ns |
| | $\overline{RD}$ or $\overline{PSEN}$ to Data Valid 8-Bit Bus, 8031, 80251 | (Note 2) | | 38 | | 42 | | 45 | | ns |
| $t_{RHQX}$ | $\overline{RD}$ Data Hold Time | (Note 1) | 0 | | 0 | | 0 | | | ns |
| $t_{RLRH}$ | $\overline{RD}$ Pulse Width | (Note 1) | 32 | | 35 | | 38 | | | ns |
| $t_{RHQZ}$ | $\overline{RD}$ to Data High-Z | (Note 1) | | 25 | | 35 | | 38 | | ns |
| $t_{EHEL}$ | E Pulse Width | | 32 | | 36 | | 38 | | | ns |
| $t_{THEH}$ | R/$\overline{W}$ Setup Time to Enable | | 10 | | 13 | | 18 | | | ns |
| $t_{ELTL}$ | R/$\overline{W}$ Hold Time After Enable | | 0 | | 0 | | 0 | | | ns |
| $t_{AVPV}$ | Address Input Valid to Address Output Delay | (Note 4) | | 25 | | 28 | | 32 | | ns |

Note: 1. $\overline{RD}$ timing has the same timing as $\overline{DS}$, $\overline{LDS}$, $\overline{UDS}$, and $\overline{PSEN}$ signals.
2. $\overline{RD}$ and $\overline{PSEN}$ have the same timing.
3. Any input used to select an internal PSD function.
4. In multiplexed mode, latched addresses generated from ADIO delay to address output on any Port.
5. $\overline{RD}$ timing has the same timing as $\overline{DS}$, $\overline{LDS}$, and $\overline{UDS}$ signals.
6. In Turbo Off mode, add 10 ns to $t_{AVQV}$.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

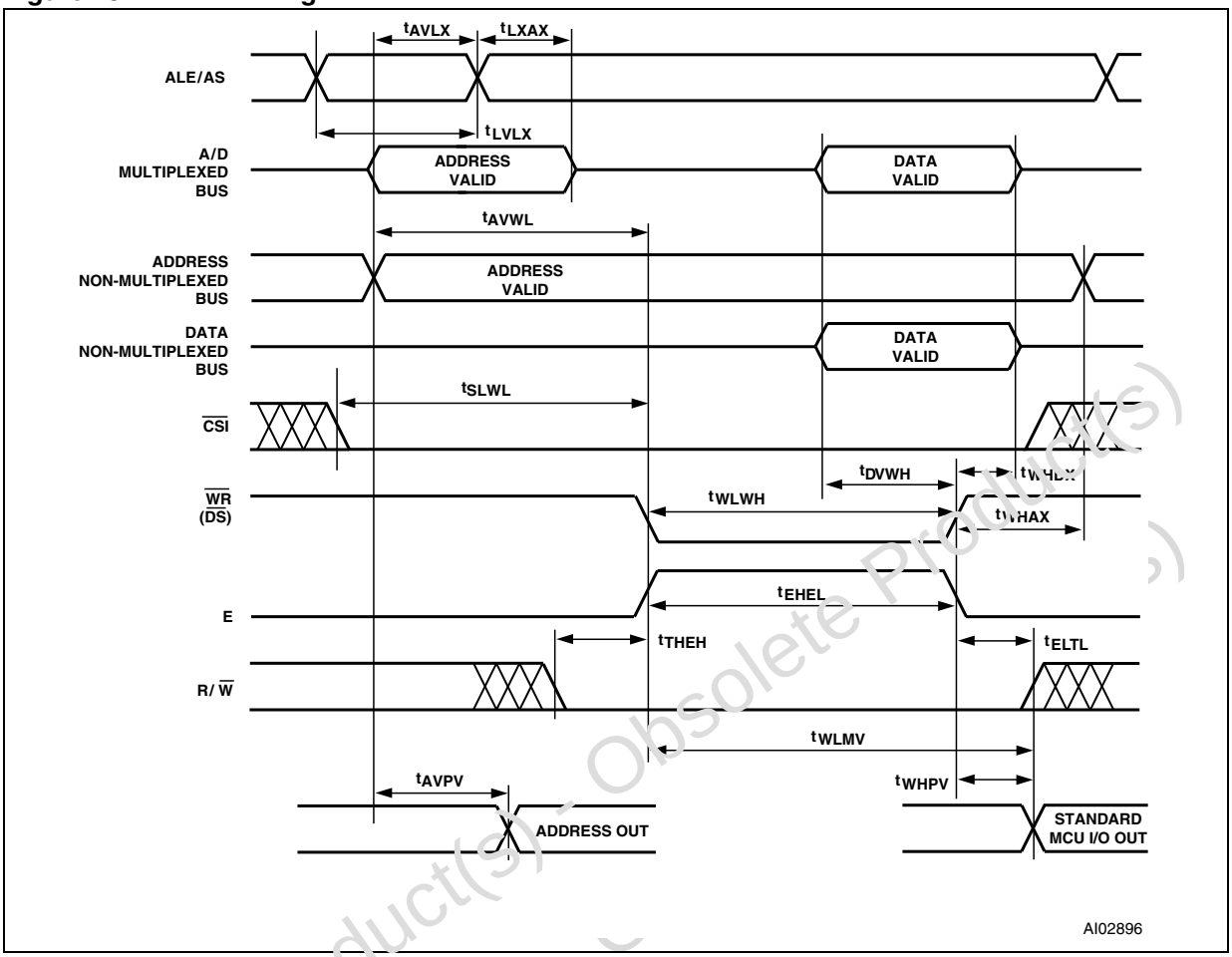
**Table 56. READ Timing (3V devices)**

| Symbol | Parameter | Conditions | -15 | | -20 | | Turbo Off | Unit |
|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | | |
| $t_{LVLX}$ | ALE or AS Pulse Width | | 26 | | 30 | | | ns |
| $t_{AVLX}$ | Address Setup Time | (Note 3) | 10 | | 12 | | | ns |
| $t_{LXAX}$ | Address Hold Time | (Note 3) | 12 | | 14 | | | ns |
| $t_{AVQV}$ | Address Valid to Data Valid | (Note 3,6) | | 150 | | 200 | + 20 | ns |
| $t_{SLQV}$ | CS Valid to Data Valid | | | 150 | | 200 | | ns |
| $t_{RLQV}$ | $\overline{RD}$ to Data Valid 8-Bit Bus | (Note 5) | | 35 | | 40 | | ns |
| | $\overline{RD}$ or $\overline{PSEN}$ to Data Valid 8-Bit Bus, 8031, 80251 | (Note 2) | | 50 | | 55 | | ns |
| $t_{RHQX}$ | $\overline{RD}$ Data Hold Time | (Note 1) | 0 | | 0 | | | ns |
| $t_{RLRH}$ | $\overline{RD}$ Pulse Width (also $\overline{DS}$, $\overline{LDS}$, $\overline{UDS}$) | | 40 | | 45 | | | ns |
| | $\overline{RD}$ or $\overline{PSEN}$ Pulse Width (8031, 80251) | | 55 | | 60 | | | ns |
| $t_{RHQZ}$ | $\overline{RD}$ to Data High-Z | (Note 1) | | 40 | | 45 | | ns |
| $t_{EHEL}$ | E Pulse Width | | 45 | | 52 | | | ns |
| $t_{THEH}$ | R/$\overline{W}$ Setup Time to Enable | | 18 | | 20 | | | ns |
| $t_{ELTL}$ | R/$\overline{W}$ Hold Time After Enable | | 0 | | 0 | | | ns |
| $t_{AVPV}$ | Address Input Valid to Address Output Delay | (Note 4) | | 35 | | 40 | | ns |

Note: 1. $\overline{RD}$ timing has the same timing as $\overline{DS}$, $\overline{LDS}$, $\overline{UDS}$, and $\overline{PSEN}$ signals.
2. $\overline{RD}$ and $\overline{PSEN}$ have the same timing for 8031.
3. Any input used to select an internal PSD function.
4. In multiplexed mode latched address generated from ADIO delay to address output on any Port.
5. $\overline{RD}$ timing has the same timing as $\overline{DS}$, $\overline{LDS}$, and $\overline{UDS}$ signals.
6. In Turbo Off mode, add 20ns to $t_{AVQV}$.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Figure 48. WRITE Timing**




Obsolete Product(s) - Obsolete Product(s)

**Table 57. WRITE, Erase and Program Timing (5V devices)**

| Symbol | Parameter | Conditions | -90 | | -12 | | -15 | | Unit |
|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | Min | Max | |
| $t_{LVLX}$ | ALE or AS Pulse Width | | 20 | | 22 | | 28 | | ns |
| $t_{AVLX}$ | Address Setup Time | (Note 1) | 6 | | 8 | | 10 | | ns |
| $t_{LXAX}$ | Address Hold Time | (Note 1) | 8 | | 9 | | 11 | | ns |
| $t_{AVWL}$ | Address Valid to Leading Edge of $\overline{WR}$ | (Notes 1,3) | 15 | | 18 | | 20 | | ns |
| $t_{SLWL}$ | $\overline{CS}$ Valid to Leading Edge of $\overline{WR}$ | (Note 3) | 15 | | 18 | | 20 | | ns |
| $t_{DVWH}$ | $\overline{WR}$ Data Setup Time | (Note 3) | 35 | | 40 | | 45 | | ns |
| $t_{WHDX}$ | $\overline{WR}$ Data Hold Time | (Note 3) | 5 | | 5 | | 5 | | ns |
| $t_{WLWH}$ | $\overline{WR}$ Pulse Width | (Note 3) | 35 | | 40 | | 45 | | ns |
| $t_{WHAX1}$ | Trailing Edge of $\overline{WR}$ to Address Invalid | (Note 3) | 8 | | 9 | | 10 | | ns |
| $t_{WHAX2}$ | Trailing Edge of $\overline{WR}$ to DPLD Address Invalid | (Note 3,6) | 0 | | 0 | | 0 | | ns |
| $t_{WHPV}$ | Trailing Edge of $\overline{WR}$ to Port Output Valid Using I/O Port Data Register | (Note 3) | | 30 | | 35 | | 38 | ns |
| $t_{DVMV}$ | Data Valid to Port Output Valid Using Macrocell Register Preset/Clear | (Notes 3,5) | | 55 | | 60 | | 65 | ns |
| $t_{AVPV}$ | Address Input Valid to Address Output Delay | (Note 2) | | 25 | | 28 | | 30 | ns |
| $t_{WLMV}$ | $\overline{WR}$ Valid to Port Output Valid Using Macrocell Register Preset/Clear | (Notes 3,4) | | 55 | | 60 | | 65 | ns |

Note: 1. Any input used to select an internal PSD function.
2. In multiplexed mode, latched address generated from ADIO delay to address output on any port.
3. $\overline{WR}$ has the same timing as E, $\overline{LDS}$, $\overline{UDS}$, $\overline{WRL}$, and $\overline{WRH}$ signals.
4. Assuming data is stable before active WRITE signal.
5. Assuming WRITE is active before data becomes valid.
6. $T_{WHAX2}$ is the address hold time for DPLD inputs that are used to generate Sector Select signals for internal PSD memory.

Obsolete Product(s) - Obsolete Product(s)

**Table 58. WRITE Timing (3V devices)**

| Symbol | Parameter | Conditions | -15 | | -20 | | Unit |
|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | |
| $t_{LVLX}$ | ALE or AS Pulse Width | | 26 | | 30 | | |
| $t_{AVLX}$ | Address Setup Time | (Note 1) | 10 | | 12 | | ns |
| $t_{LXAX}$ | Address Hold Time | (Note 1) | 12 | | 14 | | ns |
| $t_{AVWL}$ | Address Valid to Leading Edge of $\overline{WR}$ | (Notes 1,3) | 20 | | 25 | | ns |
| $t_{SLWL}$ | $\overline{CS}$ Valid to Leading Edge of $\overline{WR}$ | (Note 3) | 20 | | 25 | | ns |
| $t_{DVWH}$ | $\overline{WR}$ Data Setup Time | (Note 3) | 45 | | 50 | | ns |
| $t_{WHDX}$ | $\overline{WR}$ Data Hold Time | (Note 3) | 8 | | 10 | | ns |
| $t_{WLWH}$ | $\overline{WR}$ Pulse Width | (Note 3) | 48 | | 55 | | ns |
| $t_{WHAX1}$ | Trailing Edge of $\overline{WR}$ to Address Invalid | (Note 3) | 12 | | 17 | | ns |
| $t_{WHAX2}$ | Trailing Edge of $\overline{WR}$ to DPLD Address Invalid | (Note 3,6) | 0 | | 0 | | ns |
| $t_{WHPV}$ | Trailing Edge of $\overline{WR}$ to Port Output Valid Using I/O Port Data Register | (Note 3) | | 45 | | 50 | ns |
| $t_{DVMV}$ | Data Valid to Port Output Valid Using Macrocell Register Preset/Clear | (Notes 3,5) | | 90 | | 100 | ns |
| $t_{AVPV}$ | Address Input Valid to Address Output Delay | (Note 2) | | 48 | | 55 | ns |
| $t_{WLMV}$ | $\overline{WR}$ Valid to Port Output Valid Using Macrocell Register Preset/Clear | (Notes 3,4) | | 90 | | 100 | ns |

Note: 1. Any input used to select an internal PSD function.
2. In multiplexed mode, latched address generated from ADIO delay to address output on any port.
3. $\overline{WR}$ has the same timing as E, $\overline{LDS}$, $\overline{UDS}$, $\overline{WRL}$, and $\overline{WRH}$ signals.
4. Assuming data is stable before active WRITE signal.
5. Assuming WRITE is active before data becomes valid.
6. TWHAX2 is the address hold time for DPLD inputs that are used to generate Sector Select signals for internal PSD memory.

**Table 59. Flash Program, WRITE and Erase Times (5V devices)**

| Symbol | Parameter | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| | Flash Program | | 8.5 | | s |
| | Flash Bulk Erase[1] (pre-programmed) | | 3 | 30 | s |
| | Flash Bulk Erase (not pre-programmed) | | 10 | | s |
| $t_{WHQV3}$ | Sector Erase (pre-programmed) | | 1 | 30 | s |
| $t_{WHQV2}$ | Sector Erase (not pre-programmed) | | 2.2 | | s |
| $t_{WHQV1}$ | Byte Program | | 14 | 1200 | µs |
| | Program / Erase Cycles (per Sector) | 100,000 | | | cycles |
| $t_{WHWLO}$ | Sector Erase Time-Out | | 100 | | µs |
| $t_{Q7VQV}$ | DQ7 Valid to Output (DQ7-DQ0) Valid (Data Polling)[2] | | | 30 | ns |

Note: 1. Programmed to all zero before erase.
2. The polling status, DQ7, is valid $t_{Q7VQV}$ time units before the data byte, DQ0-DQ7, is valid for reading.

Obsolete Product(s) - Obsolete Product(s)

**Table 60. Flash Program, WRITE and Erase Times (3V devices)**

| Symbol | Parameter | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| | Flash Program | | 8.5 | | s |
| | Flash Bulk Erase[1] (pre-programmed) | | 3 | 30 | s |
| | Flash Bulk Erase (not pre-programmed) | | 10 | | s |
| $t_{WHQV3}$ | Sector Erase (pre-programmed) | | 1 | 30 | s |
| $t_{WHQV2}$ | Sector Erase (not pre-programmed) | | 2.2 | | s |
| $t_{WHQV1}$ | Byte Program | | 14 | 1200 | µs |
| | Program / Erase Cycles (per Sector) | 100,000 | | | cycles |
| $t_{WHWLO}$ | Sector Erase Time-Out | | 100 | | µs |
| $t_{Q7VQV}$ | DQ7 Valid to Output (DQ7-DQ0) Valid (Data Polling)[2] | | | 30 | ns |

Note: 1. Programmed to all zero before erase.
2. The polling status, DQ7, is valid $t_{Q7VQV}$ time units before the data byte, DQ0-DQ7, is valid for reading.

**Table 61. EEPROM WRITE Times (5V devices)**

| Symbol | Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $t_{EEHWL}$ | Write Protect After Power Up | | 5 | | ms |
| $t_{BLC}$ | EEPROM Byte Load Cycle Timing (Note [1]) | 0.2 | | 120 | µs |
| $t_{WCB}$ | EEPROM Byte Write Cycle Time | | 4 | 10 | ms |
| $t_{WCP}$ | EEPROM Page Write Cycle Time (Note [2]) | | 6 | 30 | ms |
| | Program/Erase Cycles (Per Sector) | 10,000 | | | cycles |

Note: 1. If the maximum time has elapsed between successive WRITE cycles to an EEPROM page, the transfer of this data to EEPROM cells will begin. Also, bytes cannot be written (loaded) to a page any faster than the indicated minimum type.
2. These specifications are for writing a page to EEPROM cells.

**Table 62. EEPROM WRITE Times (3V devices)**

| Symbol | Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $t_{EEHWL}$ | Write Protect After Power Up | | 5 | | ms |
| $t_{BLC}$ | EEPROM Byte Load Cycle Timing (Note [1]) | 0.2 | | 120 | µs |
| $t_{WCB}$ | EEPROM Byte Write Cycle Time | | 4 | 10 | ms |
| $t_{WCP}$ | EEPROM Page Write Cycle Time (Note [2]) | | 6 | 30 | ms |
| | Program/Erase Cycles (Per Sector) | 10,000 | | | cycles |

Note: 1. If the maximum time has elapsed between successive WRITE cycles to an EEPROM page, the transfer of this data to EEPROM cells will begin. Also, bytes cannot be written (loaded) to a page any faster than the indicated minimum type.
2. These specifications are for writing a page to EEPROM cells.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Figure 49. Peripheral I/O Read Timing**

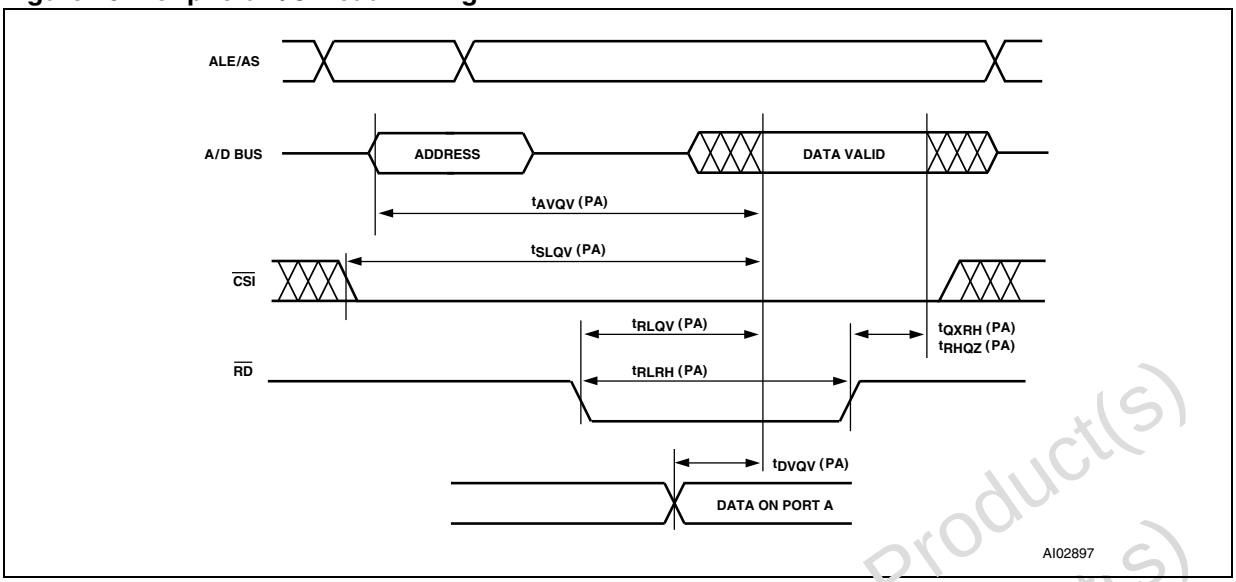



**Table 63. Port A Peripheral Data Mode READ Timing (5V devices)**

| Symbol | Parameter | Conditions | -90 | | -12 | | -15 | | Turbo Off | Unit |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | Min | Max | | |
| $t_{AVQV-PA}$ | Address Valid to Data Valid | (Note 3) | | 40 | | 45 | | 45 | + 10 | ns |
| $t_{SLQV-PA}$ | $\overline{CSI}$ Valid to Data Valid | | | 35 | | 40 | | 45 | + 10 | ns |
| $t_{RLQV-PA}$ | $\overline{RD}$ to Data Valid | (Notes 1,4) | | 32 | | 35 | | 40 | | ns |
| | $\overline{RD}$ to Data Valid 8031 Mode | | | 38 | | 42 | | 45 | | ns |
| $t_{DVQV-PA}$ | Data In to Data Out Valid | | | 30 | | 35 | | 38 | | ns |
| $t_{QXRH-PA}$ | $\overline{RD}$ Data Hold Time | | 0 | | 0 | | 0 | | | ns |
| $t_{RLRH-PA}$ | $\overline{RD}$ Pulse Width | (Note 1) | 32 | | 35 | | 38 | | | ns |
| $t_{RHQZ-PA}$ | $\overline{RD}$ to Data High-Z | (Note 1) | | 25 | | 28 | | 30 | | ns |

**Table 64. Port A Peripheral Data Mode READ Timing (3V devices)**

| Symbol | Parameter | Conditions | -15 | | -20 | | Turbo Off | Unit |
|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | | |
| $t_{AVQV-PA}$ | Address Valid to Data Valid | (Note 3) | | 55 | | 60 | + 20 | ns |
| $t_{SLQV-PA}$ | $\overline{CSI}$ Valid to Data Valid | | | 45 | | 50 | + 20 | ns |
| $t_{RLQV-PA}$ | $\overline{RD}$ to Data Valid | (Notes 1,4) | | 40 | | 45 | | ns |
| | $\overline{RD}$ to Data Valid 8031 Mode | | | 45 | | 50 | | ns |
| $t_{DVQV-PA}$ | Data In to Data Out Valid | | | 60 | | 65 | | ns |
| $t_{QXRH-PA}$ | $\overline{RD}$ Data Hold Time | | 0 | | 0 | | | ns |
| $t_{RLRH-PA}$ | $\overline{RD}$ Pulse Width | (Note 1) | 36 | | 46 | | | ns |
| $t_{RHQZ-PA}$ | $\overline{RD}$ to Data High-Z | (Note 1) | | 40 | | 45 | | ns |

Obsolete Product(s) - Obsolete Product(s) - Obsolete Product(s)

**Figure 50. Peripheral I/O WRITE Timing**

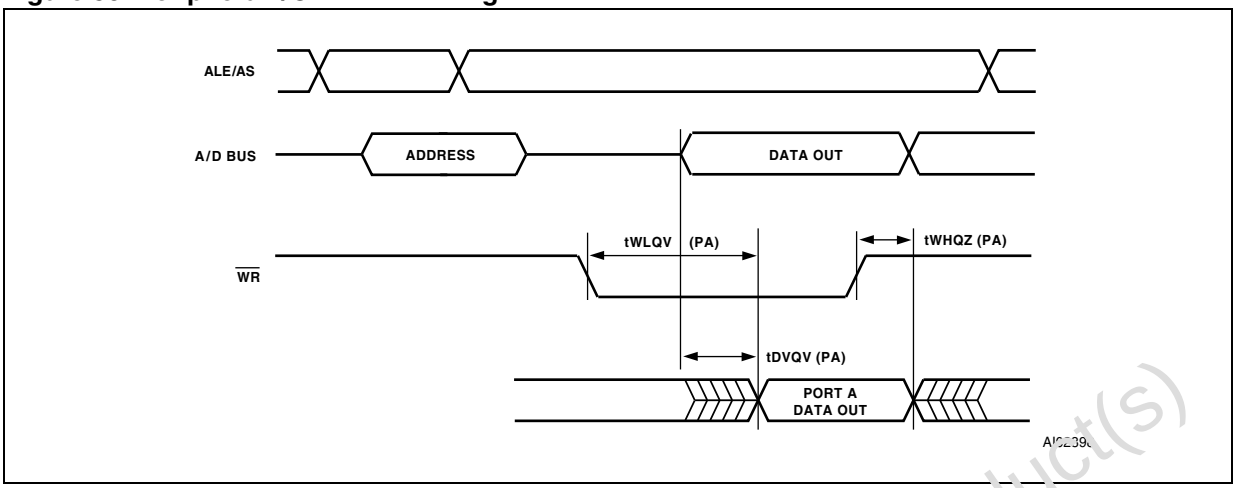



**Table 65. Port A Peripheral Data Mode WRITE Timing (5V devices)**

| Symbol | Parameter | Conditions | -90 | | -12 | | -15 | | Unit |
|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | Min | Max | |
| $t_{WLQV-PA}$ | $\overline{WR}$ to Data Propagation Delay | (Note 2) | | 35 | | 38 | | 40 | ns |
| $t_{DVQV-PA}$ | Data to Port A Data Propagation Delay | (Note 5) | | 30 | | 35 | | 38 | ns |
| $t_{WHQZ-PA}$ | $\overline{WR}$ Invalid to Port A Tri-state | (Note 2) | | 25 | | 30 | | 33 | ns |

Note: 1. $\overline{RD}$ has the same timing as $\overline{DS}$, $\overline{LDS}$, $\overline{UDS}$, and $\overline{PSEN}$ (in 8031 combined mode).
2. $\overline{WR}$ has the same timing as the E, $\overline{LDS}$, $\overline{UDS}$, $\overline{WRL}$, and $\overline{WRH}$ signals.
3. Any input used to select Port A Data Peripheral mode.
4. Data is already stable on Port A.
5. Data stable on ADIO pins to data on Port A.

**Table 66. Port A Peripheral Data Mode WRITE Timing (3V devices)**

| Symbol | Parameter | Conditions | -15 | | -20 | | Unit |
|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | |
| $t_{WLQV-PA}$ | $\overline{WR}$ to Data Propagation Delay | (Note 2) | | 45 | | 55 | ns |
| $t_{DVQV-PA}$ | Data to Port A Data Propagation Delay | (Note 5) | | 40 | | 45 | ns |
| $t_{WHQZ-PA}$ | $\overline{WR}$ Invalid to Port A Tri-state | (Note 2) | | 33 | | 35 | ns |

Note: 1. $\overline{RD}$ has the same timing as $\overline{DS}$, $\overline{LDS}$, $\overline{UDS}$, and $\overline{PSEN}$ (in 8031 combined mode) signals.
2. $\overline{WR}$ has the same timing as the E, $\overline{LDS}$, $\overline{UDS}$, $\overline{WRL}$, and $\overline{WRH}$ signals.
3. Any input used to select Port A Data Peripheral mode.
4. Data is already stable on Port A.
5. Data stable on ADIO pins to data on Port A.

Obsolete Product(s) - Obsolete Product(s)

**Figure 51. Reset ($\overline{RESET}$) Timing**

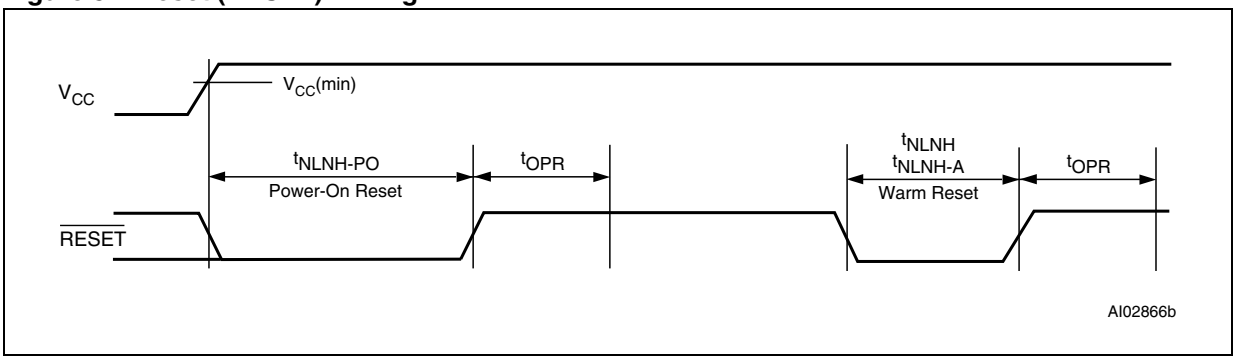



**Table 67. Reset ($\overline{RESET}$) Timing (5V devices)**

| Symbol | Parameter | Conditions | Min | Max | Unit |
|---|---|---|---|---|---|
| $t_{NLNH}$ | RESET Active Low Time [1] | | 150 | | ns |
| $t_{NLNH-PO}$ | Power On Reset Active Low Time | | 1 | | ms |
| $t_{OPR}$ | RESET High to Operational Device | | | 120 | ns |

Note: 1. Reset ($\overline{RESET}$) 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 68. Reset ($\overline{RESET}$) Timing (3V devices)**

| Symbol | Parameter | Conditions | Min | Max | Unit |
|---|---|---|---|---|---|
| $t_{NLNH}$ | RESET Active Low Time [1] | | 300 | | ns |
| $t_{NLNH-PO}$ | Power On Reset Active Low Time [2] | | 1 | | ms |
| $t_{OPR}$ | RESET High to Operational Device | | | 300 | ns |

Note: 1. Reset ($\overline{RESET}$) does not reset Flash memory Program or Erase cycles.
2. $t_{NLNH-PO}$ is 10ms for devices manufactured before the rev.A.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Figure 52. ISC Timing**

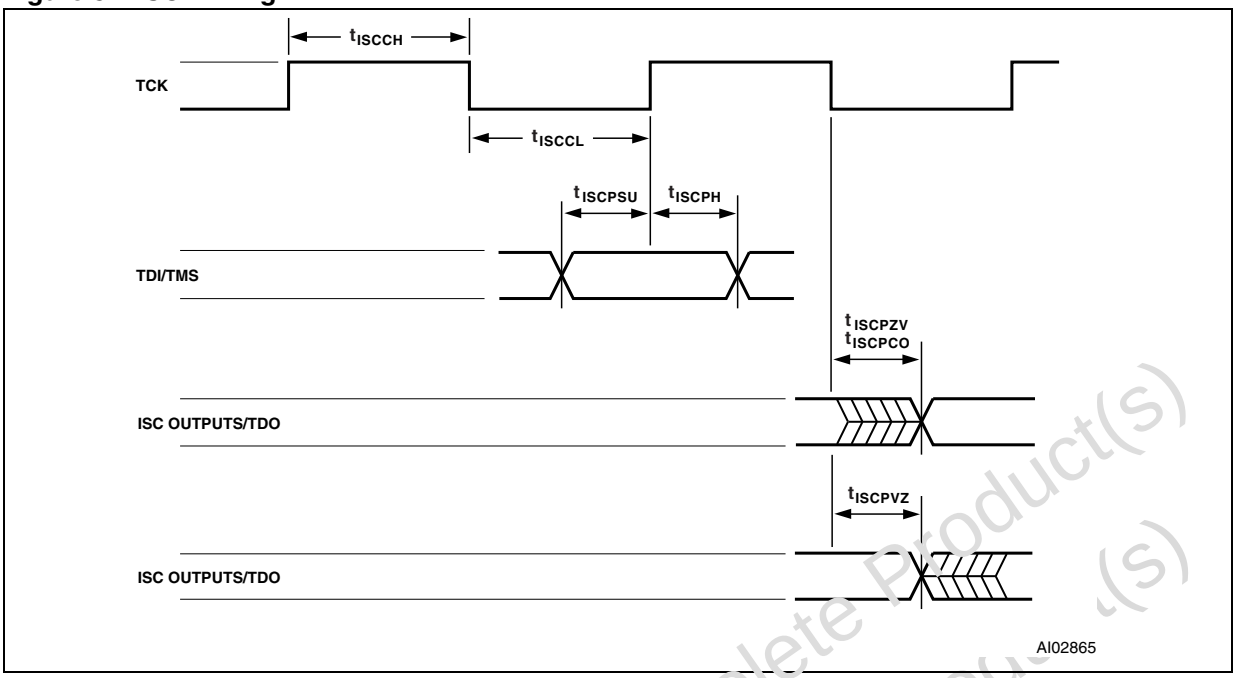



**Table 69. ISC Timing (5V devices)**

| Symbol | Parameter | Conditions | -90 | | -12 | | -15 | | Unit |
|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | Min | Max | |
| $t_{ISCCF}$ | Clock (TCK, PC1) Frequency (except for PLD) | (Note 1) | | 18 | | 16 | | 14 | MHz |
| $t_{ISCCH}$ | Clock (TCK, PC1) High Time (except for PLD) | (Note 1) | 26 | | 29 | | 31 | | ns |
| $t_{ISCCL}$ | Clock (TCK, PC1) Low Time (except for PLD) | (Note 1) | 26 | | 29 | | 31 | | ns |
| $t_{ISCCFP}$ | Clock (TCK, PC1) Frequency (PLD only) | (Note 2) | | 2 | | 2 | | 2 | MHz |
| $t_{ISCCHP}$ | Clock (TCK, PC1) High Time (PLD only) | (Note 2) | 240 | | 240 | | 240 | | ns |
| $t_{ISCCLP}$ | Clock (TCK, PC1) Low Time (PLD only) | (Note 2) | 240 | | 240 | | 240 | | ns |
| $t_{ISCPSU}$ | ISC Port Set Up Time | | 8 | | 10 | | 10 | | ns |
| $t_{ISCPH}$ | ISC Port Hold Up Time | | 5 | | 5 | | 5 | | ns |
| $t_{ISCPCO}$ | ISC Port Clock to Output | | | 23 | | 24 | | 25 | ns |
| $t_{ISCPZV}$ | ISC Port High-Impedance to Valid Output | | | 23 | | 24 | | 25 | ns |
| $t_{ISCPVZ}$ | ISC Port Valid Output to High-Impedance | | | 23 | | 24 | | 25 | ns |

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

Obsolete Product(s) - Obsolete Product(s)

**Table 70. ISC Timing (3V devices)**

| Symbol | Parameter | Conditions | -15 | | -20 | | Unit |
|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | |
| $t_{ISCCF}$ | Clock (TCK, PC1) Frequency (except for PLD) | (Note 1) | | 10 | | 9 | MHz |
| $t_{ISCCH}$ | Clock (TCK, PC1) High Time (except for PLD) | (Note 1) | 45 | | 51 | | ns |
| $t_{ISCCL}$ | Clock (TCK, PC1) Low Time (except for PLD) | (Note 1) | 45 | | 51 | | ns |
| $t_{ISCCFP}$ | Clock (TCK, PC1) Frequency (PLD only) | (Note 2) | | 2 | | 2 | MHz |
| $t_{ISCCHP}$ | Clock (TCK, PC1) High Time (PLD only) | (Note 2) | 240 | | 240 | | ns |
| $t_{ISCCLP}$ | Clock (TCK, PC1) Low Time (PLD only) | (Note 2) | 240 | | 240 | | ns |
| $t_{ISCPSU}$ | ISC Port Set Up Time | | 13 | | 15 | | ns |
| $t_{ISCPH}$ | ISC Port Hold Up Time | | 10 | | 10 | | ns |
| $t_{ISCPCO}$ | ISC Port Clock to Output | | | 36 | | 40 | ns |
| $t_{ISCPZV}$ | ISC Port High-Impedance to Valid Output | | | 36 | | 40 | ns |
| $t_{ISCPVZ}$ | ISC Port Valid Output to High-Impedance | | | 36 | | 40 | ns |

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

**Table 71. Power-down Timing (5V devices)**

| Symbol | Parameter | Conditions | -90 | | -12 | | -15 | | Unit |
|---|---|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | Min | Max | |
| $t_{LVDV}$ | ALE Access Time from Power-down | | | 90 | | 120 | | 150 | ns |
| $t_{CLWH}$ | Maximum Delay from APD Enable to Internal PDN Valid Signal | Using CLKIN (PD1) | $15 * t_{CLCL}$ (1) | | | | | | µs |

Note: 1. $t_{CLCL}$ is the period of CLKIN (PD1).

**Table 72. Power-down Timing (3V devices)**

| Symbol | Parameter | Conditions | -15 | | -20 | | Unit |
|---|---|---|---|---|---|---|---|
| | | | Min | Max | Min | Max | |
| $t_{LVDV}$ | ALE Access Time from Power-down | | | 150 | | 200 | ns |
| $t_{CLWH}$ | Maximum Delay from APD Enable to Internal PDN Valid Signal | Using CLKIN (PD1) | $15 * t_{CLCL}$ (1) | | | | µs |

Note: 1. $t_{CLCL}$ is the period of CLKIN (PD1).

Obsolete Product(s) - Obsolete Product(s)

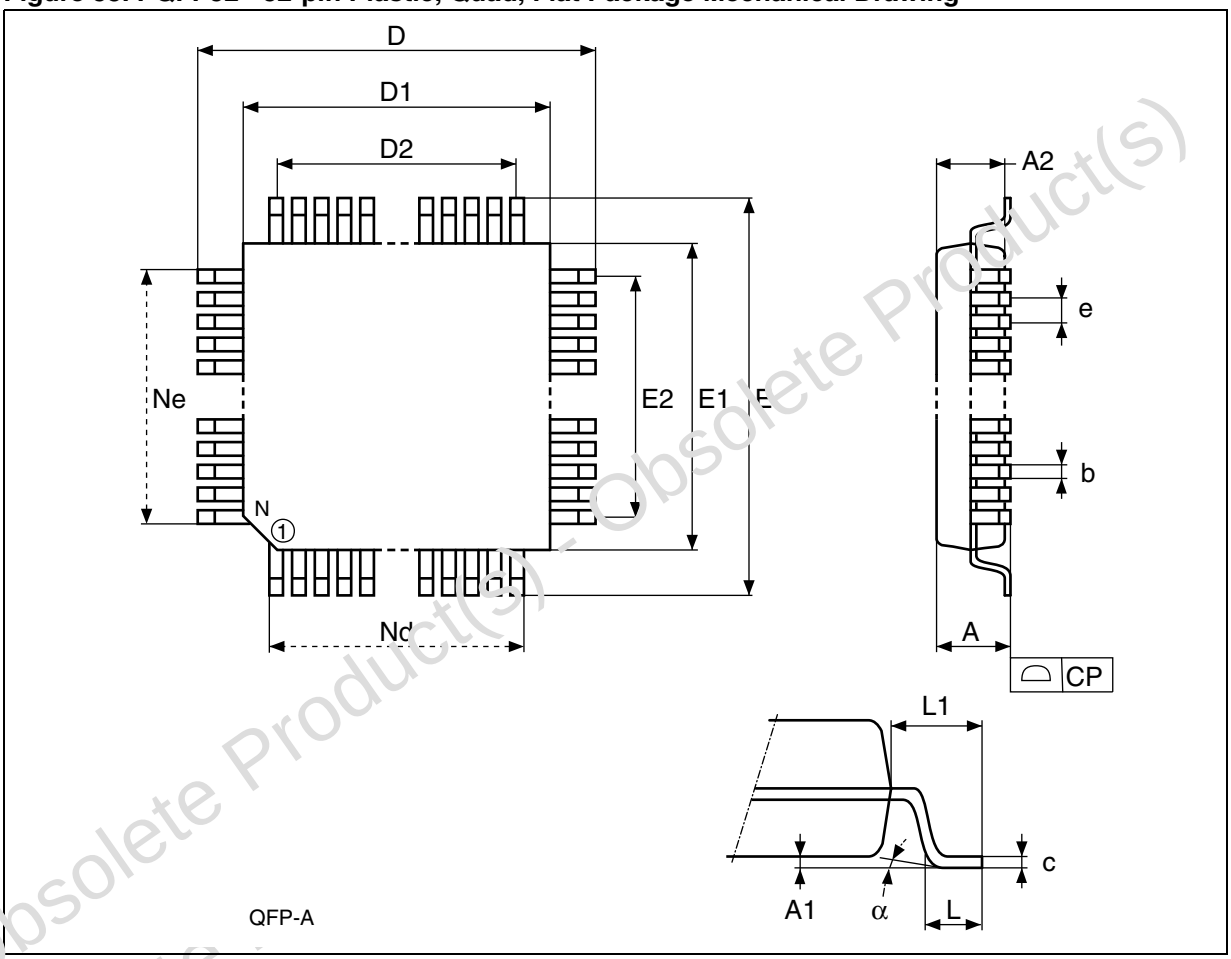
## PACKAGE MECHANICAL

In order to meet environmental requirements, ST offers these devices in ECOPACK® packages. These packages have a Lead-free second level interconnect . The category of second level interconnect is marked on the package and on the inner box label, in compliance with JEDEC Standard JESD97.

The maximum ratings related to soldering conditions are also marked on the inner box label.

ECOPACK is an ST trademark. ECOPACK specifications are available at: www.st.com.

**Figure 53. PQFP52 - 52-pin Plastic, Quad, Flat Package Mechanical Drawing**




Note: Drawing is not to scale.

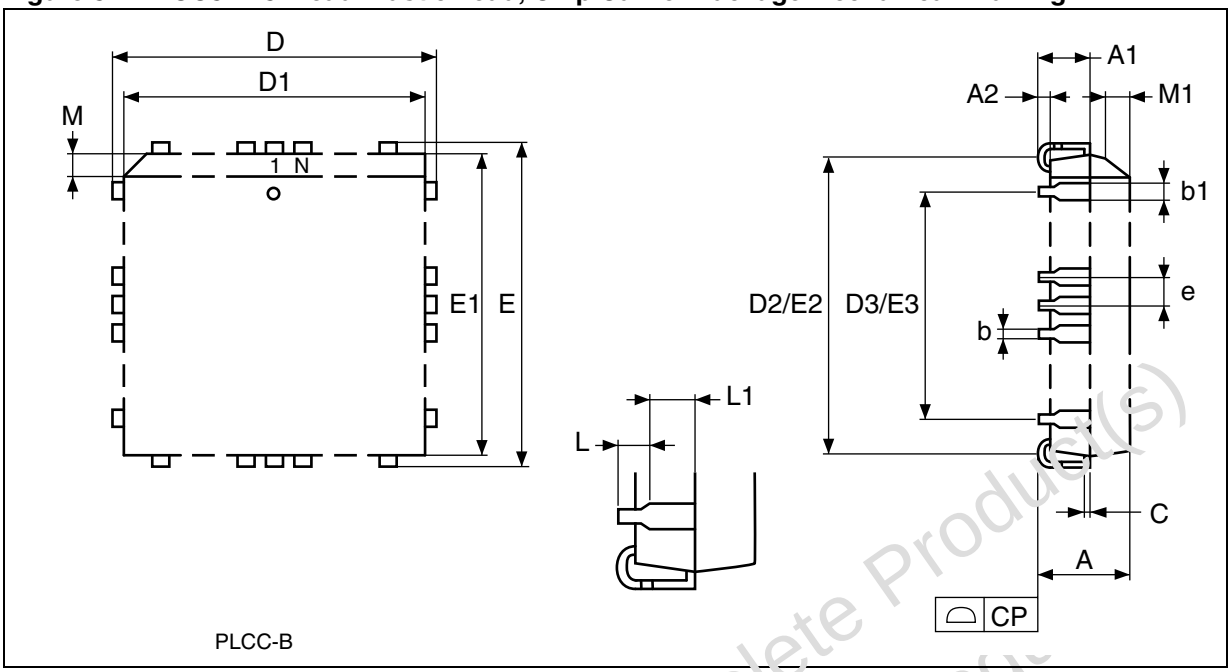
Obsolete Product(s) - Obsolete Product(s)

**Table 73. PQFP52 - 52-pin Plastic, Quad, Flat Package Mechanical Dimensions**

| Symb. | mm | | | inches | | |
|---|---|---|---|---|---|---|
| | Typ. | Min. | Max. | Typ. | Min. | Max. |
| A | | | 2.35 | | | 0.093 |
| A1 | | | 0.25 | | | 0.010 |
| A2 | 2.00 | 1.80 | 2.10 | 0.079 | 0.077 | 0.083 |
| b | | 0.22 | 0.38 | | 0.009 | 0.015 |
| c | | 0.11 | 0.23 | | 0.004 | 0.009 |
| D | 13.20 | 13.15 | 13.25 | 0.520 | 0.518 | 0.522 |
| D1 | 10.00 | 9.95 | 10.05 | 0.394 | 0.392 | 0.396 |
| D2 | 7.80 | – | – | 0.307 | – | – |
| E | 13.20 | 13.15 | 13.25 | 0.520 | 0.518 | 0.522 |
| E1 | 10.00 | 9.95 | 10.05 | 0.394 | 0.392 | 0.396 |
| E2 | 7.80 | – | – | 0.307 | – | – |
| e | 0.65 | – | – | 0.026 | | |
| L | 0.88 | 0.73 | 1.03 | 0.035 | 0.029 | 0.041 |
| L1 | 1.60 | – | – | 0.063 | | |
| α | | 0° | 7° | | 0° | 7° |
| N | 52 | | | 52 | | |
| Nd | 13 | | | 13 | | |
| Ne | 13 | | | 13 | | |
| CP | | | 0.10 | | | 0.004 |

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

**Figure 54. PLCC52 - 52-lead Plastic Lead, Chip Carrier Package Mechanical Drawing**



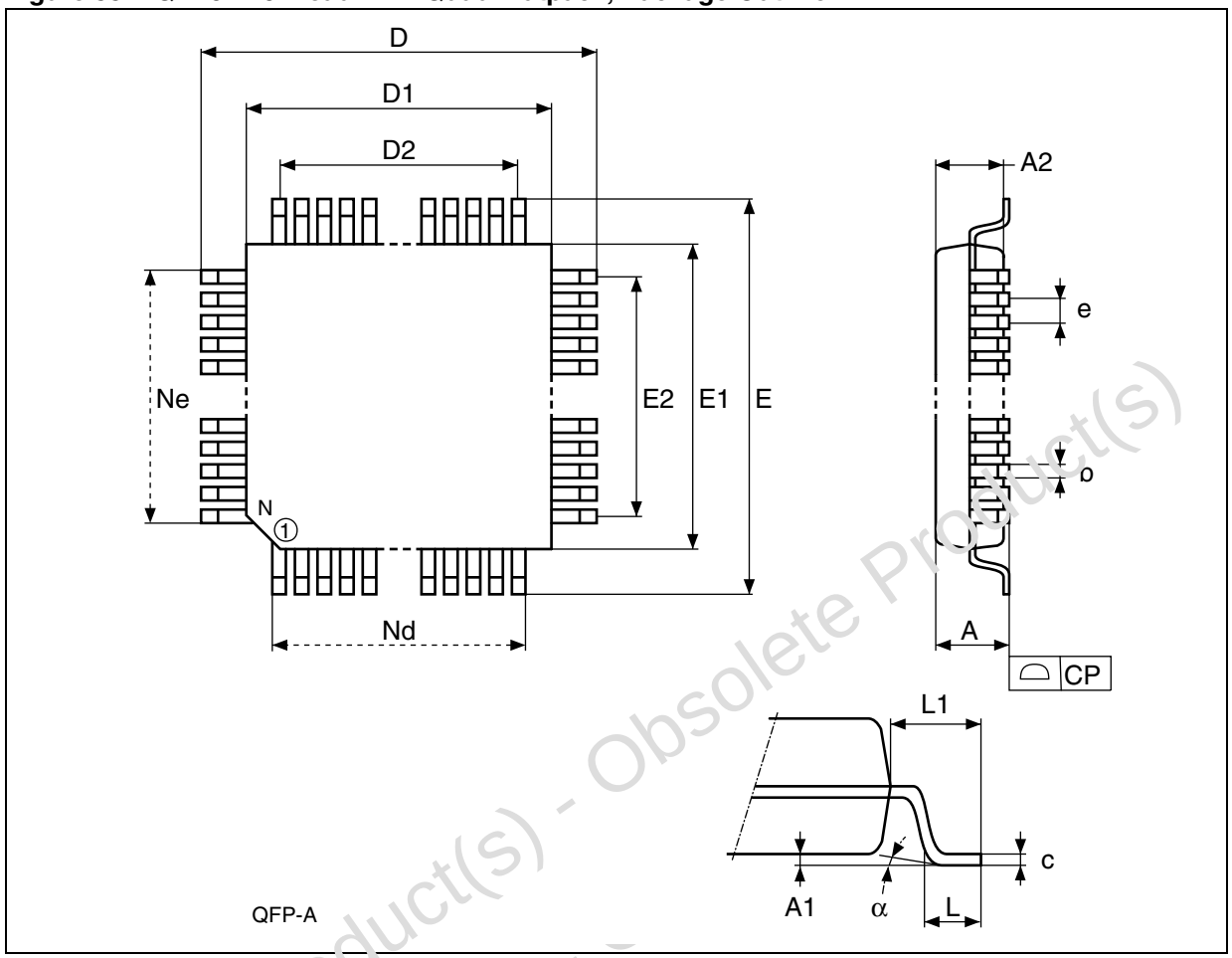

Note: Drawing is not to scale.

**Table 74. PLCC52 - 52-lead Plastic Lead, Chip Carrier Package Mechanical Dimensions**

| Symbol | mm | | | inches | | |
|---|---|---|---|---|---|---|
| | Typ. | Min. | Max. | Typ. | Min. | Max. |
| A | | 4.19 | 4.57 | | 0.165 | 0.180 |
| A1 | | 2.54 | 2.79 | | 0.100 | 0.110 |
| A2 | | – | 0.91 | | – | 0.036 |
| B | | 0.33 | 0.53 | | 0.013 | 0.021 |
| B1 | | 0.66 | 0.81 | | 0.026 | 0.032 |
| C | | 0.246 | 0.261 | | 0.0097 | 0.0103 |
| D | | 19.94 | 20.19 | | 0.785 | 0.795 |
| D1 | | 19.05 | 19.15 | | 0.750 | 0.754 |
| D2 | | 17.53 | 18.54 | | 0.690 | 0.730 |
| E | | 19.94 | 20.19 | | 0.785 | 0.795 |
| E1 | | 19.05 | 19.15 | | 0.750 | 0.754 |
| E2 | | 17.53 | 18.54 | | 0.690 | 0.730 |
| e | 1.27 | – | – | 0.050 | – | – |
| R | 0.89 | – | – | 0.035 | – | – |
| N | 52 | | | 52 | | |
| Nd | 13 | | | 13 | | |
| Ne | 13 | | | 13 | | |

Obsolete Product(s) - Obsolete Product(s)

**Figure 55. TQFP64 - 64-lead Thin Quad Flatpack, Package Outline**




Note: Drawing is not to scale.

Obsolete Product(s) - Obsolete Product(s)

**Table 75. TQFP64 - 64-lead Thin Quad Flatpack, Package Mechanical Data**

| Symb. | mm | | | inches | | |
|---|---|---|---|---|---|---|
| | Typ. | Min. | Max. | Typ. | Min. | Max. |
| A | | 1.42 | 1.54 | | 0.056 | 0.061 |
| A1 | 0.10 | 0.07 | 0.14 | 0.004 | 0.003 | 0.005 |
| A2 | 1.40 | 1.36 | 1.44 | 0.055 | 0.054 | 0.057 |
| α | 3.5° | 0.0° | 7.0° | 3.5° | 0.0° | 7.0° |
| b | 0.35 | 0.33 | 0.38 | 0.014 | 0.013 | 0.015 |
| c | | | 0.17 | | | 0.006 |
| D | 16.00 | 15.90 | 16.10 | 0.630 | 0.626 | 0.634 |
| D1 | 14.00 | 13.98 | 14.03 | 0.551 | 0.550 | 0.552 |
| D2 | 12.00 | 11.95 | 12.05 | 0.472 | 0.470 | 0.474 |
| E | 16.00 | 15.90 | 16.10 | 0.630 | 0.626 | 0.634 |
| E1 | 14.00 | 13.98 | 14.03 | 0.551 | 0.550 | 0.552 |
| E2 | 12.00 | 11.95 | 12.05 | 0.472 | 0.470 | 0.474 |
| e | 0.80 | 0.75 | 0.85 | 0.031 | 0.030 | 0.033 |
| L | 0.60 | 0.45 | 0.75 | 0.024 | 0.018 | 0.030 |
| L1 | 1.00 | 0.94 | 1.06 | 0.039 | 0.037 | 0.042 |
| CP | 0.10 | | | 0.004 | | |
| N | 64 | | | 64 | | |
| Nd | 16 | | | 16 | | |
| Ne | 16 | | | 16 | | |

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

## PART NUMBERING

**Table 76. Ordering Information Scheme**

Example: PSD8 1 3 F 1 V 12 J 1 T

**Device Type**

PSD8 = 8-bit PSD with Register Logic

**SRAM Capacity**

1 = 16 Kbit

**Flash Memory Capacity**

3 = 1 Mbit (128Kb x 8)

**2nd Flash Memory**

1 = 256 Kbit EEPROM

**Operating Voltage**

V = $V_{CC}$ = 3.0 to 3.6V

**Speed**

70 = 70ns

90 = 90ns

12 = 120ns

**Package**

J = ECOPACK PLCC52

M = ECOPACK PQFP52

U = ECOPACK TQFP64

**Temperature Range**

blank = 0 to 70°C (commercial)

1= –40 to 85°C (industrial)

**Option**

T = Tape & Reel Packing

For other options, or for more information on any aspect of this device, please contact the ST Sales Office nearest you.

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

# APPENDIX A. PQFP52 PIN ASSIGNMENTS

**Table 77. PQFP52 Connections (Figure 2)**

| Pin Number | Pin Assignments |
|---|---|
| 1 | PD2 |
| 2 | PD1 |
| 3 | PD0 |
| 4 | PC7 |
| 5 | PC6 |
| 6 | PC5 |
| 7 | PC4 |
| 8 | $V_{CC}$ |
| 9 | GND |
| 10 | PC3 |
| 11 | PC2 |
| 12 | PC1 |
| 13 | PC0 |
| 14 | PA7 |
| 15 | PA6 |
| 16 | PA5 |
| 17 | PA4 |
| 18 | PA3 |
| 19 | GND |
| 20 | PA2 |
| 21 | PA1 |
| 22 | PA0 |
| 23 | AD0 |
| 24 | AD1 |
| 25 | AD2 |
| 26 | AD3 |

| Pin Number | Pin Assignments |
|---|---|
| 27 | AD4 |
| 28 | AD5 |
| 29 | AD6 |
| 30 | AD7 |
| 31 | $V_{CC}$ |
| 32 | AD8 |
| 33 | AD9 |
| 34 | AD10 |
| 35 | AD11 |
| 36 | AD12 |
| 37 | AD13 |
| 38 | AD14 |
| 39 | AD15 |
| 40 | CNTL0 |
| 41 | $\overline{\text{RESET}}$ |
| 42 | CNTL2 |
| 43 | CNTL1 |
| 44 | PB7 |
| 45 | PB6 |
| 46 | GND |
| 47 | PB5 |
| 48 | PB4 |
| 49 | PB3 |
| 50 | PB2 |
| 51 | PB1 |
| 52 | PB0 |

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

# APPENDIX B. PLCC52 PIN ASSIGNMENTS

**Table 78. PLCC52 Connections**

| Pin Number | Pin Assignments |
|---|---|
| 1 | GND |
| 2 | PB5 |
| 3 | PB4 |
| 4 | PB3 |
| 5 | PB2 |
| 6 | PB1 |
| 7 | PB0 |
| 8 | PD2 |
| 9 | PD1 |
| 10 | PD0 |
| 11 | PC7 |
| 12 | PC6 |
| 13 | PC5 |
| 14 | PC4 |
| 15 | $V_{CC}$ |
| 16 | GND |
| 17 | PC3 |
| 18 | PC2 |
| 19 | PC1 |
| 20 | PC0 |
| 21 | PA7 |
| 22 | PA6 |
| 23 | PA5 |
| 24 | PA4 |
| 25 | PA3 |
| 26 | GND |

| Pin Number | Pin Assignments |
|---|---|
| 27 | PA2 |
| 28 | PA1 |
| 29 | PA0 |
| 30 | AD0 |
| 31 | AD1 |
| 32 | AD2 |
| 33 | AD3 |
| 34 | AD4 |
| 35 | AD5 |
| 36 | AD6 |
| 37 | AD7 |
| 38 | $V_{CC}$ |
| 39 | AD8 |
| 40 | AD9 |
| 41 | AD10 |
| 42 | AD11 |
| 43 | AD12 |
| 44 | AD13 |
| 45 | AD14 |
| 46 | AD15 |
| 47 | CNTL0 |
| 48 | $\overline{\text{RESET}}$ |
| 49 | CNTL2 |
| 50 | CNTL1 |
| 51 | PB7 |
| 52 | PB6 |

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

## APPENDIX C. TQFP64 PIN ASSIGNMENTS

**Table 79. TQFP64 Connections (Figure 4)**

| Pin Number | Pin Assignments |
|---|---|
| 1 | PD2 |
| 2 | PD1 |
| 3 | PD0 |
| 4 | PC7 |
| 5 | PC6 |
| 6 | PC5 |
| 7 | $V_{CC}$ |
| 8 | $V_{CC}$ |
| 9 | $V_{CC}$ |
| 10 | GND |
| 11 | GND |
| 12 | PC3 |
| 13 | PC2 |
| 14 | PC1 |
| 15 | PC0 |
| 16 | NC |
| 17 | NC |
| 18 | NC |
| 19 | PA7 |
| 20 | PA6 |
| 21 | PA5 |
| 22 | PA4 |
| 23 | PA3 |
| 24 | GND |
| 25 | GND |
| 26 | PA2 |
| 27 | PA1 |
| 28 | PA0 |
| 29 | AD0 |
| 30 | AD1 |
| 31 | N/D |
| 32 | AD2 |

| Pin Number | Pin Assignments |
|---|---|
| 33 | AD3 |
| 34 | AD4 |
| 35 | AD5 |
| 36 | AD6 |
| 37 | AD7 |
| 38 | $V_{CC}$ |
| 39 | $V_{CC}$ |
| 40 | AD8 |
| 41 | AD9 |
| 42 | AD10 |
| 43 | AD11 |
| 44 | AD12 |
| 45 | AD13 |
| 46 | AD14 |
| 47 | AD15 |
| 48 | CNTL0 |
| 49 | NC |
| 50 | $\overline{\text{RESET}}$ |
| 51 | CNTL2 |
| 52 | CNTL1 |
| 53 | PB7 |
| 54 | PB6 |
| 55 | GND |
| 56 | GND |
| 57 | PB5 |
| 58 | PB4 |
| 59 | PB3 |
| 60 | PB2 |
| 61 | PB1 |
| 62 | PB0 |
| 63 | NC |
| 64 | NC |

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

# REVISION HISTORY

**Table 80. Document Revision History**

| Date | Rev. | Description of Revision |
|---|---|---|
| August-2000 | 1.0 | Document written in WSI format. |
| 04-Jan-03 | 1.1 | Front page, and back two pages, in ST format, added to the PDF file. References to Waferscale, WSI, EasyFLASH and PSDsoft 2000 updated to ST, ST, Flash+PSD and PSDsoft Express. |
| 06-Dec-03 | 2.0 | Document converted to ST format. Package references corrected (Figure 1). |
| 3-Jun-04 | 3.0 | Document reformatted for DMS; Ordering Information corrected (Table 76); added TQFP64 package (Figure 1, 55; Table 75) |
| 03-Oct-2008 | 4 | Added ECOPACK text in cover page and in section PACKAGE MECHANICAL, page 100. Updated datasheet status to “not for new design”. Backup battery feature removed: updated features summary, Table 1 (pins PC2 and PC4 configurations ), PSD ARCHITECTURAL OVERVIEW, Memory section, SRAM section, Port C – Functionality and Structure section. Removed SRAM standby mode in POWER MANAGEMENT. Updated PC2 in Table 78. PLCC52 Connections. Removed $V_{STBY}$, $I_{STBY}$, $V_{STBYON}$, $V_{OH1}$, $V_{DF}$, and $I_{IDLE}$ from Table 45 and Table 46. Removed $V_{STBYON}$ timings tables. Updated disclaimer text. |

Obsolete Product(s) - Obsolete Product(s)
Obsolete Product(s) - Obsolete Product(s)

Obsolete Product(s) - Obsolete Product(s)

**Please Read Carefully:**

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries (“ST”) rese right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein time, without notice.
All ST products are sold pursuant to ST’s terms and conditions of sale.
Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assu liability whatsoever relating to the choice, selection or use of the ST products and services described herein.
No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any par document refers to any third party products or services shall not be deemed a license grant by ST for the use of such third party p or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever third party products or services or any intellectual property contained therein.

**UNLESS OTHERWISE SET FORTH IN ST’S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IM WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IM WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGH UNLESS EXPRESSLY APPROVED IN WRITING BY AN AUTHORIZED ST REPRESENTATIVE, ST PRODUCTS ARE RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, LIFE SAVING, OR SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESU PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. ST PRODUCTS WHICH ARE SPECIFIED AS "AUTOMOTIVE GRADE" MAY ONLY BE USED IN AUTOMOTIVE APPLICATIONS AT USER’S OWN RISK.**

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediat any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoev liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.

Information in this document supersedes and replaces all information previously supplied.

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

© 2008 STMicroelectronics - All rights reserved

STMicroelectronics group of companies
Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Ja Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America
**www.st.com**

Общество с ограниченной ответственностью «МосЧип» ИНН 7719860671 / КПП 771901001
Адрес: 105318, г.Москва, ул.Щербаковская д.3, офис 1107

**Данный компонент на территории Российской Федерации**

**Вы можете приобрести в компании MosChip.**

Для оперативного оформления запроса Вам необходимо перейти по данной ссылке:

http://moschip.ru/get-element

Вы можете разместить у нас заказ для любого Вашего проекта, будь то серийное производство или разработка единичного прибора.

В нашем ассортименте представлены ведущие мировые производители активных и пассивных электронных компонентов.

Нашей специализацией является поставка электронной компонентной базы двойного назначения, продукции таких производителей как XILINX, Intel (ex.ALTERA), Vicor, Microchip, Texas Instruments, Analog Devices, Mini-Circuits, Amphenol, Glenair.

Сотрудничество с глобальными дистрибьюторами электронных компонентов, предоставляет возможность заказывать и получать с международных складов практически любой перечень компонентов в оптимальные для Вас сроки.

На всех этапах разработки и производства наши партнеры могут получить квалифицированную поддержку опытных инженеров.

Система менеджмента качества компании отвечает требованиям в соответствии с ГОСТ Р ИСО 9001, ГОСТ РВ 0015-002 и ЭС РД 009

**Офис по работе с юридическими лицами:**

105318, г.Москва, ул.Щербаковская д.3, офис 1107, 1118, ДЦ «Щербаковский»

Телефон: +7 495 668-12-70 (многоканальный)

Факс: +7 495 668-12-70 (доб.304)

E-mail: info@moschip.ru

Skype отдела продаж:
moschip.ru
moschip.ru_4
moschip.ru_6
moschip.ru_9