PSD835G2

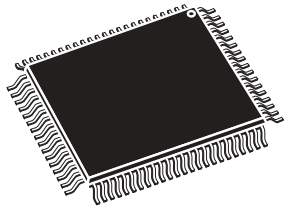
Flash PSD, 5V Supply, for 8-bit MCUs
4 Mbit + 256 Kbit Dual Flash Memories and 64 Kbit SRAM

FEATURES SUMMARY

- FLASH IN-SYSTEM PROGRAMMABLE (ISP) PERIPHERAL FOR 8-BIT MCUS
- DUAL BANK FLASH MEMORIES
  - 4 Mbits of Primary Flash Memory (8 uniform sectors, 64Kbyte)
  - 256 Kbits of Secondary Flash Memory with 4 sectors
  - Concurrent operation: READ from one memory while erasing and writing the other
- 64 KBIT OF BATTERY-BACKED SRAM
- 52 RECONFIGURABLE I/O PORTS
- ENHANCED JTAG SERIAL PORT
- PLD WITH MACROCELLS
  - Over 3000 Gates of PLD: CPLD and DPLD
  - CPLD with 16 Output Macrocells (OMCs) and 24 Input Macrocells (IMCs)
  - DPLD - user defined internal chip select decoding
- 52 INDIVIDUALLY CONFIGURABLE I/O PORT PINS

  They can be used for the following functions:
  - MCU I/Os
  - PLD I/Os
  - Latched MCU address output
  - Special function I/Os.
  - I/O ports may be configured as open-drain outputs.
- IN-SYSTEM PROGRAMMING (ISP) WITH JTAG
  - Built-in JTAG compliant serial port allows full-chip In-System Programmability
  - Efficient manufacturing allow easy product testing and programming
  - Use low cost FlashLINK cable with PC
- 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. Package



TQFP80 (U)

- HIGH ENDURANCE:
  - 100,000 Erase/WRITE Cycles of Flash Memory
  - 1,000 Erase/WRITE Cycles of PLD
  - 15 Year Data Retention
- 5V±10% SINGLE SUPPLY VOLTAGE
- STANDBY CURRENT AS LOW AS 50µA
- MEMORY SPEED
  - 70ns Flash memory and SRAM access time for V_{CC} = 4.5V to 5.5V
  - 90ns Flash memory and SRAM access time for V_{CC} = 4.5V to 5.5V

TABLE OF CONTENTS

SUMMARY DESCRIPTION

The PSD family of memory systems for microcontrollers (MCUs) 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.

The CPLD in the PSD devices features an optimized macrocell logic architecture. The PSD 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 the PSD Flash memory while the PSD is soldered to the circuit board: In-System Programming (ISP) via JTAG, and In-Application Programming (IAP).

In-System Programming (ISP) via JTAG

An IEEE 1149.1 compliant JTAG In-System Programming (ISP) interface is included on the PSD enabling the entire device (Flash memories, PLD, 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 when 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 memory the very first time? JTAG is the answer. Program the blank PSD 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 the 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 Flash memory arrays are included so that the MCU can execute code from one while erasing and programming the other. Robust product firmware updates in the field are possible over any communications 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 Flash memory blocks concurrently, reading code from one while erasing and programming the other during IAP.

Complex memory mapping. How can I map these two memories efficiently? A programmable Decode PLD (DPLD) is embedded in the PSD. 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 MCU address limit.

Separate Program and Data space. How can I write to Flash memory while it resides in Program space during field firmware updates? My 80C51 will not allow it. The PSD provides means to reclassify Flash memory as Data space during IAP, then back to Program space when complete.

PSDsoft

PSDsoft, 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 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: FlashLINK (JTAG) and PSDpro.

Figure 2. TQFP80 Connections

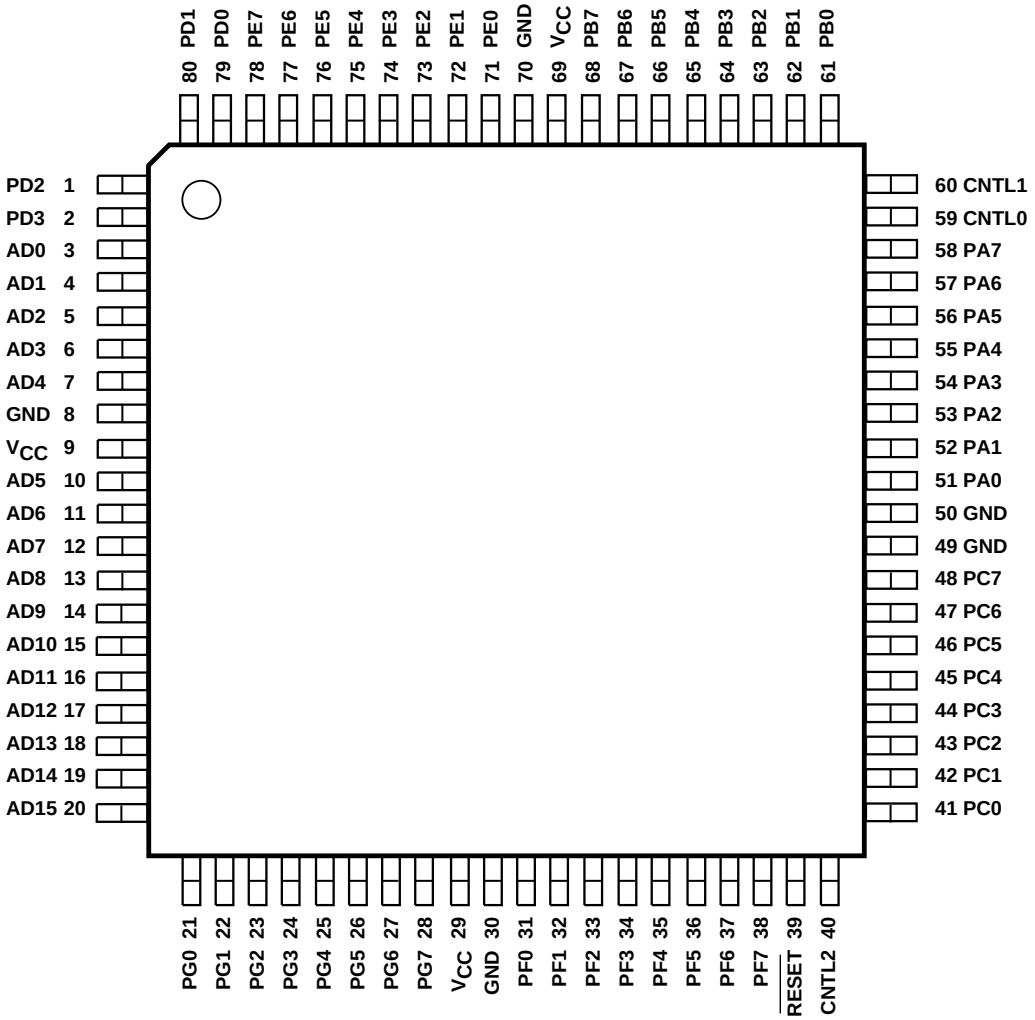


Table 1. Pin Description

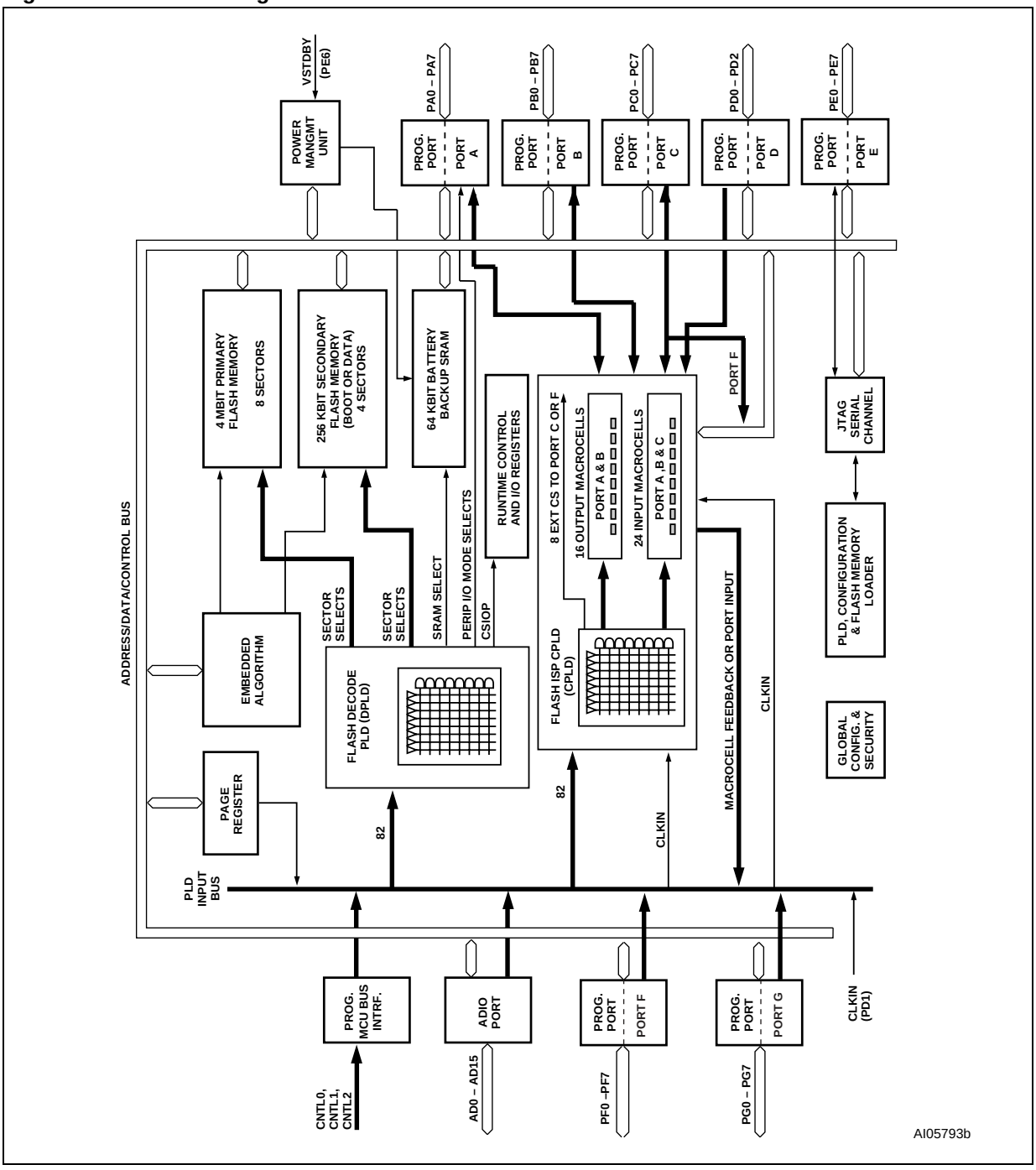
Pin Name	Pin	Type	Description
ADIO0-7	3-7-10-12	I/O	This is the lower Address/Data port. Connect your MCU address or address/data bus according to the following rules: If your MCU has a multiplexed address/data bus where the data is multiplexed with the lower address bits, connect AD0-AD7 to this port. If your MCU does not have a multiplexed address/data bus, connect A0-A7 to this port. If you are using an 80C51XA in burst mode, connect A4/D0 through A11/D7 to this port. 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	13-20	I/O	This is the upper Address/Data port. Connect your MCU address or address/data bus according to the following rules: 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. If your MCU does not have a multiplexed address/data bus, connect A8-A15 to this port. If you are using an 80C251 in page mode, connect AD8-AD15 to this port. If you are using an 80C51XA in burst mode, connect A12-A19 to this port. 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	59	I	The following control signals can be connected to this port, based on your MCU: $\overline{WR}$ – active Low Write Strobe input. R_$\overline{W}$ – active High READ/active Low WRITE input. This port is connected to the PLDs. Therefore, these signals can be used in decode and other logic equations.
CNTL1	60	I	The following control signals can be connected to this port, based on your MCU: $\overline{RD}$ – active Low Read Strobe input. E – E clock input. $\overline{DS}$ – active Low Data Strobe input. $\overline{PSEN}$ – connect $\overline{PSEN}$ to this port when it is being used as an active Low READ signal. For example, when the 80C251 outputs more than 16 address bits, $\overline{PSEN}$ is actually the READ signal. This port is connected to the PLDs. Therefore, these signals can be used in decode and other logic equations.
CNTL2	40	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 as input.

Pin Name	Pin	Type	Description
Reset	39	I	Active Low input. Resets I/O Ports, PLD macrocells and some of the Configuration Registers and JTAG registers. Must be Low at Power-up. Reset also aborts the Flash programming/erase cycle that is in progress.
PA0 PA1 PA2 PA3 PA4 PA5 PA6 PA7	58 57 56 55 54 53 52 51	I/O CMOS or Open Drain	These pins make up Port A. These port pins are configurable and can have the following functions: MCU I/O – write to or read from a standard output or input port. CPLD macrocell (McellA0-7) outputs. Inputs to the PLDs. Latched, transparent or registered PLD input.
PB0 PB1 PB2 PB3 PB4 PB5 PB6 PB7	68 67 66 65 64 63 62 61	I/O CMOS or Open Drain	These pins make up Port B. These port pins are configurable and can have the following functions: MCU I/O – write to or read from a standard output or input port. CPLD macrocell (McellB0-7) output. Inputs to the PLDs. Latched, transparent or registered PLD input.
PC0 PC1 PC2 PC3 PC4 PC5 PC6 PC7	48 47 46 45 44 43 42 41	I/O CMOS or Open Drain	These pins make up Port C. These port pins are configurable and can have the following functions: MCU I/O – write to or read from a standard output or input port. External Chip Select (ECS0-7) output. Latched, transparent or registered PLD input.
PD0	79	I/O CMOS or Open Drain	PD0 pin of Port D. This port pin can be configured to have the following functions: ALE/AS input latches addresses on ADIO0-ADIO15 pins. AS input latches addresses on ADIO0-ADIO15 pins on the rising edge. Input to the PLDs. Transparent PLD input.
PD1	80	I/O CMOS or Open Drain	PD1 pin of Port D. This port pin can be configured to have the following functions: MCU I/O – write to or read from a standard output or input port. Input to the PLDs. CLKIN – clock input to the CPLD macrocells, the APD Unit's Power-down counter, and the CPLD AND Array.
PD2	1	I/O CMOS or Open Drain	PD2 pin of Port D. This port pin can be configured to have the following functions: MCU I/O – write to or read from a standard output or input port. Input to the PLDs. PSD Chip Select Input (CSI). When Low, the MCU can access the PSD memory and I/O. When High, the PSD memory blocks are disabled to conserve power. The trailing edge of CSI can be used to get the PSD out of power-down mode.

Pin Name	Pin	Type	Description
PD3	2	I/O CMOS or Open Drain	PD3 pin of Port D. This port pin can be configured to have the following functions: MCU I/O – write to or read from a standard output or input port. Input to the PLDs.
PE0	71	I/O CMOS or Open Drain	PE0 pin of Port E. This port pin can be configured to have the following functions: MCU I/O – write to or read from a standard output or input port. Latched address output. TMS input for JTAG/ISP interface.
PE1	72	I/O CMOS or Open Drain	PE1 pin of Port E. This port pin can be configured to have the following functions: MCU I/O – write to or read from a standard output or input port. Latched address output. TCK input for JTAG/ISP interface (Schmidt Trigger).
PE2	73	I/O CMOS or Open Drain	PE2 pin of Port E. This port pin can be configured to have the following functions: MCU I/O – write to or read from a standard output or input port. Latched address output. TDI input for JTAG/ISP interface.
PE3	74	I/O CMOS or Open Drain	PE3 pin of Port E. This port pin can be configured to have the following functions: MCU I/O – write to or read from a standard output or input port. Latched address output. TDO input for JTAG/ISP interface.
PE4	75	I/O CMOS or Open Drain	PE4 pin of Port E. This port pin can be configured to have the following functions: MCU I/O – write to or read from a standard output or input port. Latched address output. TSTAT input for the ISP interface. Ready/Busy for in-circuit Parallel Programming.
PE5	76	I/O CMOS or Open Drain	PE5 pin of Port E. This port pin can be configured to have the following functions: MCU I/O – write to or read from a standard output or input port. Latched address output. TERR active Low input for ISP interface.
PE6	77	I/O CMOS or Open Drain	PE6 pin of Port E. This port pin can be configured to have the following functions: MCU I/O – write to or read from a standard output or input port. Latched address output. V_{STBY} SRAM standby voltage input for battery backup SRAM.

Pin Name	Pin	Type	Description
PE7	78	I/O CMOS or Open Drain	PE7 pin of Port E. This port pin can be configured to have the following functions: MCU I/O – write to or read from a standard output or input port. Latched address output. V_{BATON} battery backup indicator output. Goes High when power is drawn from an external battery.
PF0-PF7	31-38	I/O CMOS or Open Drain	PF0 through PF7 pins of Port F. This port pins can be configured to have the following functions: MCU I/O – write to or read from a standard output or input port. Input to the PLDs. Latched address outputs. As address A0-A3 inputs in 80C51XA mode. As data bus port (D07) in non-multiplexed bus configuration.
PG0-PG7	8, 30, 49, 50, 70	I/O CMOS or Open Drain	PG0 through PG7 pins of Port G. This port pins can be configured to have the following functions: MCU I/O – write to or read from a standard output or input port. Latched address outputs.
V_{CC}	9, 29, 69		Supply Voltage
GND	8, 30, 49, 50, 70		Ground pins

Figure 3. PSD Block Diagram



PSD ARCHITECTURAL OVERVIEW

PSD devices contain several major functional blocks. Figure 3., page 15 shows the architecture of the PSD device family. The functions of each block are described briefly in the following sections. Many of the blocks perform multiple functions and are user configurable.

Memory

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 24. The 4 Mbit (512K x 8) Flash memory is the primary memory of the PSD. It is divided into 8 equally-sized sectors that are individually selectable.

The 256 Kbit (32K x 8) secondary Flash memory is divided into 4 equally-sized sectors. Each sector is individually selectable.

The 64 Kbit SRAM is intended for use as a scratch-pad memory or as an extension to the MCU SRAM. If an external battery is connected to Voltage Standby (V_{STBY}, PC2), data is retained in the event of power failure.

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.

Page Register

The 8-bit Page Register expands the address range of the MCU 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 sectors of the Flash memories into different memory spaces for IAP.

PLDs

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

The DPLD is used to decode addresses and to generate Sector Select signals 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 (OMC) and 8 combinatorial outputs. The PSD also has 24 Input Macrocells (IMC) 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 Power-Management design techniques. The speed and power consumption of the PLD is controlled by the Turbo Bit in PMMR0 and other bits in the PMMR2. These registers are set by the MCU at run-time. There is a slight penalty to PLD propagation time when invoking the power management features.

I/O Ports

The PSD has 52 I/O pins distributed over the seven ports (Port A, B, C, D, E, F and G). Each I/O pin can be individually configured for different functions. Ports can be configured as standard MCU I/O ports, PLD I/O, or latched address outputs for MCUs using multiplexed address/data buses.

The JTAG pins can be enabled on Port E for In-System Programming (ISP). Ports F and G can also be configured as data ports for a non-multiplexed bus.

Ports A and B can also be configured as a data port for a non-multiplexed bus.

MCU Bus Interface

PSD interfaces easily with most 8-bit MCUs that have either multiplexed or non-multiplexed address/data buses. The device is configured to respond to the MCU's control signals, which are also used as inputs to the PLDs. For examples, please see MCU Bus Interface Examples, page 52.

Table 2. PLD I/O

Name	Inputs	Outputs	Product Terms
Decode PLD (DPLD)	82	17	43
Complex PLD (CPLD)	82	24	150

Table 3. JTAG SIgnals on Port E

Port E Pins	JTAG Signal
PE0	TMS
PE1	TCK
PE2	TDI
PE3	TDO
PE4	TSTAT
PE5	TERR

JTAG Port

In-System Programming (ISP) can be performed through the JTAG signals on Port E. 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 E. Table 3., page 16 indicates the JTAG pin assignments.

In-System Programming (ISP)

Using the JTAG signals on Port E, the entire PSD device (memory, logic, configuration) can be programmed or erased without the use of the MCU.

In-Application re-Programming (IAP)

The primary Flash memory can also be programmed in-system by the MCU executing the programming algorithms out of the secondary memory, or SRAM. Since this is a sizable separate block, the application can also continue to operate. The secondary memory can be programmed the same way by executing out of the primary Flash memory. The PLD or other PSD Configuration blocks 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.

Power Management Unit (PMU)

The Power Management Unit (PMU) gives the user control of the power consumption on selected functional blocks based on system requirements. The PMU includes an Automatic Power-down (APD) Unit that turns off device functions during MCU 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 PMMR0 can be reset to '0' and the CPLD latches its outputs and goes to sleep until the next transition on its inputs.

Additionally, bits in PMMR2 can be set by the MCU to block signals from entering the CPLD to reduce power consumption. Please see POWER MANAGEMENT, page 70 for more details.

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

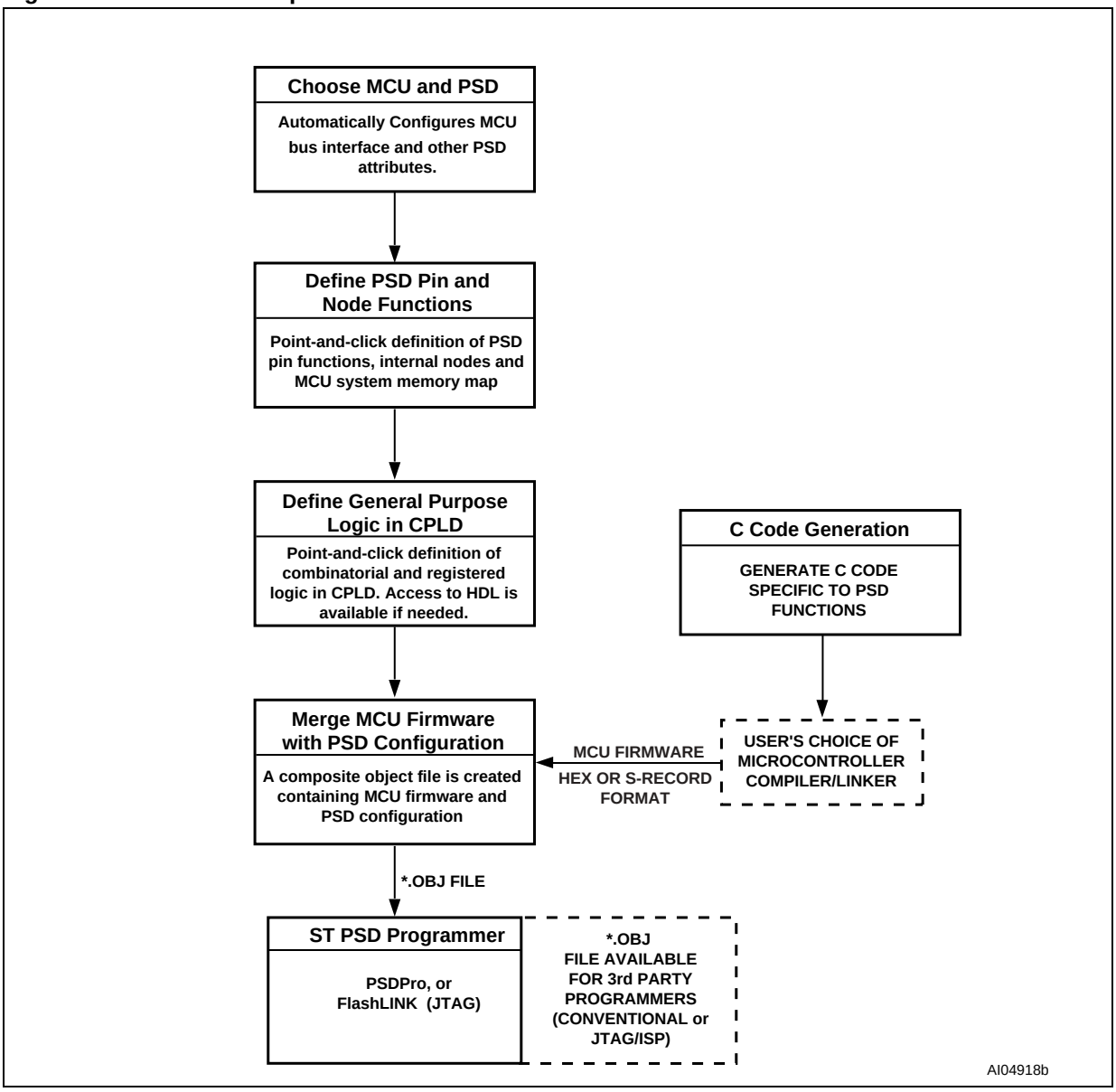
Functional Block	JTAG/ISP	Device Programmer	IAP
Primary Flash Memory	Yes	Yes	Yes
Secondary Flash Memory	Yes	Yes	Yes
PLD Array (DPLD and CPLD)	Yes	Yes	No
PSD Configuration	Yes	Yes	No

DEVELOPMENT SYSTEM

The PSD family is supported by PSDsoft, 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 Description Language (HDL) equations, unless desired, to define PSD pin functions and memory map information. The general design flow is shown in Figure 4. PSDsoft is available from our web site (the address is given on the back page of this data sheet) or other distribution channels.

PSDsoft directly supports two low cost device programmers form 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 our web site for the current list.

Figure 4. PSDsoft Development Tool



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

Table 5. Register Address Offset

Register Name	Port A	Port B	Port C	Port D	Port E	Port F	Port G	Other [1]	Description
Data In	00	01	10	11	30	40	41		Reads Port pin as input, MCU I/O input mode
Control					32	42	43		Selects mode between MCU I/O or Address Out
Data Out	04	05	14	15	34	44	45		Stores data for output to Port pins, MCU I/O output mode
Direction	06	07	14	15	36	46	47		Configures Port pin as input or output
Drive Select	08	09	18	19	38	48	49		Configures Port pins as either CMOS or Open Drain on some pins, while selecting high slew rate on other pins.
Input Macrocell	0A	0B		1A					Reads Input Macrocells
Enable Out	0C	0D	1C	1B		4C			Reads the status of the output enable to the I/O Port driver
Output Macrocells A	20								READ – reads output of macrocells A WRITE – loads macrocell flip-flops
Output Macrocells B		21							READ – reads output of macrocells B WRITE – loads macrocell flip-flops
Mask Macrocells A	22								Blocks writing to the Output Macrocells A
Mask Macrocells B		23							Blocks writing to the Output Macrocells B
Primary Flash Protection								C0	Read only – Primary Flash Sector Protection
Secondary Flash memory Protection								C2	Read only – PSD Security and Secondary Flash memory 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.
Memory_ID0								F0	Read only – Primary Flash memory and SRAM size
Memory_ID1								F1	Read only – Secondary Flash memory type and size

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

REGISTER BIT DEFINITION

All the registers of the PSD are included here, for reference. Detailed descriptions of these registers can be found in the following sections.

Table 6. Data-In Registers – Ports A, B, C, D, E, F, G

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

Note: Bit Definitions (Read only registers):
Read Port pin status when Port is in MCU I/O input mode.

Table 7. Data-Out Registers – Ports A, B, C, D, E, F, G

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

Note: Bit Definitions:
Latched data for output to Port pin when pin is configured in MCU I/O output mode.

Table 8. Direction Registers – Ports A, B, C, D, E, F, G

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

Note: Bit Definitions:
Port pin <**i**> 0 = Port pin <i> is configured in Input mode (default).
Port pin <**i**> 1 = Port pin <i> is configured in Output mode.

Table 9. Control Registers – Ports E, F, G

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

Note: Bit Definitions:
Port pin <**i**> 0 = Port pin <i> is configured in MCU I/O mode (default).
Port pin <**i**> 1 = Port pin <i> is configured in Latched Address Out mode.

Table 10. Drive Registers – Ports A, B, D, E, G

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

Note: Bit Definitions:
Port pin <**i**> 0 = Port pin <i> is configured for CMOS Output driver (default).
Port pin <**i**> 1 = Port pin <i> is configured for Open Drain output driver.

Table 11. Drive Registers – Ports C, F

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

Note: Bit Definitions:
Port pin <**i**> 0 = Port pin <i> is configured for CMOS Output driver (default).
Port pin <**i**> 1 = Port pin <i> is configured in Slew Rate mode.

Table 12. Enable-Out Registers – Ports A, B, C, F

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

Note: Bit Definitions (Read only registers):
Port pin <i> 0 = Port pin <i> is in tri-state driver (default).
Port pin <i> 1 = Port pin <i> is enabled.

Table 13. Input Macrocells – Ports A, B, C

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

Note: Bit Definitions (Read only registers):
Read Input Macrocell (IMC7-IMC0) status on Ports A, B and C.

Table 14. Output Macrocells A Register

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

Note: Bit Definitions:
Write Register: Load MCellA7-MCellA0 with '0' or '1.'
Read Register: Read MCellA7-MCellA0 output status.

Table 15. Output Macrocells B Register

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

Note: Bit Definitions:
Write Register: Load MCellB7-MCellB0 with '0' or '1.'
Read Register: Read MCellB7-MCellB0 output status.

Table 16. Mask Macrocells A Register

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

Note: Bit Definitions:
McellA<i>**_Prot** 0 = Allow MCellA<i> flip-flop to be loaded by MCU (default).
McellA<i>**_Prot** 1 = Prevent MCellA<i> flip-flop from being loaded by MCU.

Table 17. Mask Macrocells B Register

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

Note: Bit Definitions:
McellB<i>**_Prot** 0 = Allow MCellB<i> flip-flop to be loaded by MCU (default).
McellB<i>**_Prot** 1 = Prevent MCellB<i> flip-flop from being loaded by MCU.

Table 18. Flash Memory Protection Register

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

Note: Bit Definitions (Read only register):
Sec<i>**_Prot** 1 = Primary Flash memory Sector <i> is write protected.
Sec<i>**_Prot** 0 = Primary Flash memory Sector <i> is not write protected.

Table 19. Flash Boot Protection Register

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

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

Table 20. JTAG Enable Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
not used	not used	not used	not used	not used	not used	not used	JTAGEnable

Note: Bit Definitions:
JTAG_Enable 1 = JTAG Port is enabled.
JTAG_Enable 0 = JTAG Port is disabled.

Table 21. Page Register

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

Note: Bit Definitions:
Configure Page input to PLD. Default is PGR7-PGR0=00.

Table 22. PMMR0 Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
not used (set to '0')	not used (set to '0')	PLD MCells CLK	PLD Array CLK	PLD Turbo	not used (set to '0')	APD Enable	not used (set to '0')

Note: 1. The bits of this register are cleared to zero following Power-up. Subsequent Reset ($\overline{\text{Reset}}$) pulses do not clear the registers.
2. Bit Definitions:
APD Enable0 = Automatic Power-down (APD) is disabled.
1 = Automatic Power-down (APD) is enabled.
PLD Turbo0 = PLD Turbo is on.
1 = PLD Turbo is off, saving power.
PLD Array CLK0 = CLKIN to the PLD AND array is connected. Every CLKIN change powers up the PLD when Turbo Bit is off.
1 = CLKIN to the PLD AND array is disconnected, saving power.
PLD MCells CLK0 = CLKIN to the PLD Macrocells is connected.
1 = CLKIN to the PLD Macrocells is disconnected, saving power.

Table 23. PMMR2 Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
not used (set to '0')	PLD Array WRH	PLD Array ALE	PLD Array CNTL2	PLD Array CNTL1	PLD Array CNTL0	not used (set to '0')	PLD Array Addr

Note: Bit Definitions:
PLD Array Addr 0 = Address A7-A0 are connected to the PLD array.
1 = Address A7-A0 are blocked from the PLD array, saving power.
(Note: in XA mode, A3-A0 come from PF3-PF0, and A7-A4 come from ADIO7-ADIO4)
PLD Array CNTL2 0 = CNTL2 input to the PLD AND array is connected.
1 = CNTL2 input to the PLD AND array is disconnected, saving power.
PLD Array CNTL1 0 = CNTL1 input to the PLD AND array is connected.
1 = CNTL1 input to the PLD AND array is disconnected, saving power.
PLD Array CNTL0 0 = CNTL0 input to the PLD AND array is connected.
1 = CNTL0 input to the PLD AND array is disconnected, saving power.
PLD Array ALE 0 = ALE input to the PLD AND array is connected.
1 = ALE input to the PLD AND array is disconnected, saving power.
PLD Array WRH 0 = WRH/DBE input to the PLD AND array is connected.
1 = WRH/DBE input to the PLD AND array is disconnected, saving power.

Table 24. VM Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Peripheral mode	not used (set to '0')	not used (set to '0')	FL_data	Boot_data	FL_code	Boot_code	SR_code

Note: 1. On reset, Bit1-Bit4 are loaded to configurations that are selected by the user in PSDsoft. Bit0 and Bit7 are always cleared on reset. Bit0-Bit4 are active only when the device is configured in Philips 80C51XA mode.

2. Bit Definitions:
SR_code0 = $\overline{\text{PSEN}}$ cannot access SRAM in 80C51XA modes.
1 = $\overline{\text{PSEN}}$ can access SRAM in 80C51XA modes.
Boot_code0 = $\overline{\text{PSEN}}$ cannot access Secondary NVM in 80C51XA modes.
1 = $\overline{\text{PSEN}}$ can access Secondary NVM in 80C51XA modes.
FL_code0 = $\overline{\text{PSEN}}$ cannot access Primary Flash memory in 80C51XA modes.
1 = $\overline{\text{PSEN}}$ can access Primary Flash memory in 80C51XA modes.
Boot_data0 = $\overline{\text{RD}}$ cannot access Secondary NVM in 80C51XA modes.
1 = $\overline{\text{RD}}$ can access Secondary NVM in 80C51XA modes.
FL_data0 = $\overline{\text{RD}}$ cannot access Primary Flash memory in 80C51XA modes.
1 = $\overline{\text{RD}}$ can access Primary Flash memory in 80C51XA modes.
Peripheral mode0 = Peripheral mode of Port F is disabled.
1 = Peripheral mode of Port F is enabled.

Table 25. Memory_ID0 Register

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

Note: Bit Definitions:
F_size[3:0] 4h = Primary Flash memory size is 4 Mbit
5h = Primary Flash memory size is 8Mbit
S_size[3:0] 0h = There is no SRAM
1h = SRAM size is 16 Kbit
3h = SRAM size is 64 Kbit

Table 26. Memory_ID1 Register

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
not used (set to '0')	not used (set to '0')	B_type 1	B_type 0	B_size 3	B_size 2	B_size 1	B_size 0

Note: Bit Definitions:
B_size[3:0] 0h = There is no Secondary NVM
2h = Secondary NVM size is 256 Kbit
B_type[1:0] 0h = Secondary NVM is Flash memory
1h = Secondary NVM is EEPROM

DETAILED OPERATION

As shown in Figure 3., page 15, 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/ISP 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:

– Primary Flash memory
– Secondary Flash memory
– SRAM

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

Table 27. Memory Block Size and Organization

	Primary Flash Memory		Secondary Flash Memory		SRAM	
Sector Number	**Sector Size (Bytes)**	**Sector Select Signal**	**Sector Size (Bytes)**	**Sector Select Signal**	**SRAM Size (Bytes)**	**SRAM Select Signal**
0	64K	FS0	8K	CSBOOT0	16K	RS0
1	64K	FS1	8K	CSBOOT1		
2	64K	FS2	8K	CSBOOT2		
3	64K	FS3	8K	CSBOOT3		
4	64K	FS4				
5	64K	FS5				
6	64K	FS6				
7	64K	FS7				
Total	512K	8 Sectors	32K	4 Sectors	16K	

Primary Flash Memory and Secondary Flash memory Description

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

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

During a Program or Erase cycle in Flash memory, the status can be output on Ready/Busy (PE4). This pin is set up using PSDsoft.

Memory Block Select Signals

The DPLD generates the Select signals for all the internal memory blocks (see PLDs, page 38). Each of the eight sectors of the primary Flash memory has a Select signal (FS0-FS7) which can contain up to three product terms. Each of the four sectors of the secondary Flash memory has a Select signal (CSBOOT0-CSBOOT3) which can contain up to three product terms. Having three product terms for each Select signal allows a given sector to be mapped in different areas of system memory. When using an MCU with separate Program and Data space, these flexible Select signals allow dynamic re-mapping of sectors from one memory space to the other before and after IAP.

Upper and Lower Block IN MAIN FLASH SECTOR

The PSD835G2's main Flash memory has eight 64-KByte sectors. The 64-KByte sector size may cause some difficulty in code mapping for an 8-bit MCU with only 64-KByte address space. To resolve this mapping issue, the PSD835G2 provides additional logic (see Figure 6., page 26) for the user to split the 8 sectors such that each sector has a lower and upper 32-KByte block, and the two blocks can reside in different pages but in the same address range.

If your design works with 64KB sectors, you don't need to configure this logic. If the design requires 32KB blocks in each sector, you need to define a "FA15" PLD equation in PSDsoft as the A15 address input to the main Flash module. FA15 consists of 3 product terms and will control whether the MCU is accessing the lower or upper 32KB in the selected sector. Figure 4 shows an example for Flash sector chip select FS0. A typical equation is FA15 = pgr4 of the Page Register. When pgr4 is 0 (page 00), the lower 32KB is selected. When pgr4 is switched to '1' by the user, the upper 32KB is selected. PSDsoft will automatically generate the PLD equations shown, based on your point and click selections.

If no FA15 equation is defined in PSDsoft, the A15 that comes from the MCU address bus will be routed as input to the primary Flash memory instead of FA15. The FA15 equation has no impact on the Sector Erase operation.

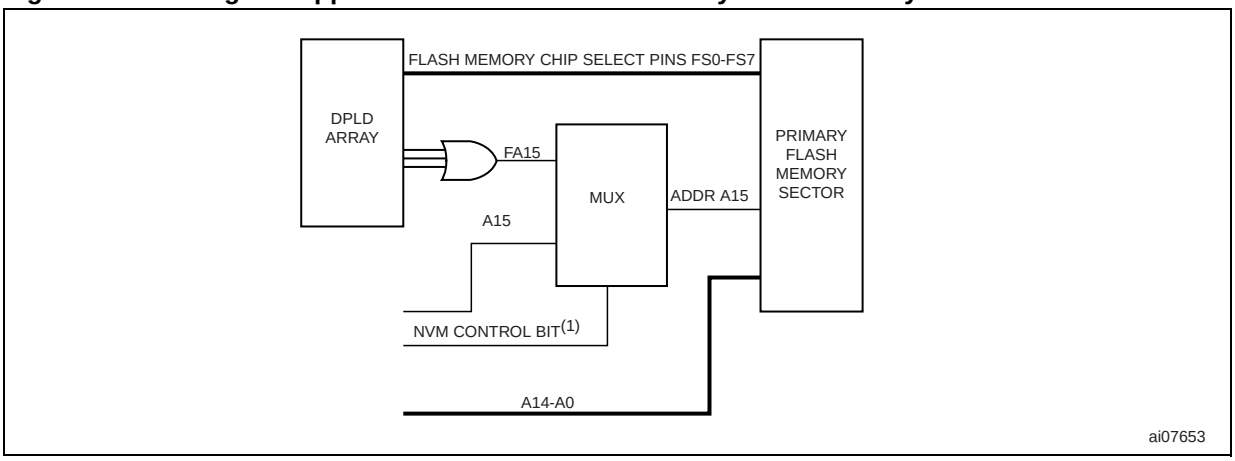
Note: FA15 affects all eight sectors of the primary Flash memory simultaneously. You cannot direct FA15 to a particular Flash sector only.

Figure 5. Example for Flash Sector Chip Select FS0

```
page = [pgr7... pgr0]; "Page Register output
 "Sector Chip Select Equation
 FS0 = ((0000h <= address <= 7FFFh) & page = 00h) #        "select first 32KB block
        ((0000h <= address <= 7FFFh) & page = 10h);        "select second 32KB block
 FA15 = pgr4;              "as address A15 input to the primary Flash memory
```

ai07652

Figure 6. Selecting the Upper or Lower Block in a Primary Flash Memory Sector



Ready/$\overline{\text{Busy}}$ (PE4)

This signal can be used to output the Ready/$\overline{\text{Busy}}$ status of the PSD. The output on Ready/Busy (PE4) is a ’0’ (Busy) when Flash memory blocks are being written to, *or* when the Flash memory block is being erased. The output is a '1' (Ready) when no WRITE or Erase cycle is in progress.

Memory Operation

The primary Flash memory and secondary Flash memory are addressed through the MCU Bus Interface. The MCU can access these memories in one of two ways:

– The MCU 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 MCU 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 memory to invoke an embedded algorithm. These instructions are summarized in Table Table 28., page 27.

Typically, the MCU can read Flash memory using READ operations, just as it would read a ROM device. However, Flash memory can only be altered using specific Erase and Program instructions. For example, the MCU cannot write a single byte directly to Flash memory as it would write a byte to RAM. To program a byte into Flash memory, the MCU must execute a Program instruction, then test the status of the Program cycle. This status test is achieved by a READ operation or polling Ready/Busy (PE4).

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

Table 28. Instructions

Instruction	FS0-FS7 or CSBOOT0-CSBOOT3	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5	Cycle 6	Cycle 7
Read(5)	1	"Read" RA @ RD						
Read Primary Flash ID(6,13)	1	AAh@ 555h	55h@ AAAh	90h@ 555h	Read identifier @X01h			
Read Sector Protection(6,8,13)	1	AAh@ 555h	55h@ AAAh	90h@ 555h	Read identifier 00h or 01h @X02h			
Program a Flash Byte(13)	1	AAh@ 555h	55h@ AAAh	A0h@ 555h	PD@ PA			
Flash Sector Erase(7)	1	AAh@ 555h	55h@ AAAh	80h@ 555h	AAh@ 555h	55h@ AAAh	30h@ SA	30h(7)@ next SA
Flash Bulk Erase	1	AAh@ 555h	55h@ AAAh	80h@ 555h	AAh@ 555h	55h@ AAAh	10h@ 555h	
Suspend Sector Erase(11)	1	B0h@ XXXh						
Resume Sector Erase(12)	1	30h@ XXXh						
Reset(6)	1	F0h@ any address						
Unlock Bypass	1	AAh@ 555h	55h@ AAAh	20h@ 555h				
Unlock Bypass Program(9)	1	A0h@ XXXh	PD@ PA					
Unlock Bypass Reset(10)	1	90h@ XXXh	00h@ XXXh					

Note: 1. All bus cycles are WRITE bus cycles, except the ones with the "Read" label
2. All values are in hexadecimal:
X = Don't Care.
RA = Address of the memory location to be read
RD = Data read from location RA during the READ cycle
PA = Address of the memory location to be programmed. Addresses are latched on the falling edge of Write Strobe ($\overline{WR}$, CNTL0). PA is an even address for PSD in word programming mode.
PD = Data to be programmed at location PA. Data is latched on the rising edge of Write Strobe ($\overline{WR}$, CNTL0)
SA = Address of the sector to be erased or verified. The Sector Select pins (FS0-FS7 or CSBOOT0-CSBOOT3) of the sector or whole memory to be erased, or verified, must be Active (High).
3. Sector Select (FS0-FS7 or CSBOOT0-CSBOOT3) signals are active High, and are defined in PSDsoft.
4. Only address Bits A11-A0 are used in instruction decoding. A15-A12 (or A16-A12) are don't care.
5. No Unlock or instruction cycles are required when the device is in the READ mode
6. The Reset instruction is required to return to the READ mode after reading the Flash ID, or after reading the Sector Protection Status, or if the Error Flag Bit (DQ5/DQ13) goes High.
7. Additional sectors to be erased must be written at the end of the Sector Erase instruction within 80µs.
8. The data is 00h for an unprotected sector, and 01h for a protected sector. In the fourth cycle, the Sector Select is active, and (A1,A0)=(1,0)
9. The Unlock Bypass instruction is required prior to the Unlock Bypass Program instruction.
10. The Unlock Bypass Reset Flash instruction is required to return to reading memory data when the device is in the Unlock Bypass mode.
11. The system may perform READ and Program cycles in non-erasing sectors, read the Flash ID or read the Sector Protection Status when in the Suspend Sector Erase mode. The Suspend Sector Erase instruction is valid only during a Sector Erase cycle.
12. The Resume Sector Erase instruction is valid only during the Suspend Sector Erase mode.
13. The MCU cannot invoke these instructions while executing code from the same Flash memory as that for which the instruction is intended. The MCU must fetch, for example, the code from the secondary Flash memory when reading the Sector Protection Status of the primary Flash memory.

INSTRUCTIONS

An instruction consists of 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 is properly received and the time between two consecutive Bytes is shorter than the time-out period. Some instructions are structured to include READ operations after the initial WRITE operations.

The instruction must be followed exactly. Any invalid combination of instruction Bytes or time-out between two consecutive bytes while addressing Flash memory resets the device logic into READ mode (Flash memory is read like a ROM device).

The PSD supports the instructions summarized in Table 28., page 27:

Flash memory:

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

These instructions are detailed in Table 28. For efficient decoding of the instructions, the first two Bytes of an instruction are the coded cycles and are followed by an instruction Byte or a confirmation Byte. The coded cycles consist in writing the data AAh to address X555h during the first cycle and data 55h to address XAAAh during the second cycle unless the Bypass Instruction feature is used). Address signals A15-A12 are Don't Care during the instruction WRITE cycles. However, the appropriate Sector Select (FS0-FS7 or CSBOOT0-CSBOOT3) must be selected.

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

Power-up Mode

The PSD internal logic is reset upon Power-up to the READ mode. Sector Select (FS0-FS7 and CSBOOT0-CSBOOT3) must be held Low, and Write Strobe (WR, CNTL0) High, 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 Write Strobe (WR, CNTL0). Any WRITE cycle initiation is locked when V_{CC} is below V_{LKO}.

READ

Under typical conditions, the MCU may read the primary Flash memory or the secondary Flash memory using READ operations just as it would a ROM or RAM device. Alternately, the MCU may use READ operations to obtain status information about a Program or Erase cycle that is currently in progress. Lastly, the MCU may use instructions to read special data from these memory blocks. The following sections describe these READ functions.

Read Memory Contents

Primary Flash memory and secondary Flash memory are placed in the READ mode after Power-up, chip reset, or a Reset Flash instruction (see Table 28). The MCU can read the memory contents of the primary Flash memory or the secondary Flash memory by using READ operations any time the READ operation is not part of an instruction.

Read Primary Flash Identifier

The primary Flash memory identifier is read with an instruction composed of 4 operations: 3 specific WRITE operations and a READ operation (see Table 28., page 27). The identifier for the device is E8h.

Read Memory Sector Protection Status

The primary Flash memory Sector Protection Status is read with an instruction composed of 4 operations: 3 specific WRITE operations and a READ operation (see Table 28). The READ operation produces 01h if the Flash memory sector is protected, or 00h if the sector is not protected.

The sector protection status for all NVM blocks (primary Flash memory or secondary Flash memory) can also be read by the MCU accessing the Flash Protection and Flash Boot Protection registers in PSD I/O space. See Flash Memory Sector Protect, page 34 for register definitions.

Read the Erase/Program Status Bits

The PSD provides several status bits to be used by the MCU to confirm the completion of an Erase or Program cycle of Flash memory. These status bits minimize the time that the MCU spends performing these tasks and are defined in Table 29. The status bits can be read as many times as needed.

For Flash memory, the MCU can perform a READ operation to obtain these status bits while an Erase or Program instruction is being executed by the embedded algorithm. See PROGRAMMING FLASH MEMORY, page 31 for details.

Table 29. Status Bit

Functional Block	FS0-FS7/CSBOOT0-CSBOOT3	DQ7	DQ6	DQ5	DQ4	DQ3	DQ2	DQ1	DQ0
Flash Memory	V_{IH}	Data Polling	Toggle Flag	Error Flag	X	Erase Time-out	X	X	X

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

Data Polling Flag (DQ7)

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

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

Toggle Flag (DQ6)

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

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

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

Error Flag (DQ5)

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

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

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

Erase Time-out Flag (DQ3)

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

PROGRAMMING FLASH MEMORY

Flash memory must be erased prior to being programmed. The MCU may erase Flash memory all at once or by-sector. A Flash memory sector is erased to all 1s (FFh), and is programmed by setting selected bits to '0.' Although Flash memory is erased by-sector, it is programmed Word-by-Word.

The primary and secondary Flash memories require the MCU to send an instruction to program a Word or to erase sectors (see Table 28., page 27).

Once the MCU issues a Flash memory Program or Erase instruction, it must check the status bits for 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 (PE4) output pin.

Data Polling

Polling on the Data Polling Flag Bit (DQ7) is a method of checking whether a Program or Erase cycle is in progress or has completed. Figure 7 shows the Data Polling algorithm.

When the MCU issues a Program instruction, the embedded algorithm within the PSD begins. The MCU then reads the location of the Word to be programmed in Flash memory to check status. The Data Polling Flag Bit (DQ7) of this location becomes the complement of b7 of the original data byte to be programmed. The MCU continues to poll this location, comparing the Data Polling Flag Bit (DQ7) and monitoring the Error Flag Bit (DQ5). When the Data Polling Flag Bit (DQ7) matches b7 of the original data, and the Error Flag Bit (DQ5) remains '0,' the embedded algorithm is complete. If the Error Flag Bit (DQ5) is '1,' the MCU should test the Data Polling Flag Bit (DQ7) again since the Data Polling Flag Bit (DQ7) may have changed simultaneously with the Error Flag Bit (DQ5, see Figure 7).

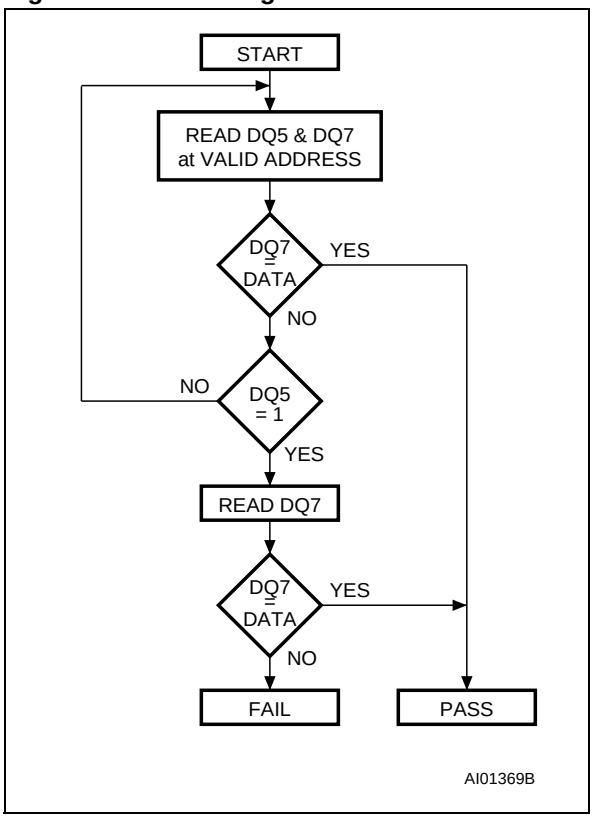
The Error Flag Bit (DQ5) is set if either an internal time-out occurred while the embedded algorithm attempted to program the Byte or if the MCU attempted to program a ’1’ to a bit that was not erased (not erased is logic 0).

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 the Flash memory with the Byte that was intended to be written.

When using the Data Polling method after an Erase cycle, Figure 7 still applies. However, the Data Polling Flag Bit (DQ7) is '0' until the Erase cycle is complete. A '1' on the Error Flag Bit (DQ5) indicates a time-out condition on the Erase cycle; a '0' indicates no error. The MCU can read any location within the sector being erased to get the Data Polling Flag Bit (DQ7) and the Error Flag Bit (DQ5).

PSDsoft generates ANSI C code functions which implement these Data Polling algorithms.

Figure 7. Data Polling Flowchart



Data Toggle

Checking the Toggle Flag Bit (DQ6) is a method of determining whether a Program or Erase cycle is in progress or has completed. Figure 8 shows the Data Toggle algorithm.

When the MCU issues a Program instruction, the embedded algorithm within the PSD begins. The MCU then reads the location of the byte to be programmed in Flash memory to check status. The Toggle Flag Bit (DQ6) of this location toggles each time the MCU reads this location until the embedded algorithm is complete. The MCU continues to read this location, checking the Toggle Flag Bit (DQ6) and monitoring the Error Flag Bit (DQ5). When the Toggle Flag Bit (DQ6) stops toggling (two consecutive READs yield the same value), and the Error Flag Bit (DQ5) remains '0,' the embedded algorithm is complete. If the Error Flag Bit (DQ5) is 1,' the MCU should test the Toggle Flag Bit (DQ6) again, since the Toggle Flag Bit (DQ6) may have changed simultaneously with the Error Flag Bit (DQ5, see Figure 8).

The Error Flag Bit (DQ5) is set if either an internal time-out 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 memory with the Byte that was intended to be written.

When using the Data Toggle method after an Erase cycle, Figure 8 still applies. the Toggle Flag Bit (DQ6) toggles until the Erase cycle is complete. A 1 on the Error Flag Bit (DQ5) indicates a time-out condition on the Erase cycle; a ’0’ indicates no error. The MCU can read any location within the sector being erased to get the Toggle Flag Bit (DQ6) and the Error Flag Bit (DQ5).

PSDsoft generates ANSI C code functions which implement these Data Toggling algorithms.

Unlock Bypass

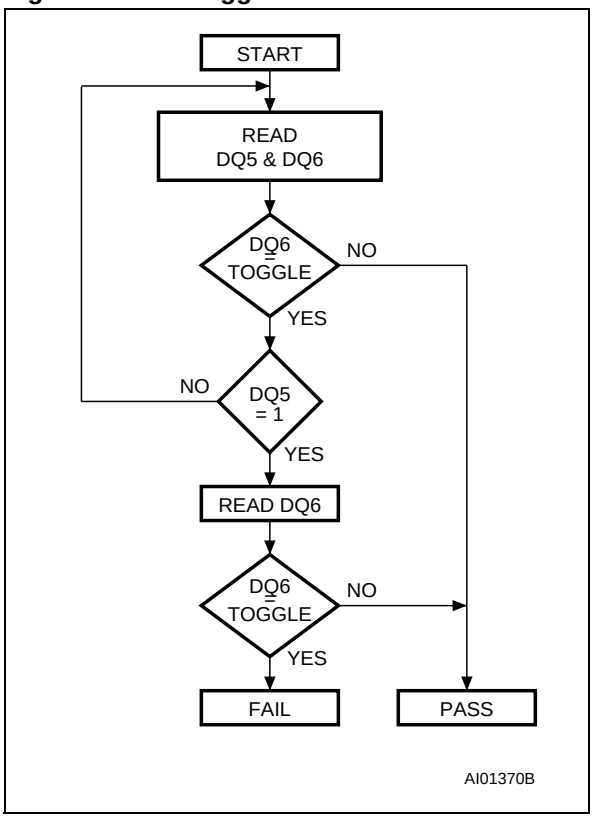
The Unlock Bypass instructions allow the system to program bytes to the Flash memories faster than using the standard Program instruction. The Unlock Bypass mode is entered by first initiating two Unlock cycles. This is followed by a third WRITE cycle containing the Unlock Bypass code, 20h (as shown in Table 28., page 27).

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

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

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

Figure 8. Data Toggle Flowchart



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 28., page 27. If any byte of the Bulk Erase instruction is wrong, the Bulk Erase instruction aborts and the device is reset to the Read Flash memory status.

During a Bulk Erase, the memory status may be checked by reading the Error Flag Bit (DQ5), the Toggle Flag Bit (DQ6), and the Data Polling Flag Bit (DQ7), as detailed in "PROGRAMMING FLASH MEMORY, page 31. The Error Flag Bit (DQ5) returns a '1' if there has been an Erase Failure (maximum number of Erase cycles has been executed).

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

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

Flash Sector Erase

The Sector Erase instruction uses six WRITE operations, as described in Table 28., page 27. Additional Flash Sector Erase codes and Flash memory sector addresses can be written subsequently to erase other Flash memory sectors in parallel, without further coded cycles, if the additional bytes are transmitted in a shorter time than the time-out period of about 100µs. The input of a new Sector Erase code restarts the time-out period.

The status of the internal timer can be monitored through the level of the Erase Time-out Flag Bit (DQ3). If the Erase Time-out Flag Bit (DQ3) is '0,' the Sector Erase instruction has been received and the time-out period is counting. If the Erase Time-out Flag Bit (DQ3) is '1,' the time-out period has expired and the PSD is busy erasing the Flash memory sector(s). Before and during Erase time-out, any instruction other than Suspend Sector Erase and Resume Sector Erase instructions abort the cycle that is currently in progress, and reset the device to READ mode. It is not necessary to program the Flash memory sector with 00h as the PSD does this automatically before erasing (Byte=FFh).

During a Sector Erase, the memory status may be checked by reading the Error Flag Bit (DQ5), the Toggle Flag Bit (DQ6), and the Data Polling Flag Bit (DQ7), as detailed in PROGRAMMING FLASH MEMORY, page 31.

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

Suspend Sector Erase

When a Sector Erase cycle is in progress, the Suspend Sector Erase instruction can be used to suspend the cycle by writing 0B0h to any even address when an appropriate Sector Select (FS0-FS7 or CSBOOT0-CSBOOT3) is High. (See Table 28., page 27). This allows reading of data from another Flash memory sector after the Erase cycle has been suspended. Suspend Sector Erase is accepted only during an Erase cycle and defaults to READ mode. A Suspend Sector Erase instruction executed during an Erase time-out period, in addition to suspending the Erase cycle, terminates the time out period.

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

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

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

Resume Sector Erase

If a Suspend Sector Erase instruction was previously executed, the erase cycle may be resumed with this instruction. The Resume Sector Erase instruction consists in writing 030h to any even address while an appropriate Sector Select (FS0-FS7 or CSBOOT0-CSBOOT3) is High. (See Table 28., page 27.)

SPECIFIC FEATURES

Flash Memory Sector Protect

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

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

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

The sector protection status can be read by the MCU through the primary and secondary Flash memory protection registers (in the CSIOP block). See Table 18., page 21 and Table 19., page 22.

Reset Flash

The Reset Flash instruction consists of one WRITE cycle (see Table 28., page 27). It can also be optionally preceded by the standard two WRITE decoding cycles (writing AAh to AAAh and 55h to 554h).

It must be executed after:

– Reading the Flash Protection Status or Flash ID using the Flash instruction.

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

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

Reset ($\overline{\text{RESET}}$) Signal

A pulse on Reset ($\overline{\text{RESET}}$) aborts any cycle that is in progress, and resets the Flash memory to the READ mode. When the reset occurs during a Program or Erase cycle, the Flash memory takes up to 25 µs to return to the READ mode. It is recommended that the Reset ($\overline{\text{RESET}}$) pulse (except for the one described in Power-Up Reset, page 74) be at least 25 µs so that the Flash memory is always ready for the MCU to fetch the bootstrap instructions after the Reset cycle is complete.

SRAM

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

The SRAM can be backed up using an external battery. The external battery should be connected to Voltage Stand-by (V_{STBY}, PE6). If you have an external battery connected to the PSD, the contents of the SRAM are retained in the event of a power loss. The contents of the SRAM are retained so long as the battery voltage remains at 2 V or greater. If the supply voltage falls below the battery voltage, an internal power switch-over to the battery occurs.

PE7 can be configured as an output that indicates when power is being drawn from the external battery. Battery-on Indicator (V_{BATON}, PE7) is High when the supply voltage falls below the battery voltage and the battery on Voltage Stand-by (V_{STBY}, PE6) is supplying power to the internal SRAM.

SRAM Select (RS0), Voltage Stand-by (V_{STBY}, PC2) and Battery-on Indicator (V_{BATON}, PC4) are all configured using PSDsoft Express Configuration.

The SRAM Select (RS0), V_{BATON} and V_{STBY} are all configured using PSDsoft.

SECTOR SELECT AND SRAM SELECT

Sector Select (FS0-FS7 for primary Flash memory, CSBOOT0-CSBOOT3 for secondary Flash memory) and SRAM Select (RS0) are all outputs of the DPLD. They are setup using PSDsoft. The following rules apply to the equations for these signals:

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

Example

FS0 is valid when the address is in the range of 8000h to BFFFh, CSBOOT0 is valid from 8000h to 9FFFh, and RS0 is valid from 8000h to 87FFh. Any address in the range of RS0 always accesses the SRAM. Any address in the range of CSBOOT0 greater than 87FFh (and less than 9FFFh) automatically addresses secondary Flash memory segment 0. Any address greater than 9FFFh accesses the primary Flash memory segment 0. You can see that half of the primary Flash memory segment 0 and one-fourth of secondary Flash memory segment 0 cannot be accessed in this example. Also note that an equation that defined FS1 to anywhere in the range of 8000h to BFFFh would *not* be valid.

Figure 9 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

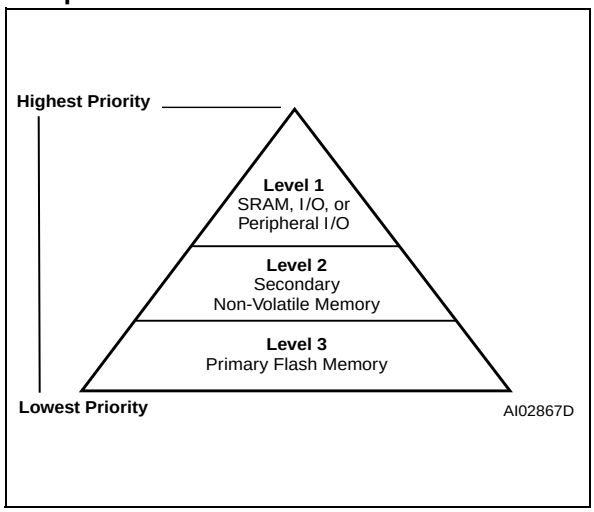
The 80C51 and compatible family of MCUs have separate address spaces for Program memory (selected using Program Select Enable (PSEN, CNTL2)) and Data memory (selected using Read Strobe (RD, CNTL1)). Any of the memories within the PSD can reside in either space or both spaces. This is controlled through manipulation of the VM register that resides in the CSIOP space.

The VM register is set using PSDsoft to have an initial value. It can subsequently be changed by the MCU so that memory mapping can be changed on-the-fly.

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

Table 24., page 23 describes the VM Register.

Figure 9. Priority Level of Memory and I/O Components



Configuration Modes for MCUs with Separate Program and Data Spaces

Separate Space Modes. Program space is separated from Data space. For example, Program Select Enable (PSEN, CNTL2) is used to access the program code from the primary Flash memory, while Read Strobe (RD, CNTL1) is used to access data from the secondary Flash memory, SRAM and I/O Port blocks. This configuration requires the VM register to be set to 0Ch (see Figure 10).

Combined Space Modes. The Program and Data spaces are combined into one memory space that allows the primary Flash memory, secondary Flash memory, and SRAM to be accessed by either Program Select Enable (PSEN, CNTL2) or Read Strobe (RD, CNTL1). For example, to configure the primary Flash memory in Combined space, Bits b2 and b4 of the VM register are set to '1' (see Figure 11).

Figure 10. 8031 Memory Modules – Separate Space

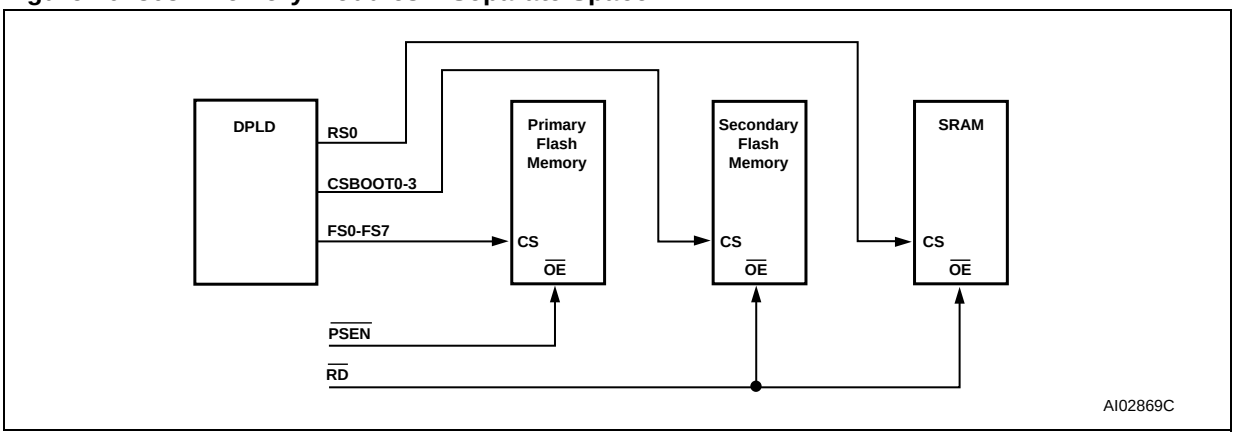
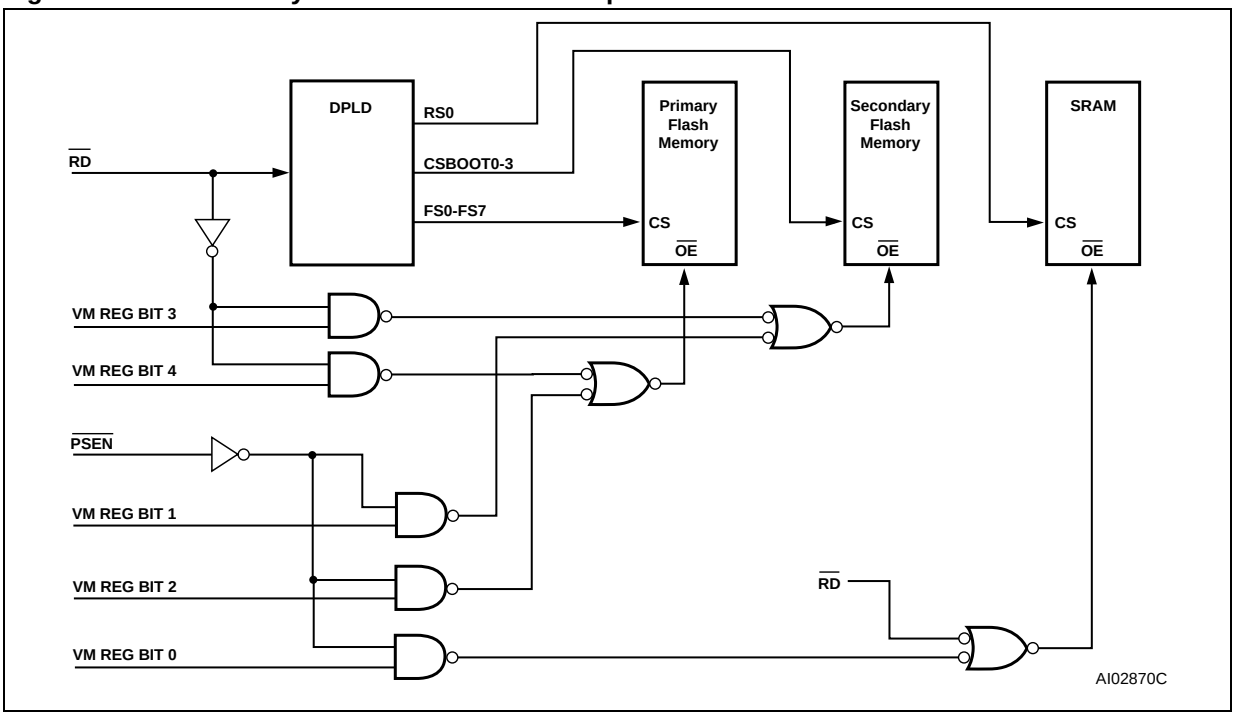


Figure 11. 8031 Memory Modules – Combined Space



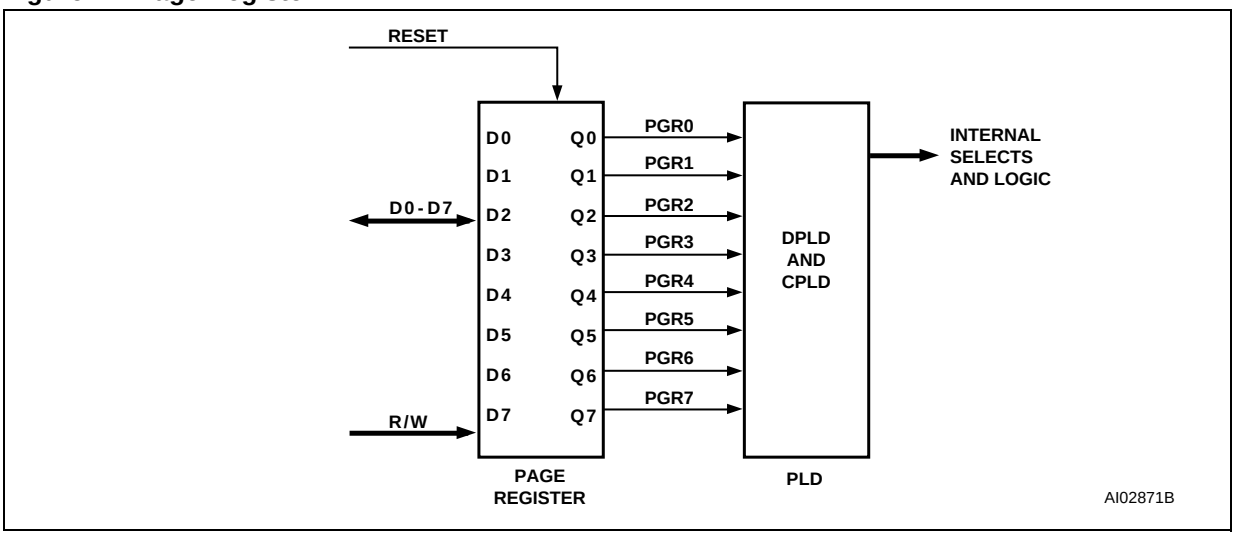
PAGE REGISTER

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

If memory paging is not needed, or if not all 8 page register bits are needed for memory paging, then these bits may be used in the CPLD for general logic. See Application Note *AN1154*.

Figure 12 shows the Page Register. The eight flip-flops in the register are connected to the internal data bus D0-D7. The MCU can write to or read from the Page Register. The Page Register can be accessed at address location CSIOP + E0h.

Figure 12. Page Register



MEMORY ID REGISTERS

The 8-bit Read-only Memory Status Registers are included in the CSIOP space. The user can determine the memory configuration of the PSD device by reading the Memory ID0 and ID1 Registers. The contents of the registers are defined in Table 25., page 23 and Table 26., page 23.

PLDS

The PLDs bring programmable logic functionality to the PSD. After specifying the logic for the PLDs in PSDsoft, 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 Decode PLD (DPLD), page 40 and Complex PLD (CPLD), page 41. Figure 13 shows the configuration of the PLDs.

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

The CPLD can be used for logic functions, such as loadable counters and shift registers, state machines, and encoding and decoding logic. These logic functions can be constructed using the 16 Output Macrocells (OMC), 24 Input Macrocells (IMC), and the AND Array. The CPLD can also be used to generate External Chip Select (ECS0-ECS2) signals.

The AND Array is used to form product terms. These product terms are specified using PSDsoft. An Input Bus consisting of 82 signals is connected to the PLDs. The signals are shown in Table 30.

The Turbo Bit in PSD

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

Additionally, five bits are available in PMMR2 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.

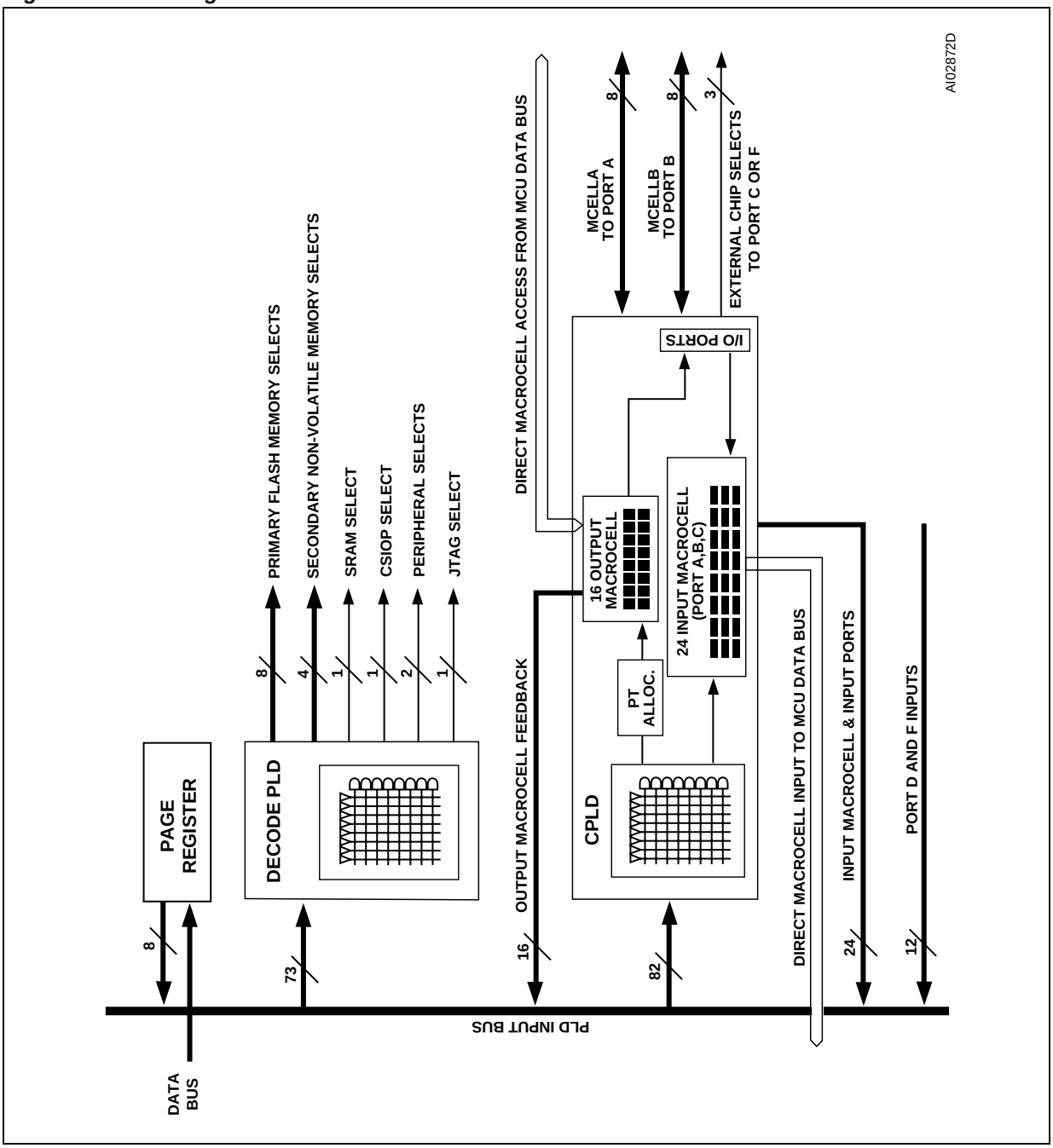
Each of the two PLDs has unique characteristics suited for its applications. They are described in the following sections.

Table 30. 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	$\overline{RST}$	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	PD3-PD0	4
Port F Inputs	PF7-PF0	8
Page Register	PGR7-PGR0	8
Macrocell A Feedback	MCELLA.FB7-FB0	8
Macrocell B Feedback	MCELLB.FB7-FB0	8
Secondary Flash memory Program Status Bit	Ready/$\overline{Busy}$	1

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

Figure 13. PLD Diagram

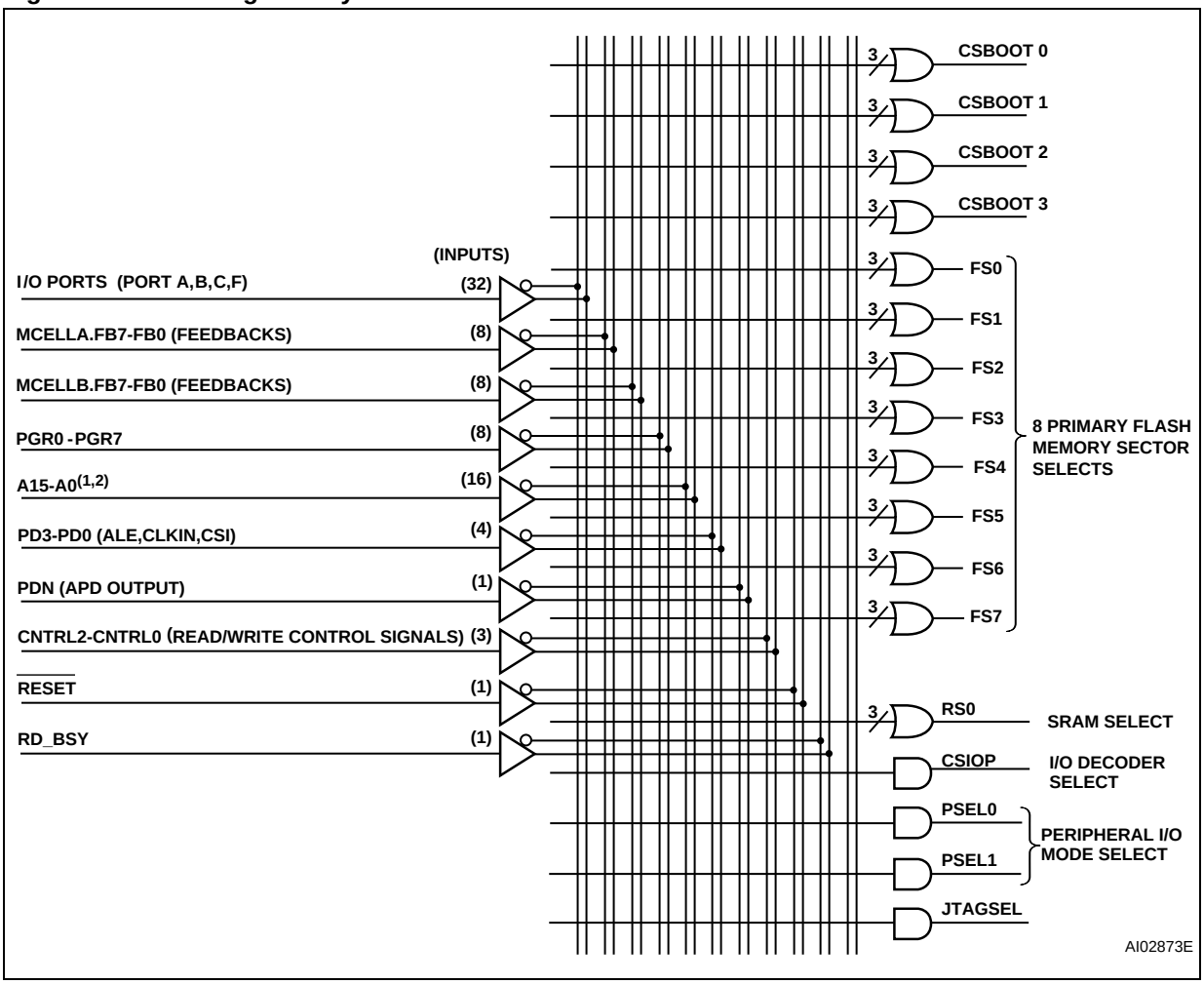


Decode PLD (DPLD)

The DPLD, shown in Figure 14, is used for decoding the address for internal and external components. The DPLD can be used to generate the following decode signals:

- 8 Sector Select (FS0-FS7) signals for the primary Flash memory (three product terms each)
- 4 Sector Select (CSBOOT0-CSBOOT3) signals for the secondary Flash memory (three product terms each)
- 1 internal SRAM Select (RS0) signal (three product terms)
- 1 internal CSIOP Select (PSD Configuration Register) signal
- 1 JTAG Select signal (enables JTAG/ISP on Port E)
- 2 internal Peripheral Select signals (Peripheral I/O mode).

Figure 14. DPLD Logic Array



Note: 1. The address inputs are A19-A4 in 80C51XA mode.
2. Additional address lines can be brought into PSD via Port A, B, C, D or F.

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 three External Chip Select (ECS0-ECS2), routed to Port D.

Although External Chip Select (ECS0-ECS2) can be produced by any Output Macrocell (OMC), these three External Chip Select (ECS0-ECS2) on Port D do not consume any Output Macrocells (OMC).

As shown in Figure 13., page 39, the CPLD has the following blocks:

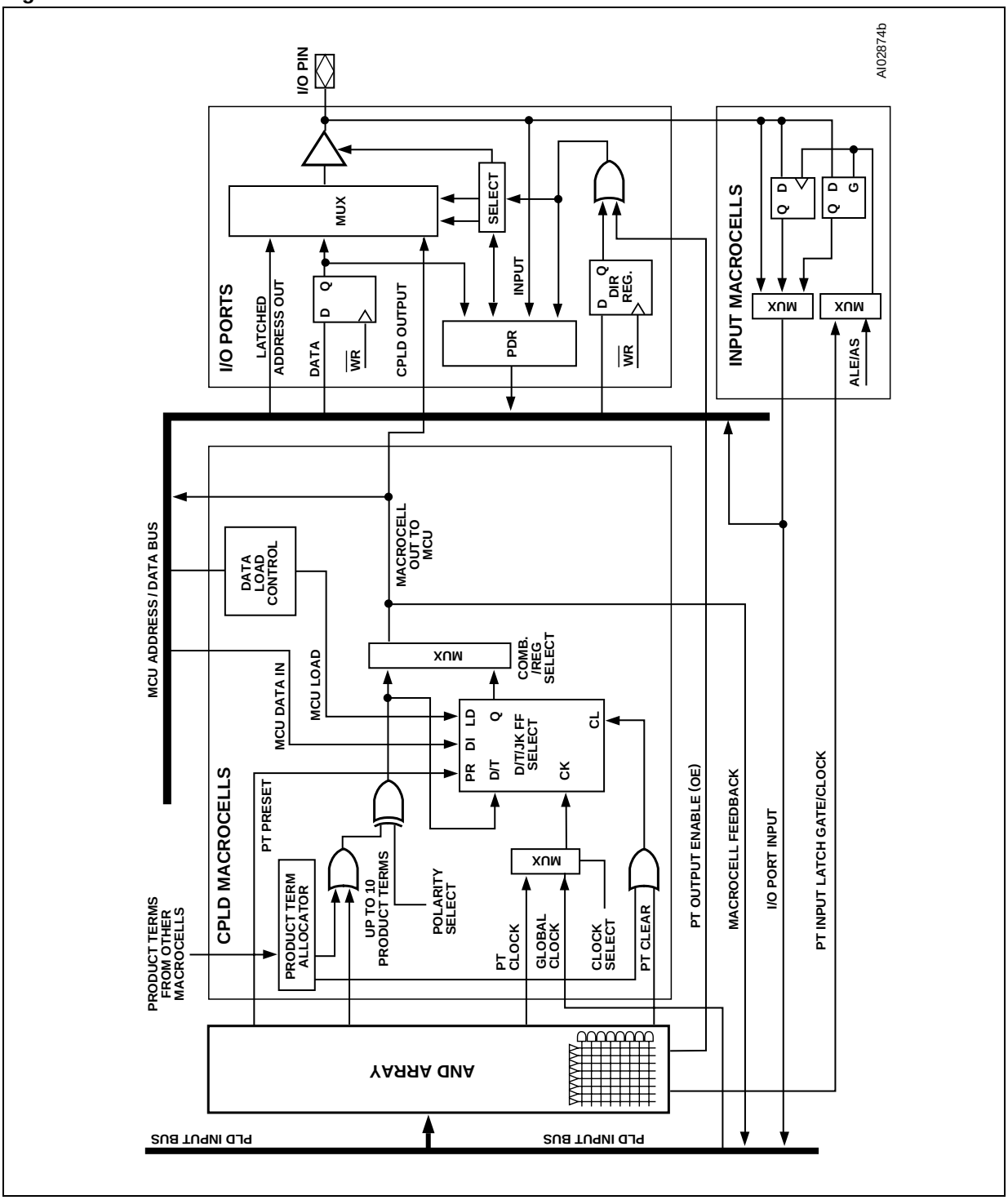
- 24 Input Macrocells (IMC)
- 16 Output Macrocells (OMC)
- Macrocell Allocator
- Product Term Allocator
- AND Array capable of generating up to 137 product terms
- Four I/O Ports.

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

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

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

Figure 15. Macrocell and I/O Port



Output Macrocell (OMC)

Eight of the Output Macrocells (OMC) are connected to Port A pins and are named as McellA0-McellA7. The other eight macrocells are connected to Port B pins and are named as McellB0-McellB7.

The Output Macrocell (OMC) architecture is shown in Figure 16., page 45. As shown in the figure, there are native product terms available from the AND Array, and borrowed product terms available (if unused) from other Output Macrocells (OMC). The polarity of the product term is controlled by the XOR gate. The Output Macrocell (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 Output Macrocell (OMC) block can be configured as a D, T, JK, or SR type in the PSDsoft program. The flip-flop's clock, preset, and clear inputs may be driven from a product term of the AND Array. Alternatively, CLKIN (PD1) can be used for the clock input to the flip-flop. The flip-flop is clocked to the rising edge of CLKIN (PD1). The preset and clear are active High inputs. Each clear input can use up to two product terms.

Table 31. Output Macrocell Port and Data Bit Assignments

Output Macrocell	Port Assignment	Native Product Terms	Maximum Borrowed Product Terms	Data Bit for Loading or Reading
McellA0	Port A0	3	6	D0
McellA1	Port A1	3	6	D1
McellA2	Port A2	3	6	D2
McellA3	Port A3	3	6	D3
McellA4	Port A4	3	6	D4
McellA5	Port A5	3	6	D5
McellA6	Port A6	3	6	D6
McellA7	Port A7	3	6	D7
McellB0	Port B0	4	5	D0
McellB1	Port B1	4	5	D1
McellB2	Port B2	4	5	D2
McellB3	Port B3	4	5	D3
McellB4	Port B4	4	6	D4
McellB5	Port B5	4	6	D5
McellB6	Port B6	4	6	D6
McellB7	Port B7	4	6	D7

Product Term Allocator

The CPLD has a Product Term Allocator. The PSD 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:

– McellA0-McellA7 all have three native product terms and may borrow up to six more
– McellB0-McellB3 all have four native product terms and may borrow up to five more
– McellB4-McellB7 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 that consume other Output Macrocells (OMC). If external product terms are used, extra delay is added for the equation that required the extra product terms.

This is called product term expansion. PSDsoft performs this expansion as needed.

Loading and Reading the Output Macrocells (OMC)

The Output Macrocells (OMC) block occupies a memory location in the MCU address space, as defined by the CSIOP block (see I/O PORTS, page 59). The flip-flops in each of the 16 Output Macrocells (OMC) can be loaded from the data bus by a MCU. Loading the Output Macrocells (OMC) 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 Output Macrocells (OMC) on the trailing edge of the Write Strobe (WR, CNTL0) signal.

The OMC Mask Register

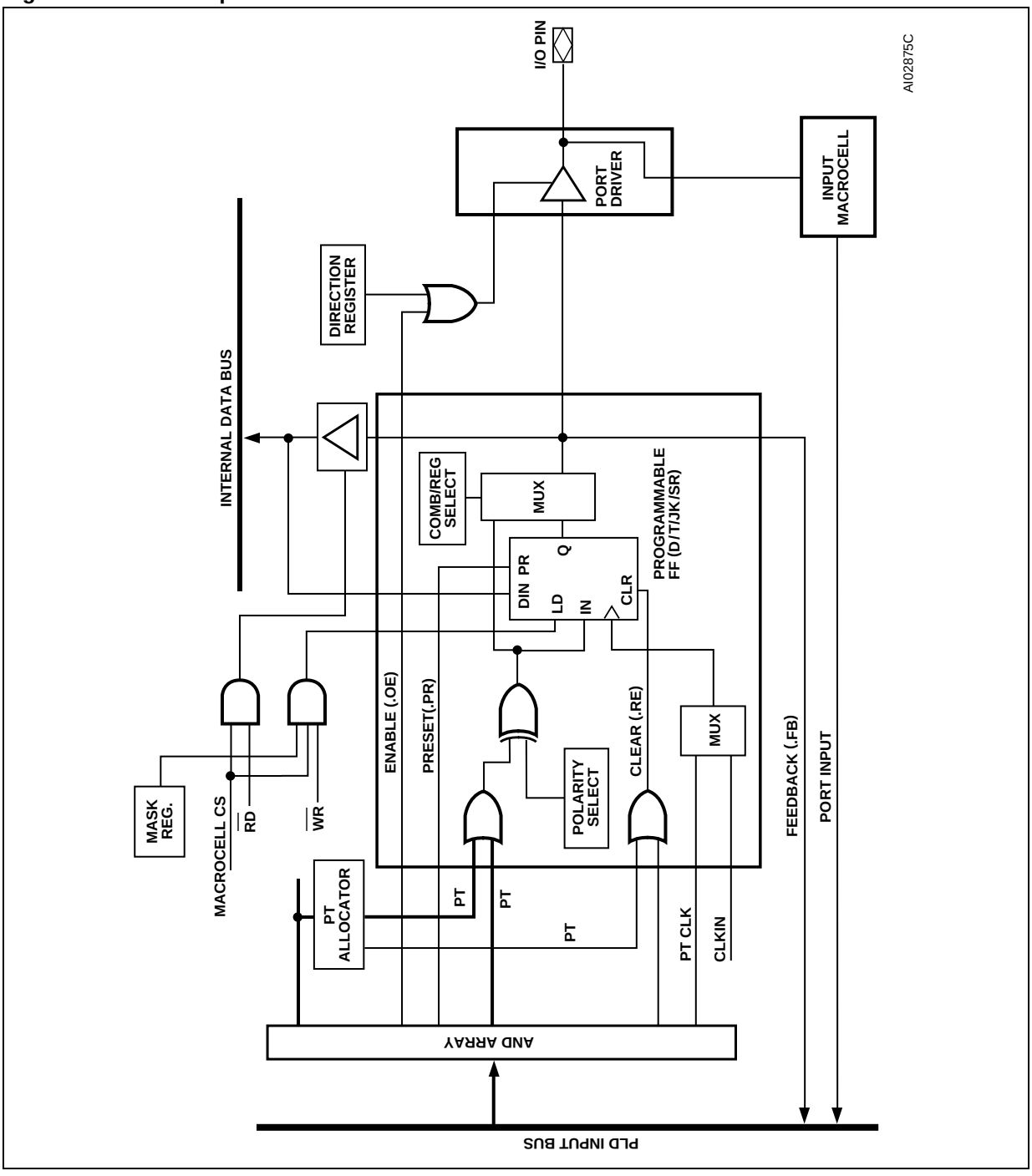
There is one Mask Register for each of the two groups of eight Output Macrocells (OMC). The Mask Registers can be used to block the loading of data to individual Output Macrocells (OMC). The default value for the Mask Registers is 00h, which allows loading of the Output Macrocells (OMC). When a given bit in a Mask Register is set to a '1,' the MCU is blocked from writing to the associated Output Macrocells (OMC). For example, suppose McellA0-McellA3 are being used for a state machine. You would not want a MCU WRITE to McellA to overwrite the state machine registers. Therefore, you would want to load the Mask Register for McellA (Mask Macrocell AB) with the value 0Fh.

The Output Enable of the OMC

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

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

Figure 16. CPLD Output Macrocell



Input Macrocells (IMC)

The CPLD has 24 Input Macrocells (IMC), one for each pin on Ports A, B, and C. The architecture of the Input Macrocells (IMC) is shown in Figure 17., page 47. The Input Macrocells (IMC) are individually configurable, and can be used as a latch, register, or to pass incoming Port signals prior to driving them onto the PLD input bus. The outputs of the Input Macrocells (IMC) can be read by the MCU through the internal data bus.

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

Configurations for the Input Macrocells (IMC) are specified by PSDsoft. Outputs of the Input Macrocells (IMC) can be read by the MCU via the IMC buffer. See I/O PORTS, page 59.

Input Macrocells (IMC) can use Address Strobe (ALE/AS, PD0) to latch address bits higher than A15. Any latched addresses are routed to the PLDs as inputs.

Input Macrocells (IMC) are particularly useful with handshaking communication applications where two processors pass data back and forth through a common mailbox. Figure 18., page 48 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 Input Macrocells (IMC) and the Master can then read the Input Macrocells (IMC) directly.

Note that the “Slave-Read” and “Slave-Wr” signals are product terms that are derived from the Slave MCU inputs Read Strobe ($\overline{RD}$, CNTL1), Write Strobe ($\overline{WR}$, CNTL0), and Slave_CS.

Figure 17. Input Macrocell

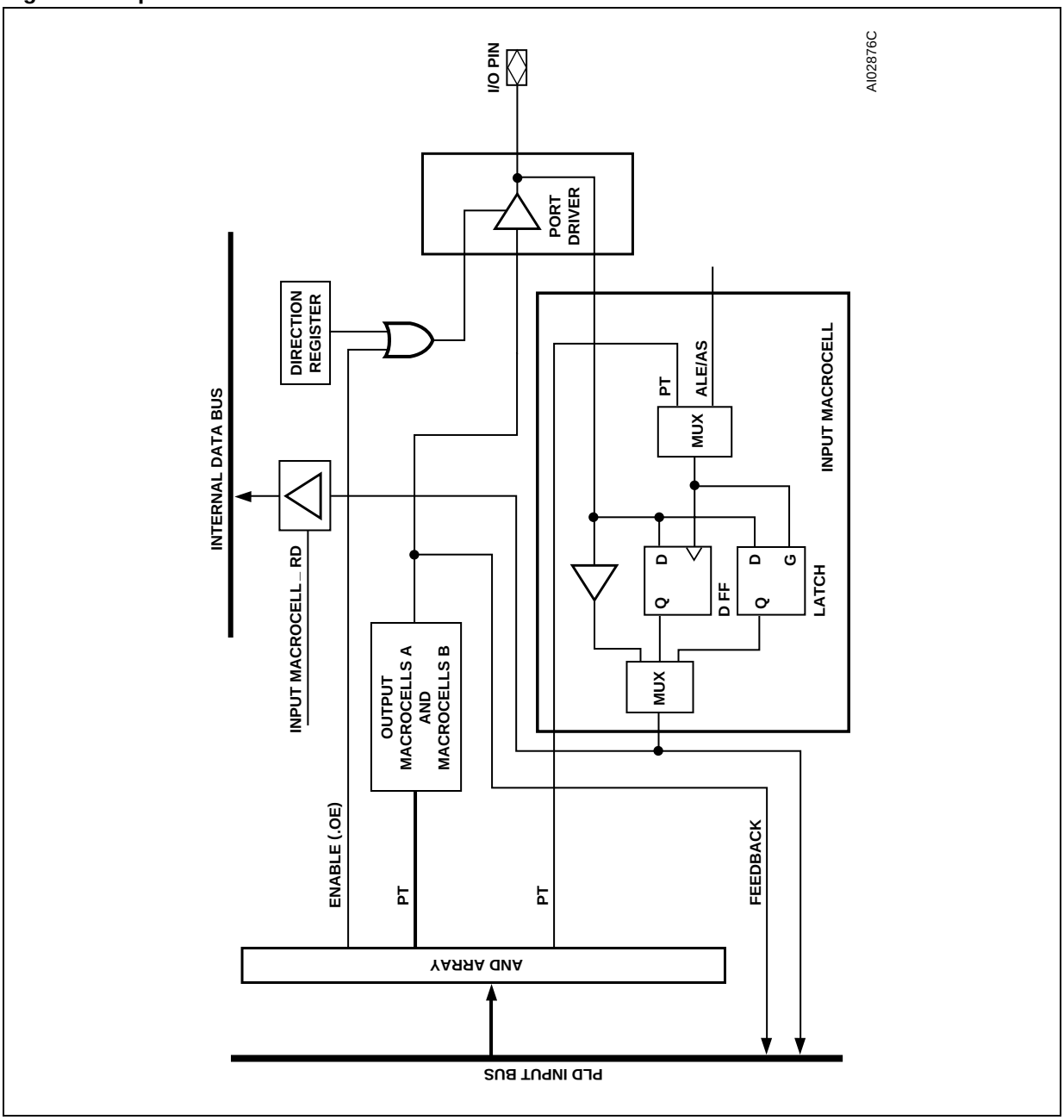
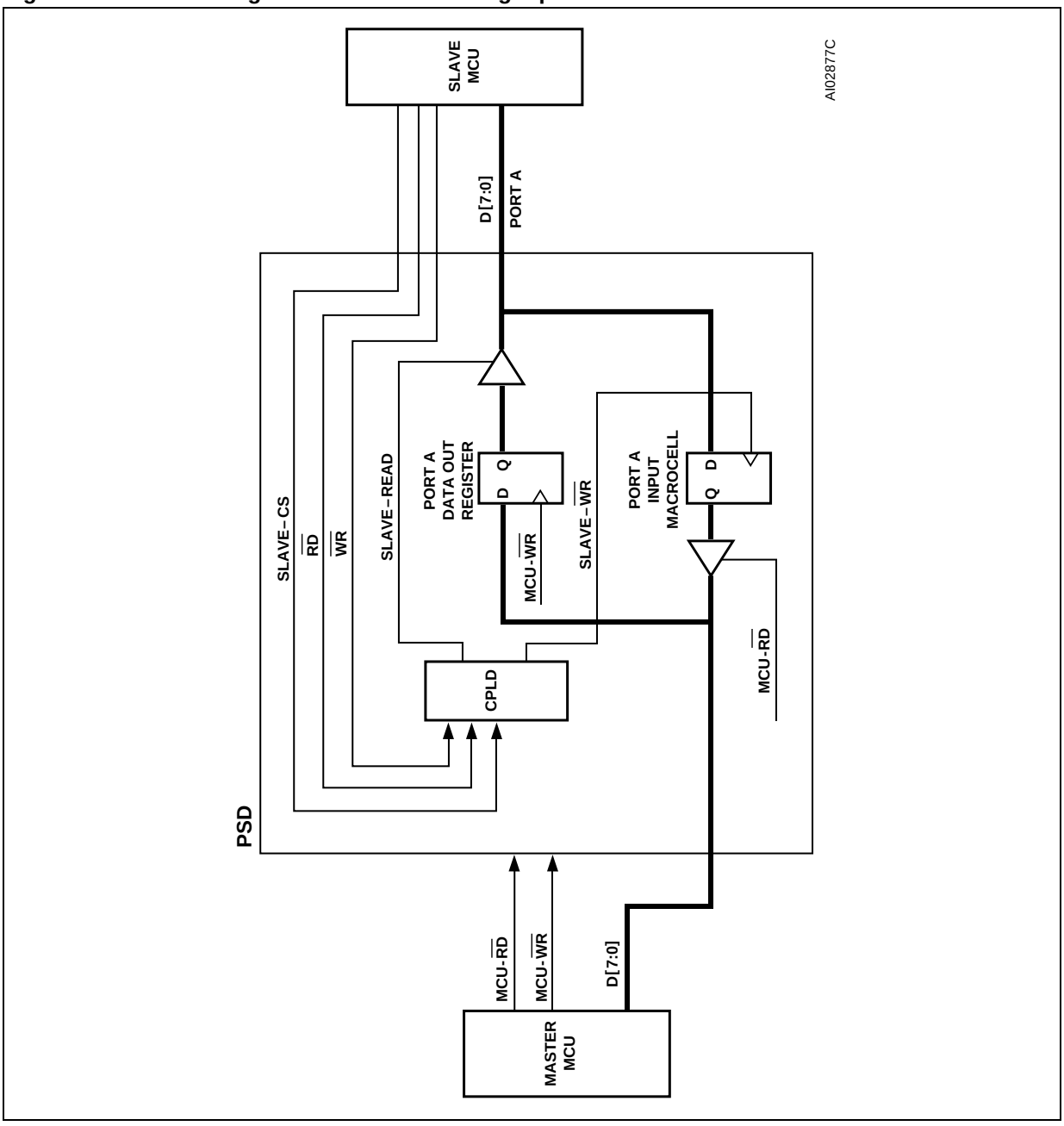


Figure 18. Handshaking Communication Using Input Macrocells

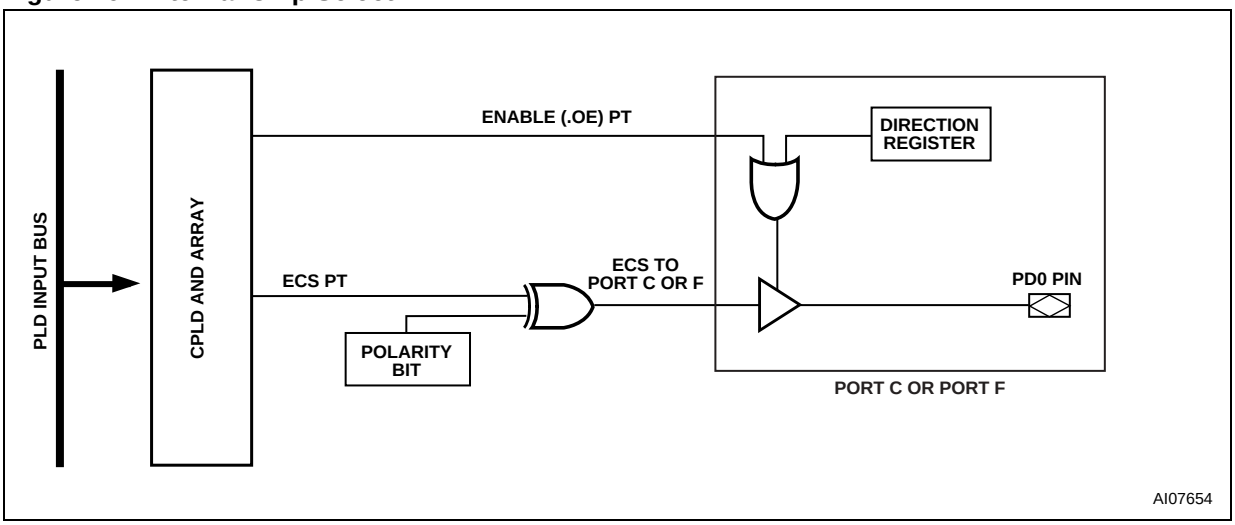


External Chip

The CPLD also provides eight Chip Select outputs that can be used to select external devices. The Chip Selects can be routed to either Port C or Port F, depending on the pin declaration in the PSDsoft. Each Chip Select (ECS0-ECS7) 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 19).

Figure 19. External Chip Select



MCU BUS INTERFACE

The “no-glue logic” 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 32. The interface type is specified using the PSDsoft.

Table 32. MCUs and their Control Signals

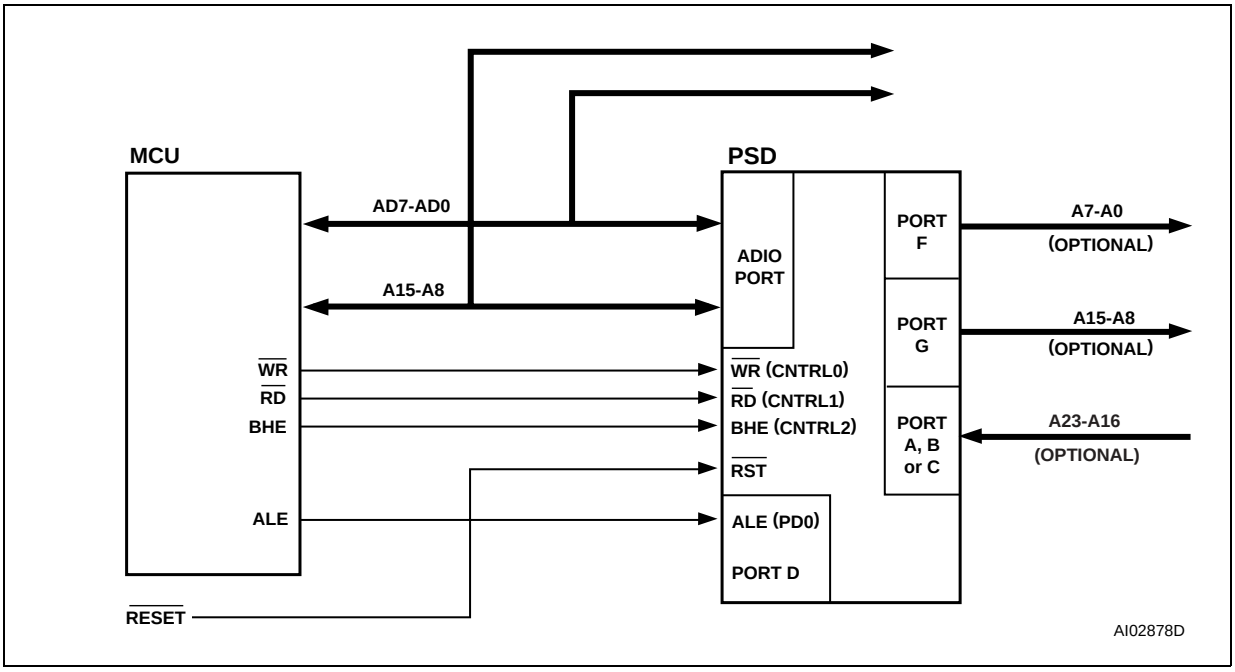
MCU	Data Bus Width	CNTL0	CNTL1	CNTL2	PC7	PD0[2]	ADIO0	PA3-PA0	PA7-PA4
8031/8051	8	WR	RD	PSEN	(Note 1)	ALE	A0	(Note 1)	(Note 1)
80C51XA	8	WR	RD	PSEN	(Note 1)	ALE	A4	A3-A0	(Note 1)
80C251	8	WR	PSEN	(Note 1)	(Note 1)	ALE	A0	(Note 1)	(Note 1)
80C251	8	WR	RD	PSEN	(Note 1)	ALE	A0	(Note 1)	(Note 1)
80198	8	WR	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)
68HC05C0	8	WR	RD	(Note 1)	(Note 1)	AS	A0	(Note 1)	(Note 1)
68HC912	8	R/$\overline{W}$	E	(Note 1)	DBE	AS	A0	(Note 1)	(Note 1)
Z80	8	WR	RD	(Note 1)	(Note 1)	(Note 1)	A0	D3-D0	D7-D4
Z8	8	R/$\overline{W}$	DS	(Note 1)	(Note 1)	AS	A0	(Note 1)	(Note 1)
68330	8	R/$\overline{W}$	DS	(Note 1)	(Note 1)	AS	A0	(Note 1)	(Note 1)
M37702M2	8	R/$\overline{W}$	E	(Note 1)	(Note 1)	ALE	A0	D3-D0	D7-D4

Note: 1. Unused CNTL2 pin can be configured as PLD input. Other unused pins (PD3-PD0, PA3-PA0) can be configured for other I/O functions.
2. ALE/AS input is optional for MCUs with a non-multiplexed bus

PSD Interface to a Multiplexed 8-Bit Bus

Figure 20 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 E, For G. The PSD drives the ADIO data bus only when one of its internal resources is accessed and Read Strobe (RD, CNTL1) is active. Should the system address bus exceed sixteen bits, Ports A, B, C, or F may be used as additional address inputs.

Figure 20. An Example of a Typical 8-bit Multiplexed Bus Interface



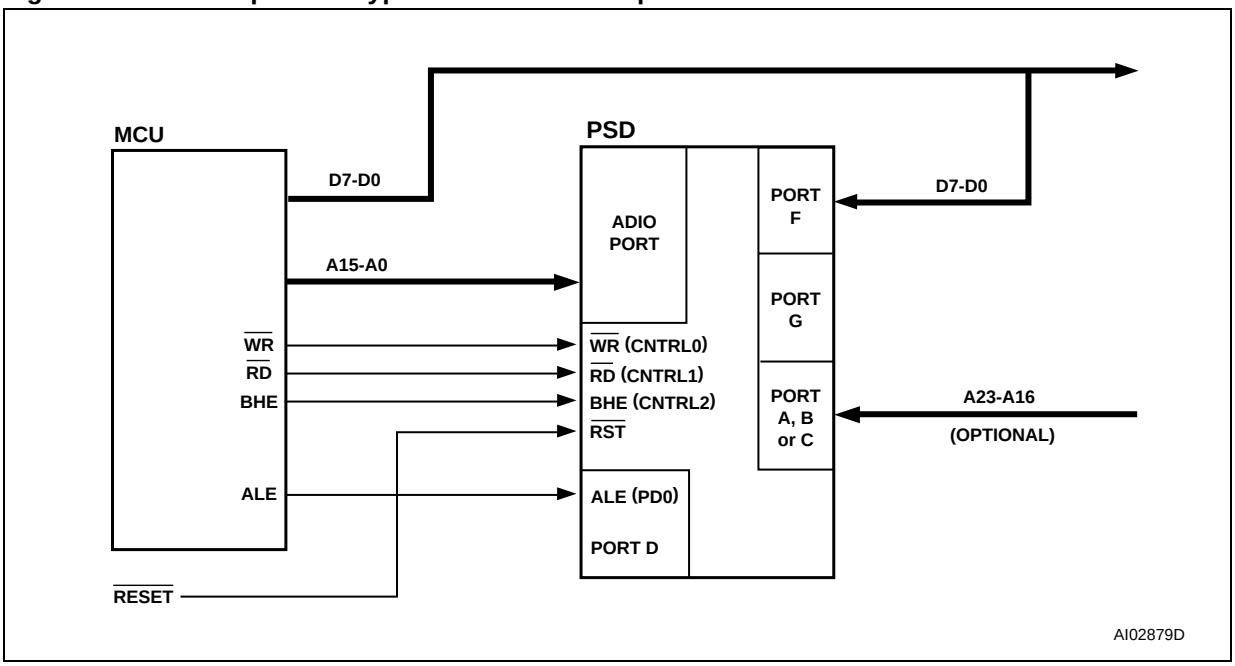
PSD Interface to a Non-Multiplexed 8-Bit Bus

Figure 21 shows an example of a system using a MCU 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 F. Port F is in tri-state mode when the PSD is not accessed by the MCU. Should the system address bus exceed sixteen bits, Ports A, B or C may be used for additional address inputs.

MCU Bus Interface Examples

Figures 22 through 25 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.

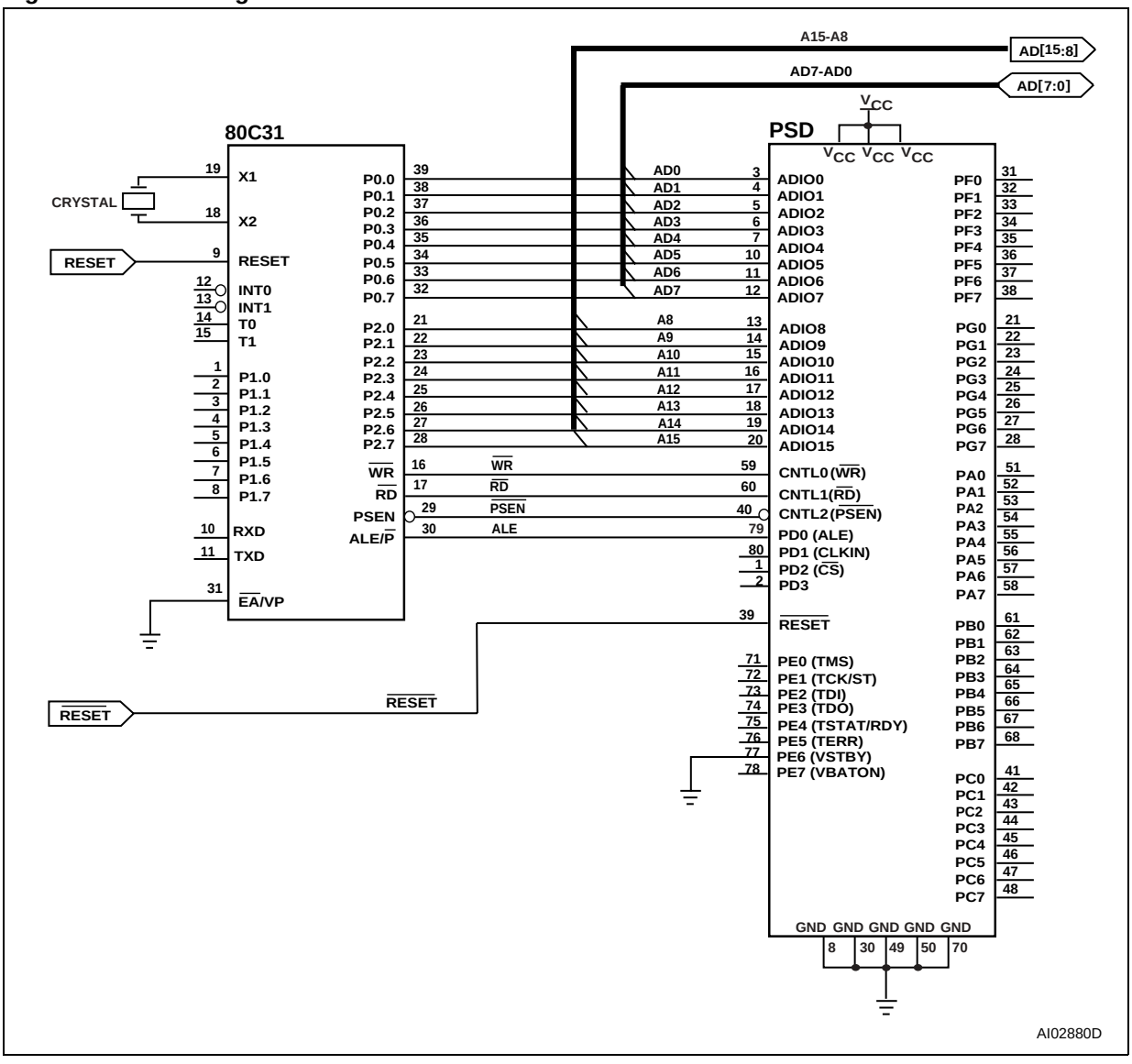
Figure 21. An Example of a Typical 8-bit Non-Multiplexed Bus Interface



80C31

Figure 22 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 (PSEN, CNTL2), Read Strobe (RD, CNTL1), and Write Strobe (WR, CNTL0) may be used for accessing the internal memory and I/O Ports blocks. Address Strobe (ALE/AS, PD0) latches the address.

Figure 22. Interfacing the PSD with an 80C31



80C251

The Intel 80C251 MCU features a user-configurable bus interface with four possible bus configurations, as shown in Table 33.

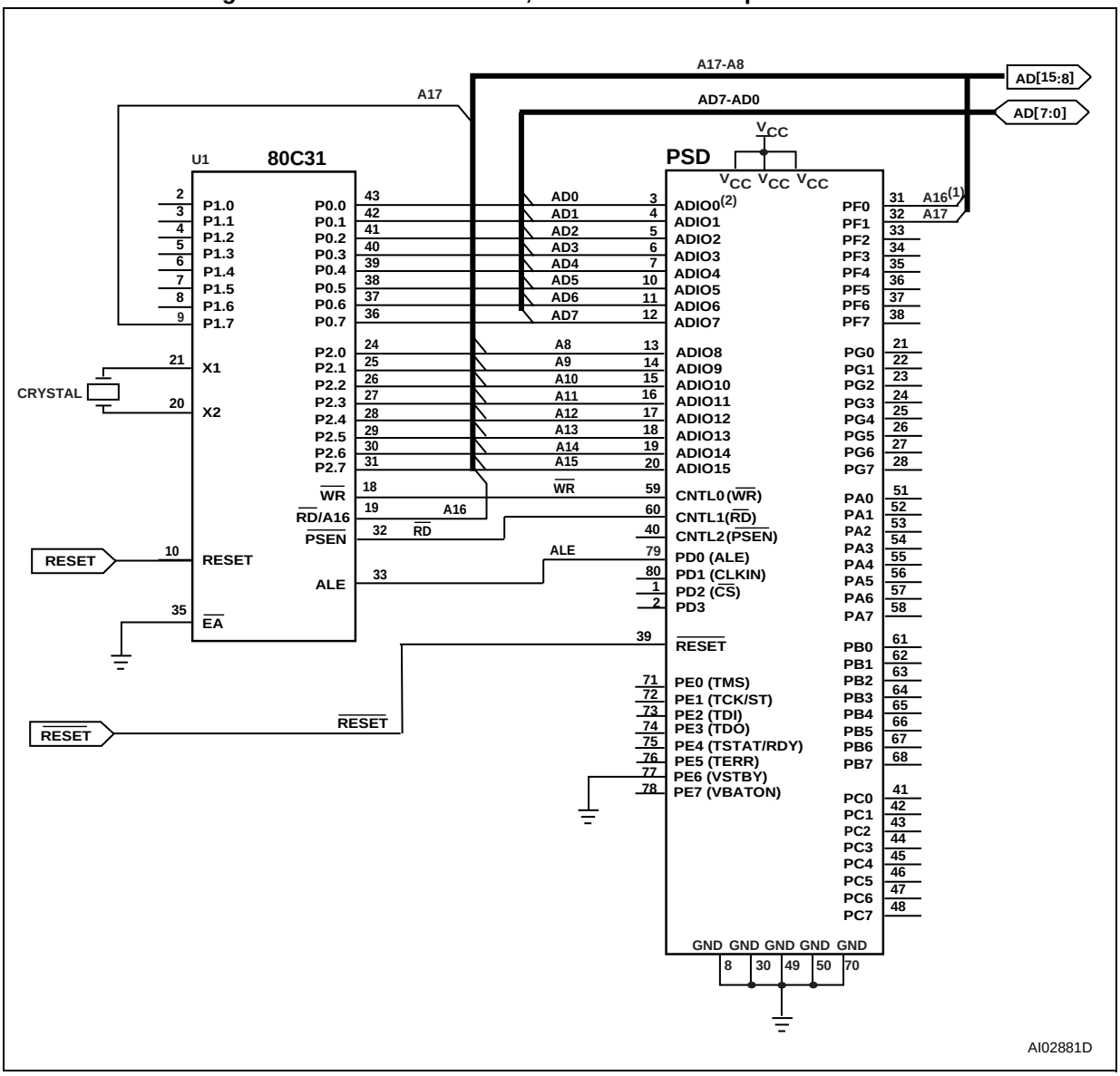
The first configuration is 80C31 compatible, and the bus interface to the PSD is identical to that shown in Figure 22., page 53. The second and third configurations have the same bus connection as shown in Table 34., page 55. There is only one Read Strobe (PSEN) connected to CNTL1 on the PSD. The A16 connection to PA0 allows for a larger address input to the PSD. The fourth configuration is shown in Figure 23., page 56. Read Strobe (RD) is connected to CNTL1 and Program Select Enable (PSEN) is connected to CNTL2.

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 Address Strobe (ALE/AS, PD0) is active in every bus cycle. In Page mode, data (D7-D0) is multiplexed with address (A15-A8). In a bus cycle where there is a Page hit, Address Strobe (ALE/AS, PD0) is not active and only addresses (A7-A0) 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 Address Strobe (ALE/AS, PD0) is not required. The PSD access time is measured from address (A7-A0) valid to data in valid.

Table 33. 80C251 Configurations

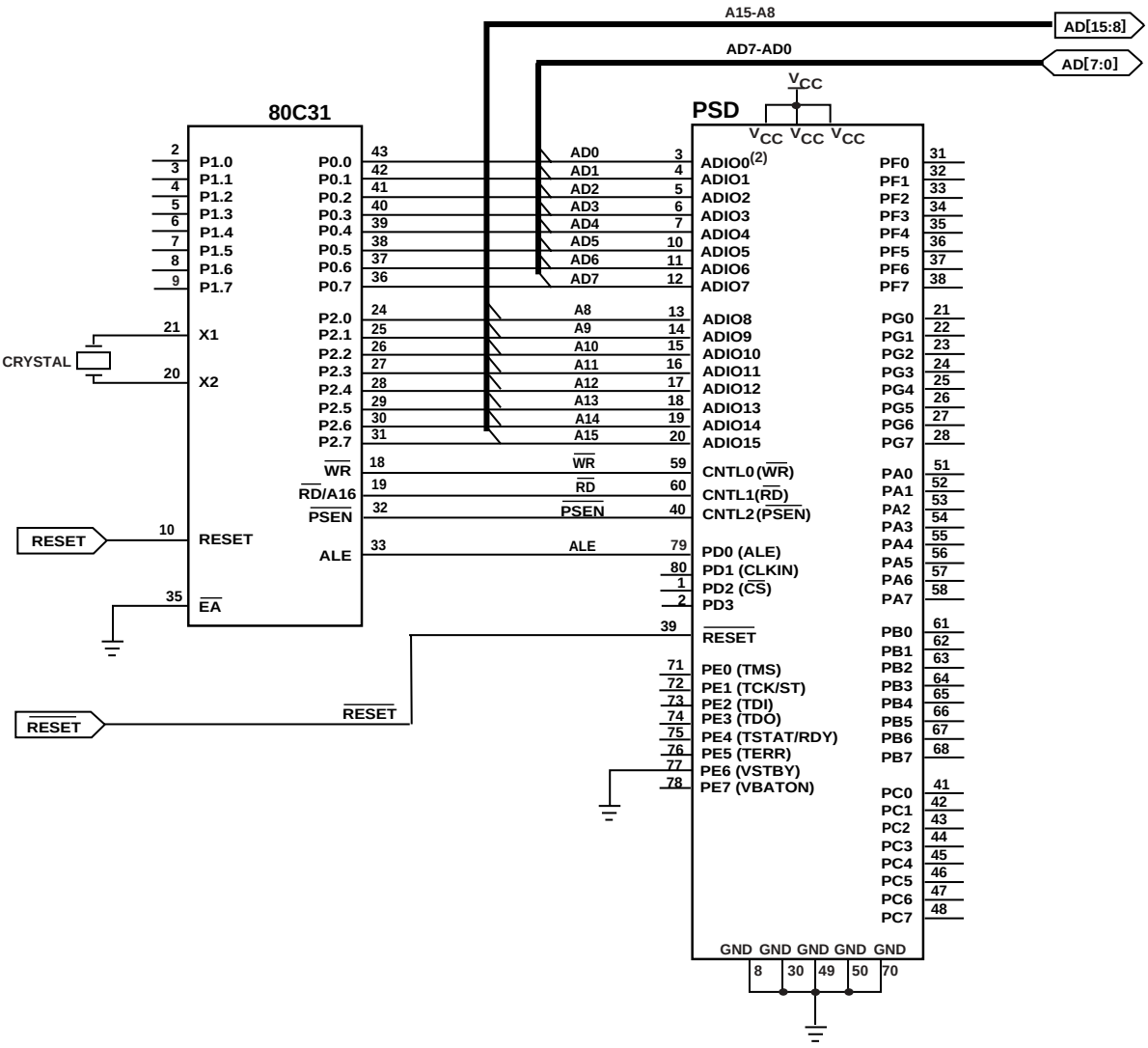
Configuration	80C251 READ/WRITE Pins	Connecting to PSD Pins	Page Mode
1	WR RD PSEN	CNTL0 CNTL1 CNTL2	Non-Page Mode, 80C31 compatible A7-A0 multiplex with D7-D0
2	WR PSEN only	CNTL0 CNTL1	Non-Page Mode A7-A0 multiplex with D7-D0
3	WR PSEN only	CNTL0 CNTL1	Page Mode A15-A8 multiplex with D7-D0
4	WR RD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

Table 34. 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.

Figure 23. Interfacing the PSD with the 80C251, with $\overline{RD}$ and $\overline{PSEN}$ Inputs



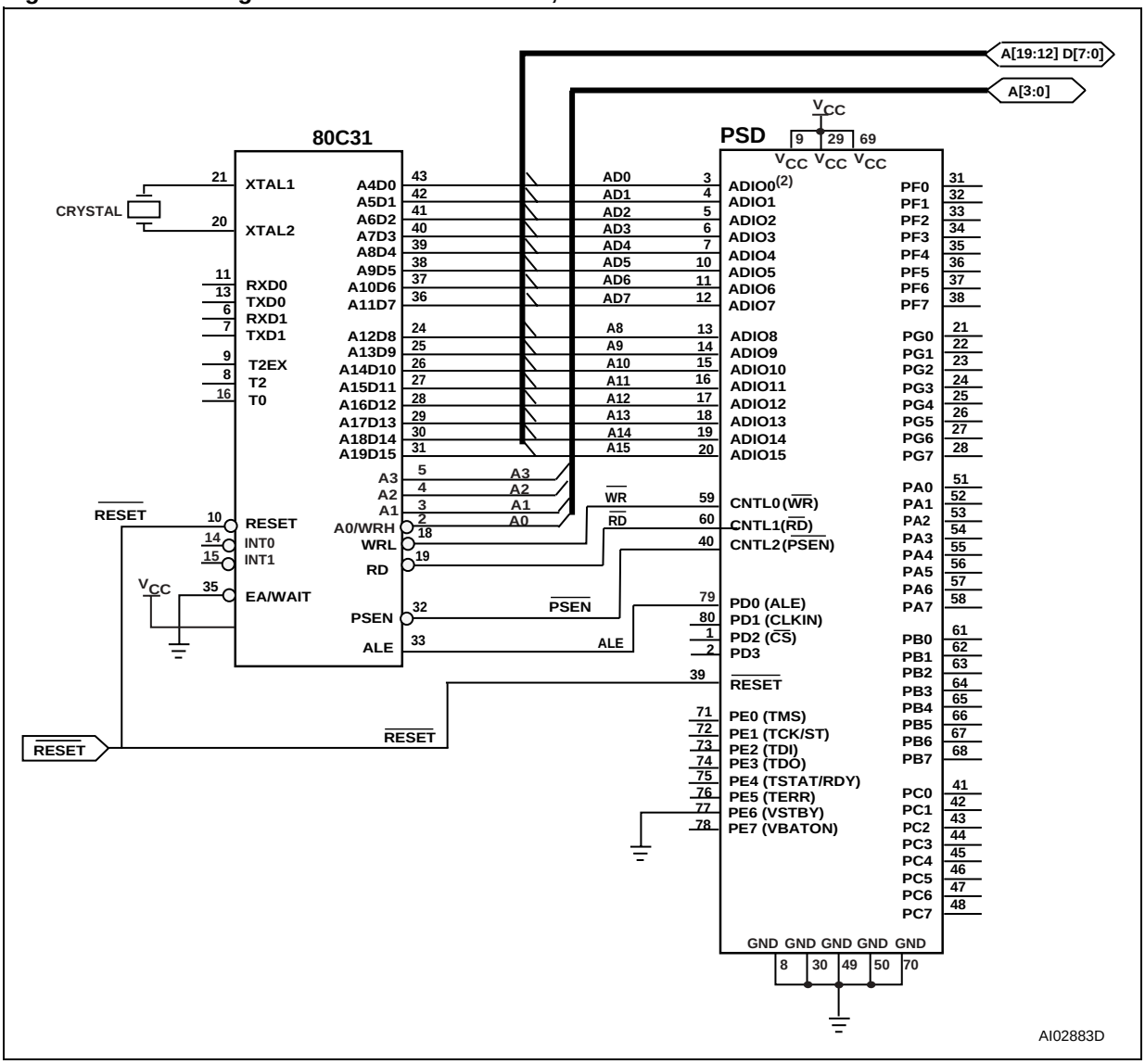
80C51XA

The Philips 80C51XA MCU 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 (as shown in Figure 24).

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 signals 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 Address Strobe (ALE/AS, PD0) does not apply.

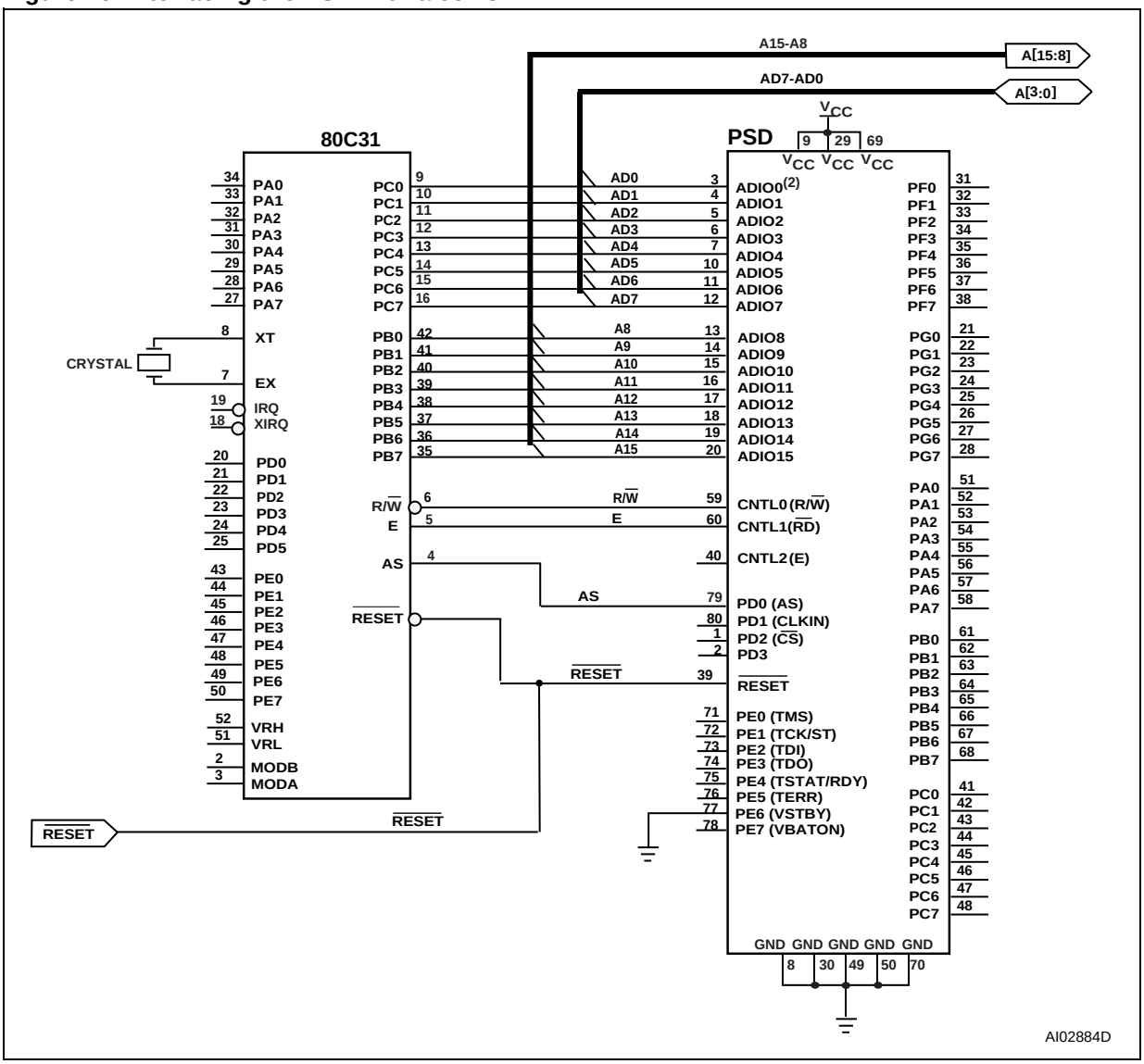
Figure 24. Interfacing the PSD with the 80C51X, 8-bit Data Bus



68HC11

Figure 25 shows a bus interface to a 68HC11 where the PSD is configured in 8-bit multiplexed mode with E and R/W settings. The DPLD can be used to generate the READ and WR signals for external devices.

Figure 25. Interfacing the PSD with a 68HC11



I/O PORTS

There are seven programmable I/O ports: Ports A, B, C, D, E and F. Each of the ports is eight bits except for Port D, which is 4 bits. Each port pin is individually user-configurable, thus allowing multiple functions per port. The ports are configured using PSDsoft or by the MCU writing to on-chip registers in the CSIOP 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 block is shown in Figure 26., page 60. Individual Port architectures are shown in Figure 28., page 66 to Figure 30., page 69. In general, once the purpose for a port pin has been defined, that pin is no longer available for other purposes. Exceptions are noted.

As shown in Figure 26., page 60, the ports contain an output multiplexer whose select signals are driven by the configuration bits in the Control Registers (Ports E, F and G only) and PSDsoft Configuration. Inputs to the multiplexer include the following:

– Output data from the Data Out register
– Latched address outputs
– CPLD macrocell output
– External Chip Select (ECS0-ECS2) from the CPLD.

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

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 is 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 MCU. The Port Data Buffer (PDB) feedback path allows the MCU to check the contents of the registers.

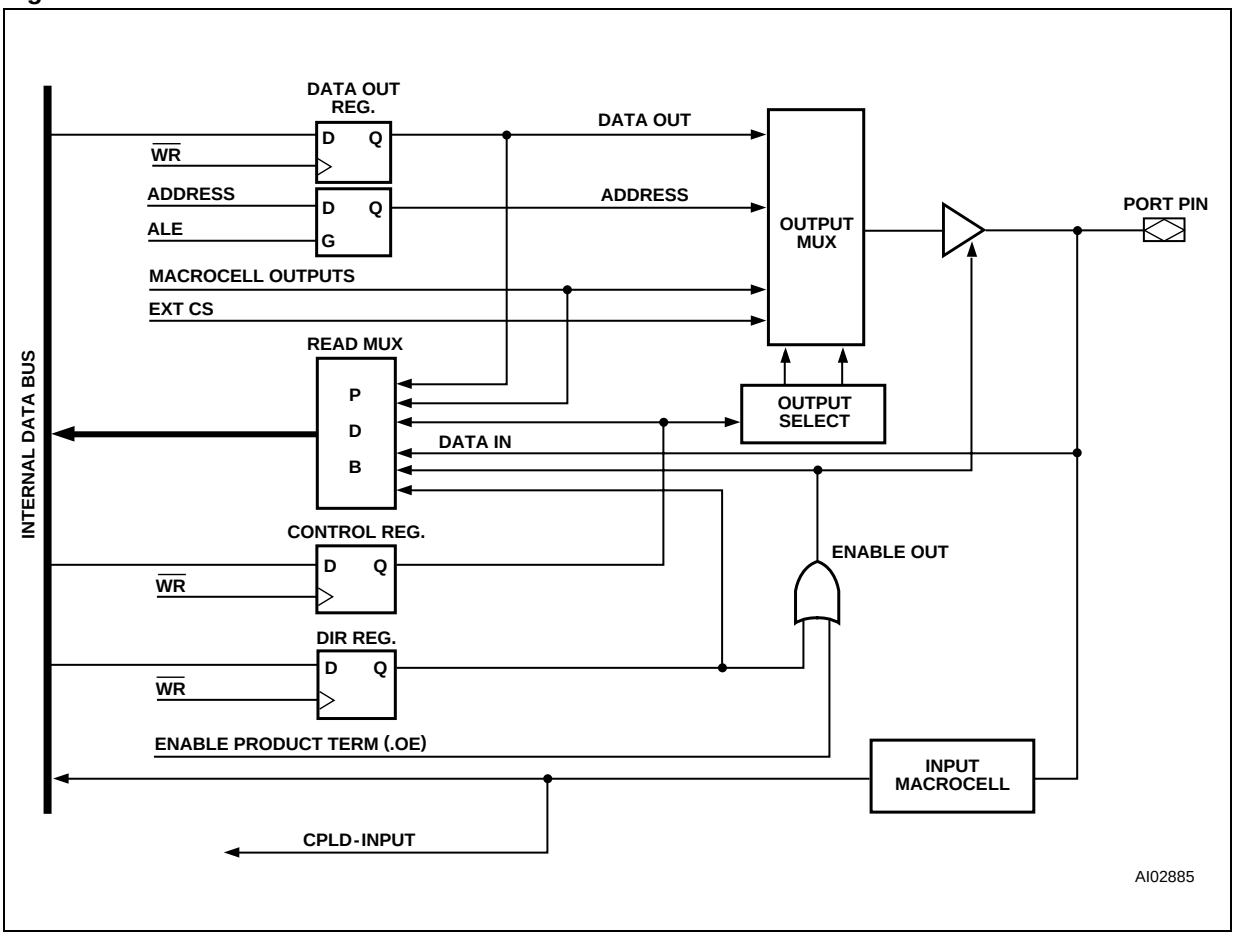
Ports A, B, and C have embedded Input Macrocells (IMC). The Input Macrocells (IMC) can be configured as latches, registers, or direct inputs to the PLDs. The latches and registers are clocked by Address Strobe (ALE/AS, PD0) or a product term from the PLD AND Array. The outputs from the Input Macrocells (IMC) drive the PLD input bus and can be read by the MCU. See Input Macrocells (IMC), page 46.

Port Operating Modes

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

Table 35., page 61 summarizes which modes are available on each port. Table 38., page 64 shows how and where the different modes are configured. Each of the port operating modes are described in the following sections.

Figure 26. General I/O Port Architecture



MCU I/O Mode

In the MCU I/O mode, the MCU uses the I/O Ports block to expand its own I/O ports. By setting up the CSIOP space, the ports on the PSD are mapped into the MCU address space. The addresses of the ports are listed in Table 5., page 19.

A port pin can be put into MCU I/O mode by writing a '0' to the corresponding bit in the Control Register (Ports E, F and G). 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 Direction Register, page 64. When the pin is configured as an output, the content of the Data Out Register drives the pin. When configured as an input, the MCU can read the port input through the Data In buffer. See Figure 26., page 60.

Ports A, B and C do not have Control Registers, and are in MCU I/O mode by default. They can be used for PLD I/O if they are specified in PSDsoft.

PLD I/O Mode

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

Address Out Mode

For MCUs with a multiplexed address/data bus, Address Out Mode can be used to drive latched addresses on to the port pins. These port pins can, in turn, drive external devices. Either the output enable or the corresponding bits of both the Direction Register and Control Register must be set to a '1' for pins to use Address Out Mode. This must be done by the MCU at run-time. See Table 37., page 62 for the address output pin assignments on Ports E, F and G for various MCUs.

Note: Do not drive address signals 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 35. Port Operating Modes

Port Mode	Port A	Port B	Port C	Port D	Port E	Port F	Port G
MCU I/O	Yes	Yes	Yes	Yes	Yes	Yes	Yes
PLD I/O							
McellA Outputs	Yes	No	No	No	No	No	No
McellB Outputs	No	Yes	No	No	No	No	No
Additional Ext. CS Outputs	No	No	Yes	No	No	Yes	No
PLD Inputs	Yes	Yes	Yes	Yes	No	Yes	No
Address Out	No	No	No	No	Yes (A7-A0)	Yes (A7-A0)	Yes (A7-A0) or (A15-A8)
Address In	Yes	Yes	Yes	Yes	No	Yes	No
Data Port	No	No	No	No	No	Yes	No
Peripheral I/O	No	No	No	No	No	Yes	No
JTAG ISP	No	No	No	No	Yes[1]	No	No

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

Table 36. Port Operating Mode Settings

Mode	Defined in PSDsoft	Control Register Setting	Direction Register Setting	VM Register Setting	JTAG Enable
MCU I/O	Declare pins only	0 (Note 4)	1 = output, 0 = input (Note 2)	N/A	N/A
PLD I/O	Declare pins and logic equations	N/A	(Note 2)	N/A	N/A
Data Port (Port F)	Selected for MCU with non-mux bus	N/A	N/A	N/A	N/A
Address Out (Port E, F, G)	Declare pins only	1	1 (Note 2)	N/A	N/A
Address In (Port A,B,C,D, F)	Declare pins or logic equations for Input Macrocells	N/A	N/A	N/A	N/A
Peripheral I/O (Port F)	Logic equations (PSEL0 & 1)	N/A	N/A	PIO Bit = 1	N/A
JTAG ISP (Note 3)	Declare pins only	N/A	N/A	N/A	JTAG_Enable

Note: 1. N/A = Not Applicable
2. The direction of the Port A,B,C, and F 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 E.
4. Control Register setting is not applicable to Ports A, B and C.

Table 37. I/O Port Latched Address Output Assignments

MCU	Port E (PE3-PE0)	Port E (PE7-PE4)	Port F (PF3-PF0)	Port F (PF7-PF4)	Port G (PG3-PG0)	Port G (PG7-PG4)
8051XA	N/A[1]	Address (A7-A4)	N/A[1]	Address (A7-A4)	Address (A11-A8)	Address (A15-A12)
80C251 (Page Mode)	N/A	N/A	N/A	N/A	Address (A11-A8)	Address (A7-A4)
All Other 8-Bit Multiplexed	Address (A3-A0)	Address (A7-A4)	Address (A3-A0)	Address (A7-A4)	Address (A3-A0)	Address (A7-A4)
8-Bit Non-Multiplexed Bus	N/A	N/A	N/A	N/A	Address (A3-A0)	Address (A7-A4)

Note: 1. N/A = Not Applicable.

Address In Mode

For MCUs that have more than 16 address signals, the higher addresses can be connected to Port A, B, C, D or F and are routed as inputs to the PLDs. The address input can be latched in the Input Macrocell (IMC) by Address Strobe (ALE/AS, PD0). Any input that is included in the DPLD equations for the SRAM, or primary or secondary Flash memory is considered to be an address input.

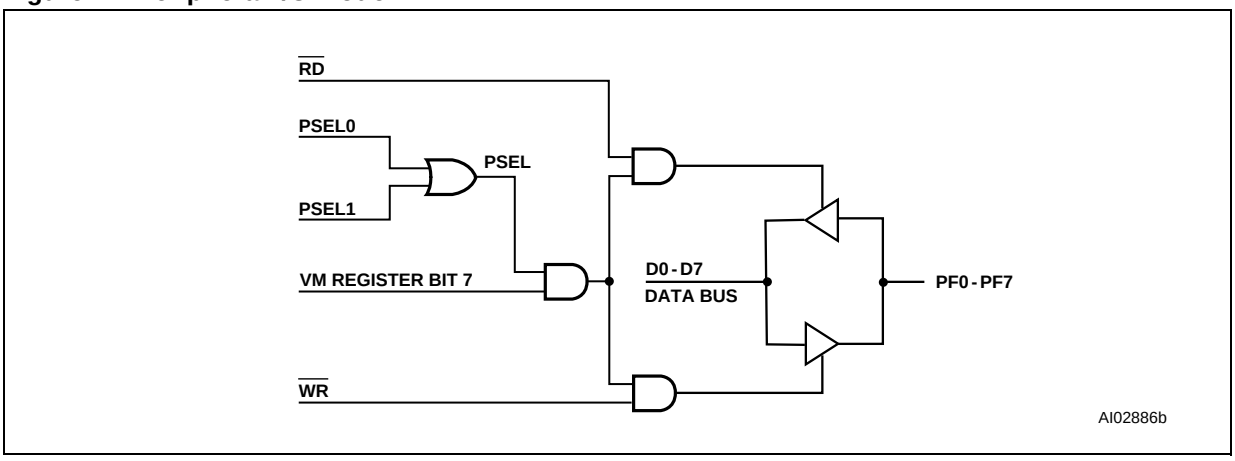
Data Port Mode

Port F can be used as a data bus port for an MCU with a non-multiplexed address/data bus. The Data Port is connected to the data bus of the MCU. The general I/O functions are disabled in Port F if the port is configured as a Data Port. Data Port Mode is automatically configured in PSDsoft when a non-multiplexed bus MCU is selected.

Peripheral I/O Mode

Peripheral I/O mode can be used to interface with external 8-bit peripherals. In this mode, all of Port F serves as a tri-state, bi-directional data buffer for the MCU. Peripheral I/O Mode is enabled by setting Bit 7 of the VM Register to a '1.' Figure 27 shows how Port A acts as a bi-directional buffer for the MCU data bus if Peripheral I/O Mode is enabled. An equation for PSEL0 and/or PSEL1 must be written in PSDsoft. The buffer is tri-stated when PSEL0 or PSEL1 is not active.

Figure 27. Peripheral I/O Mode



JTAG In-System Programming (ISP)

Port E is JTAG compliant, and can be used for In-System Programming (ISP). You can multiplex JTAG operations with other functions on Port E because In-System Programming (ISP) is not performed in normal Operating mode. For more information on the JTAG Port, see PROGRAMMING IN-CIRCUIT USING THE JTAG/ISP INTERFACE, page 76.

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 5., page 19. The addresses in Table 5 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 38, are used for setting the Port configurations. The default Power-up state for each register in Table 38 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 E, F and G have an associated Control Register.

Direction Register

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

Figure 28., page 66 and Figure 29., page 67 show the Port Architecture diagrams for Ports A/B/C and E/F/G, respectively. The direction of data flow for Ports A, B, C and F 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 41. Since Port D only contains four pins (shown in Figure 29., page 67), the Direction Register for Port D has only the four 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.

Note that 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 at a high slew rate when the corresponding bit in the Drive Register is set to '1.' The default rate is slow slew.

Table 42., page 65 shows the Drive Register for Ports A, B, C, D, E and F. It summarizes which pins can be configured as Open Drain outputs and which pins the slew rate can be set for.

Table 38. Port Configuration Registers (PCR)

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

Note: 1. See Table 42., page 65 for Drive Register bit definition.

Table 39. Port Pin Direction Control, Output Enable P.T. Not Defined

Direction Register Bit	Port Pin Mode
0	Input
1	Output

Table 40. 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 41. 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 42. 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	Open Drain	Open Drain	Open Drain	Open Drain
Port B	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain
Port C	Slew Rate	Slew Rate	Slew Rate	Slew Rate	Slew Rate	Slew Rate	Slew Rate	Slew Rate
Port D	NA[1]	NA[1]	NA[1]	NA[1]	Open Drain	Open Drain	Open Drain	Open Drain
Port E	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain
Port F	Slew Rate	Slew Rate	Slew Rate	Slew Rate	Slew Rate	Slew Rate	Slew Rate	Slew Rate
Port G	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain	Open Drain

Note: 1. NA = Not Applicable.

Port Data Registers

The Port Data Registers, shown in Table 43, are used by the MCU to write data to or read data from the ports. Table 43 shows the register name, the ports having each register type, and MCU access for each register type. The registers are described below.

Data In

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

Data Out Register

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

Output Macrocells (OMC)

The CPLD Output Macrocells (OMC) occupy a location in the MCU's address space. The MCU can read the output of the Output Macrocells (OMC). If the OMC Mask Register bits are not set, writing to the macrocell loads data to the macrocell flip-flops. See PLDs, page 38.

OMC Mask Register

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

Table 43. Port Data Registers

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

Input Macrocells (IMC)

The Input Macrocells (IMC) can be used to latch or store external inputs. The outputs of the Input Macrocells (IMC) are routed to the PLD input bus, and can be read by the MCU. See PLDs, page 38.

Enable Out

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

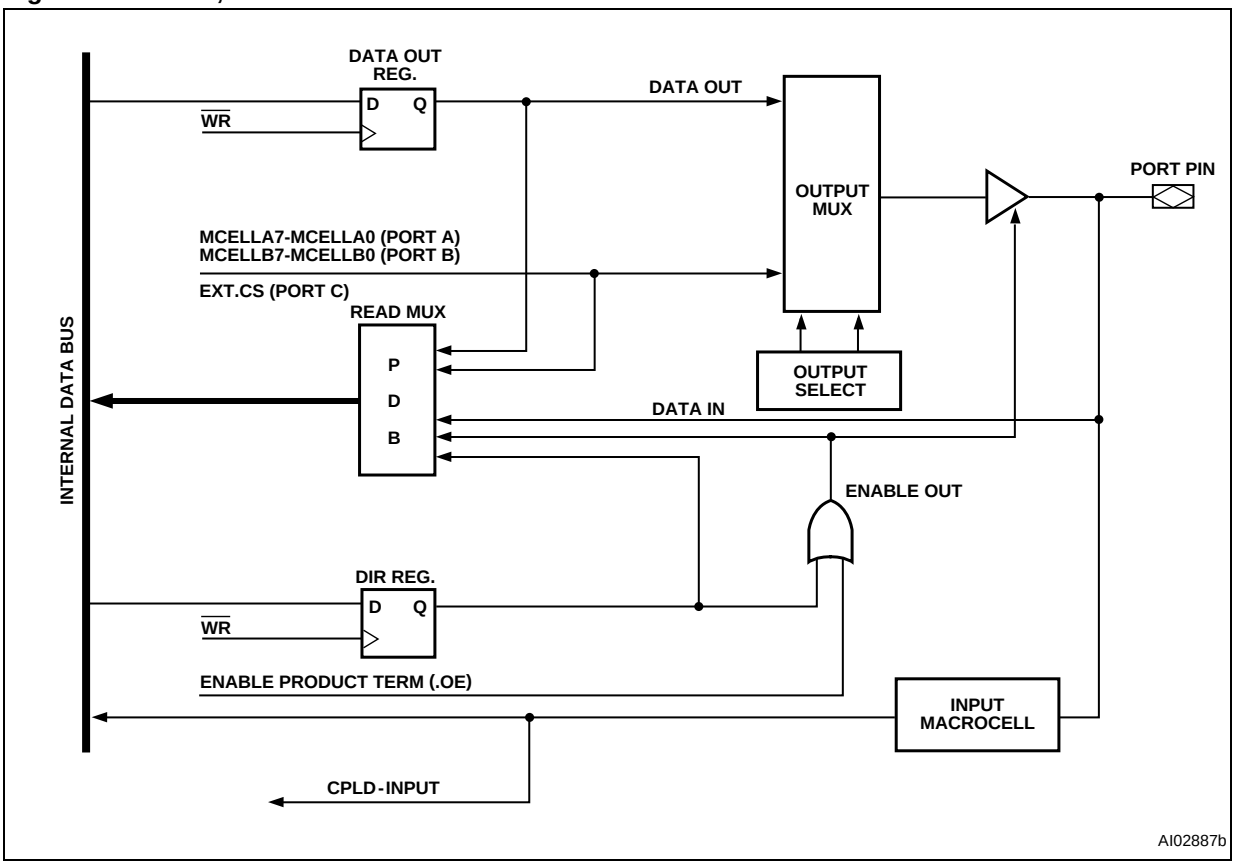
Ports A,B and C – Functionality and Structure

Ports A and B have similar functionality and structure, as shown in Figure 28.

The two ports can be configured to perform one or more of the following functions:

- MCU I/O Mode
- CPLD Output – Macrocells McellA7-McellA0 can be connected to Port A, McellB7-McellB0 can be connected to Port B, External Chip Select ECS7-ECS0 can be connected to Port C.
- CPLD Input – Via the Input Macrocells (IMC).
- Address In – Additional high address inputs using the Input Macrocells (IMC).
- Open Drain/Slew Rate – pins PC7-PC0 can be configured to fast slew rate, pins PA7-PA0 and PB7-PB0 can be configured to Open Drain Mode.

Figure 28. Port A, B and C Structure



Port D – Functionality and Structure

Port D has four I/O pins. It can be configured to program one or more of the following functions (see Figure 29):

– MCU I/O Mode

– CPLD Input – direct input to CPLD, no Input Macrocell (IMC).

Port D pins can be configured in PSDsoft as input pins for other dedicated functions:

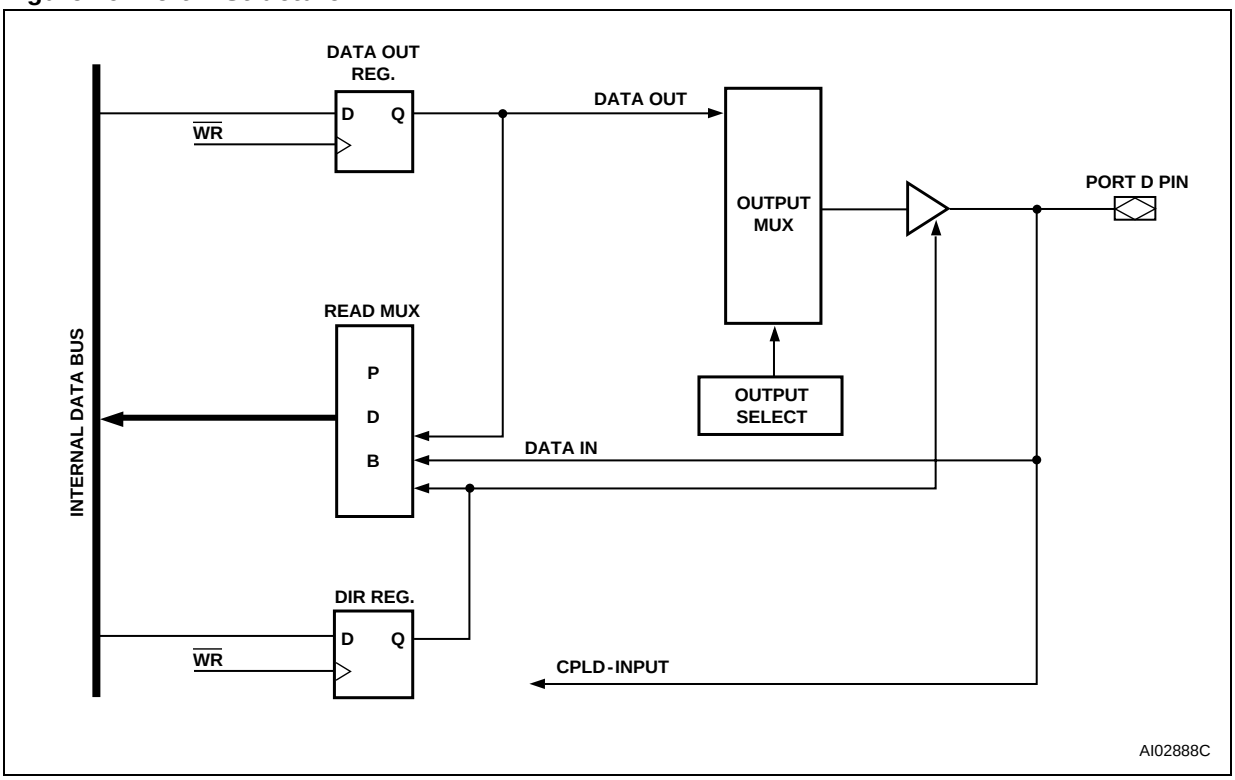
– PD0 – ALE, as Address Strobe input.

– PD1 – CLKIN, as Clock input to the Macrocell flip-flops and APD counter.

– PD2 – CSI, as active Low Chip Select input. A High input will disable the Flash/SRAM memories and the CSIOP.

– PD3 – as DBE input from 68HC912.

Figure 29. Port D Structure



Port E – Functionality and Structure

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

– MCU I/O Mode
– In-System Programming – JTAG port can be enabled for programming/erase of the PSD device. Refer to PROGRAMMING IN-CIRCUIT USING THE JTAG/ISP INTERFACE, page 76 for more information.
– Open Drain – Port E pins can be configured in Open Drain Mode.
– Battery Backup features – PE6 can be configured as a Battery Input (V_{STBY}) pin. PE7 can be configured as a Battery On Indicator output pin, indicating when V_{CC} is less than V_{BAT}.
– Latched Address Output – Provided latched address (A7-A0) output.

Port F – Functionality and Structure

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

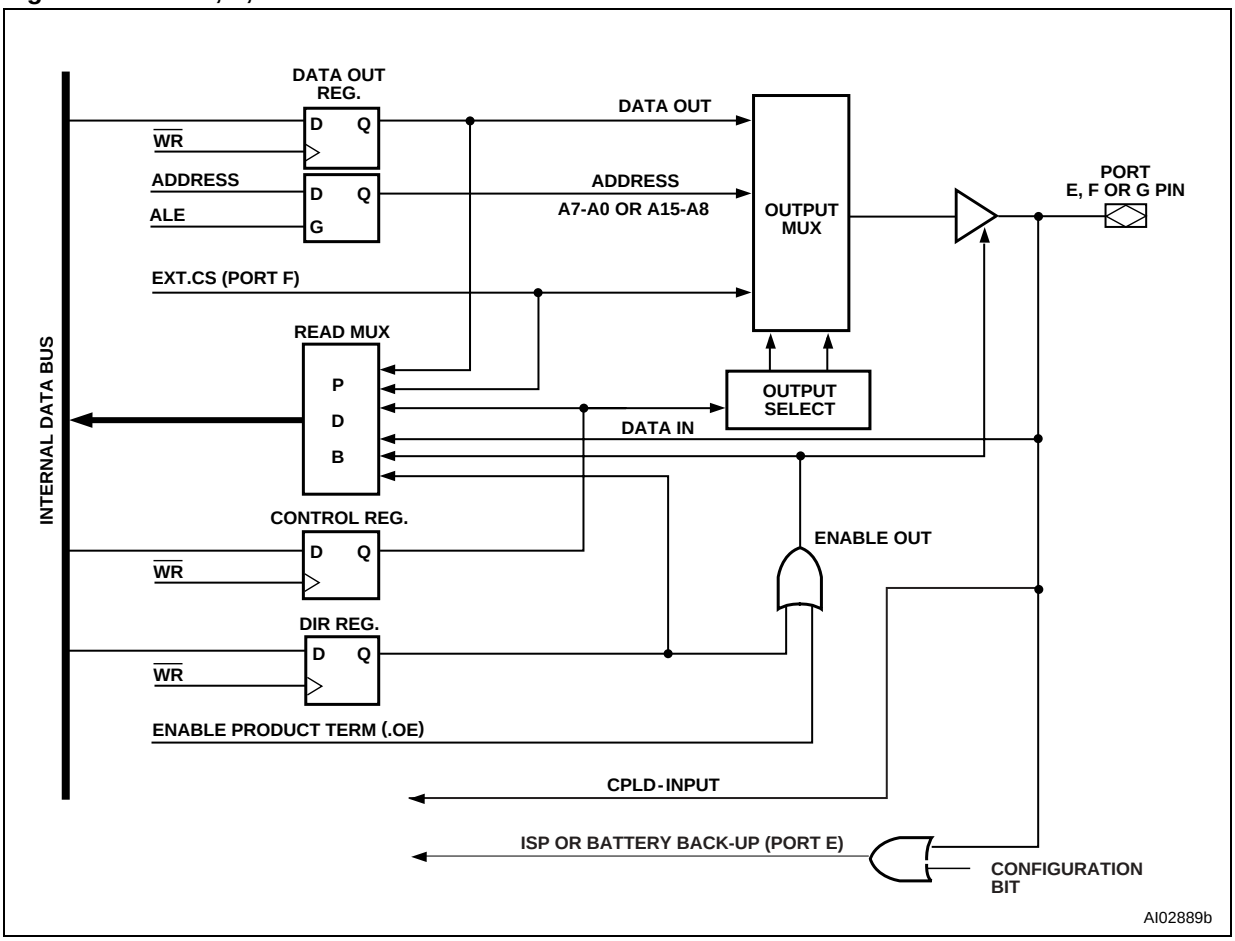
– MCU I/O Mode
– CPLD Output – External Chip Select ECS7-ECS0 can be connected to Port F (or Port C).
– CPLD Input – as direct input of the CPLD array.
– Address In – addition high address inputs. Direct input to the CPLD array, no Input Macrocell (IMC) latching is available.
– Latched Address Out – Provide latched address out per Table 47., page 75.
– Slew Rate – pins can be set up for fast slew rate.
– Data Port – connected to D7-D0 when Port F is configured as Data Port for a non-multiplexed bus.
– Peripheral I/O Mode.

Port G – Functionality and Structure

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

– MCU I/O Mode
– Latched Address Out – Provide latched address out per Table 47., page 75.
– Open Drain – pins can be configured in Open Drain Mode.

Figure 30. Port E, F, G Structure



POWER MANAGEMENT

The PSD835G2 offers configurable power saving options. These options may be used individually or in combinations, as follows:

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

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

– As with the Power Management mode, the Automatic Power Down (APD) unit allows the PSD to reduce to standby current automatically. The APD Unit can also block MCU address/data signals from reaching the memories and PLDs. This feature is available on all the devices of the PSD family. The APD Unit is described in more detail in Automatic Power-down (APD) Unit and Power-down Mode, page 71.

  Built-in logic monitors the Address Strobe of the MCU for activity. If there is no activity for a certain time period (MCU is asleep), the APD Unit initiates Power-down mode (if enabled). Once in Power-down mode, all address/data signals are blocked from reaching PSD memory and PLDs, and the memories are deselected internally. This allows the memory and PLDs to remain in standby mode even if the address/data signals are changing state externally (noise, other devices on the MCU bus, etc.). Keep in mind that any unblocked PLD input signals that are changing states keep the PLD out of Stand-by mode, but not the memories.

– PSD Chip Select Input ($\overline{CSI}$, PD2) can be used to disable the internal memories, placing them in standby mode even if inputs are changing. This feature does not block any internal signals or disable the PLDs. This is a good alternative to using the APD Unit. There is a slight penalty in memory access time when PSD Chip Select Input ($\overline{CSI}$, PD2) makes its initial transition from deselected to selected.

– The PMMRs can be written by the MCU at run-time to manage power. All PSD devices support “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 34., page 78). Significant power savings can be achieved by blocking signals that are not used in PLD logic equations at run-time. PSDsoft creates a fuse map that automatically blocks the low address Byte (A7-A0) or the Control signals (CNTL0-CNTL2, ALE and WRH/DBE) if none of these signals are used in PLD logic equations.

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

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

The APD Unit, shown in Figure 31, 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), Power-down (PDN) goes High, and the PSD enters Power-down mode, as discussed next.

Power-down Mode. By default, if you enable the APD Unit, Power-down mode is automatically enabled. The device enters Power-down mode if Address Strobe (ALE/AS, PD0) remains inactive for fifteen periods of CLKIN (PD1).

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

– If Address Strobe (ALE/AS, PD0) starts pulsing again, the PSD returns to normal Operating mode. The PSD also returns to normal Operating mode if either PSD Chip Select Input (CSI, PD2) is Low or the Reset (RESET) input is 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 CLKIN (PD1). Note that blocking CLKIN (PD1) from the PLDs does not block CLKIN (PD1) from the APD Unit.
– All PSD memories enter Standby mode and are drawing standby current. However, the PLD and I/O ports blocks do *not* go into Standby Mode because you don't want to have to wait for the logic and I/O to "wake-up" before their outputs can change. See Table 44 for Power-down mode effects on PSD ports.
– Typical standby current is of the order of microamperes. These standby current values assume that there are no transitions on any PLD input.

Table 44. 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 31. APD Unit

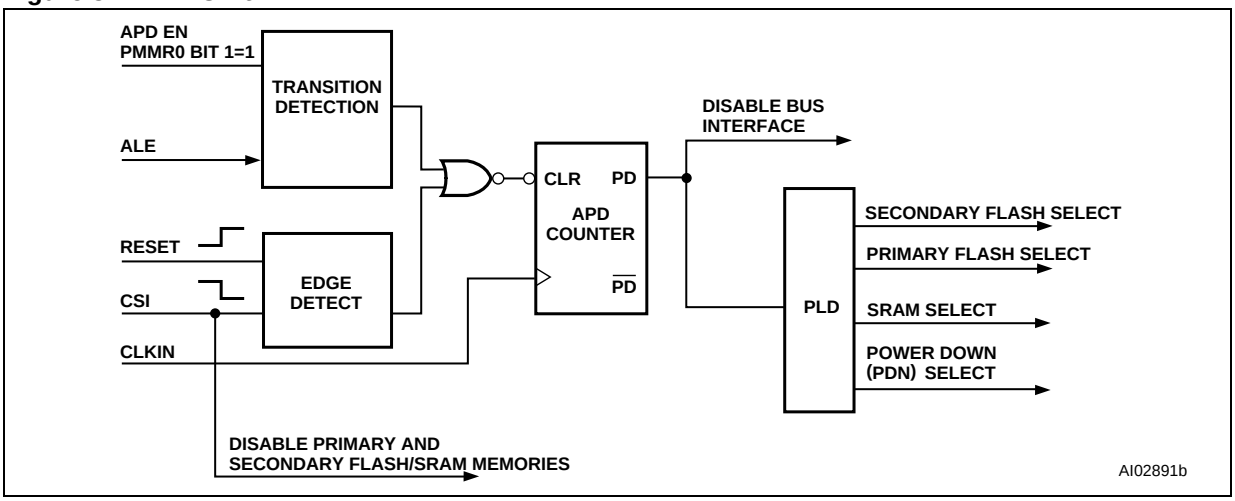


Table 45. PSD Timing and Stand-by Current during Power-down Mode

Mode	PLD Propagation Delay	Memory Access Time	Access Recovery Time to Normal Access	5V V_{CC} Typical Standby Current
Power-down	Normal t_{PD} (Note 1)	No Access	t_{LVDV}	50µA (Note 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.'

Other Power Saving Options

The PSD offers other reduced power saving options that are independent of the Power-down mode. Except for the SRAM Standby and Chip Select Input (CSI, PD2) features, they are enabled by setting bits in the PMMR0 and PMMR2 registers (see Table 22., page 22 and Table 23., page 22 for a bit definition of the two registers).

PLD Power Management

The power and speed of the PLDs are controlled by the Turbo Bit (Bit 3) in PMMR0. By setting the bit to '1,' the Turbo mode is off and the PLDs consume the specified standby current when the inputs are not switching for an extended time of 70ns. The propagation delay time is increased 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 reset to '0' (turned on), the PLDs run at full power and speed. The Turbo Bit affects the PLD's DC power, AC power, and propagation delay. Refer to MAXIMUM RATING, page 81 for PLD timings.

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

SRAM Standby Mode (Battery Backup). The PSD supports a battery backup mode in which the contents of the SRAM are retained in the event of a power loss. The SRAM has a Voltage Standby pin (V_{STBY}, PC2) that can be connected to an external battery. When V_{CC} becomes lower than V_{STBY} then the PSD automatically connects to Voltage Stand-by (V_{STBY}, PC2) as a power source to the SRAM. The SRAM Standby Current (I_{STBY}) is typically 0.5µA. The SRAM data retention voltage is 2 V minimum. The Battery-on Indicator (V_{BATON}) can be routed to PE7. This signal indicates when the V_{CC} has dropped below V_{STBY} and the SRAM is running on battery power.

PSD Chip Select Input (CSI, PD2)

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

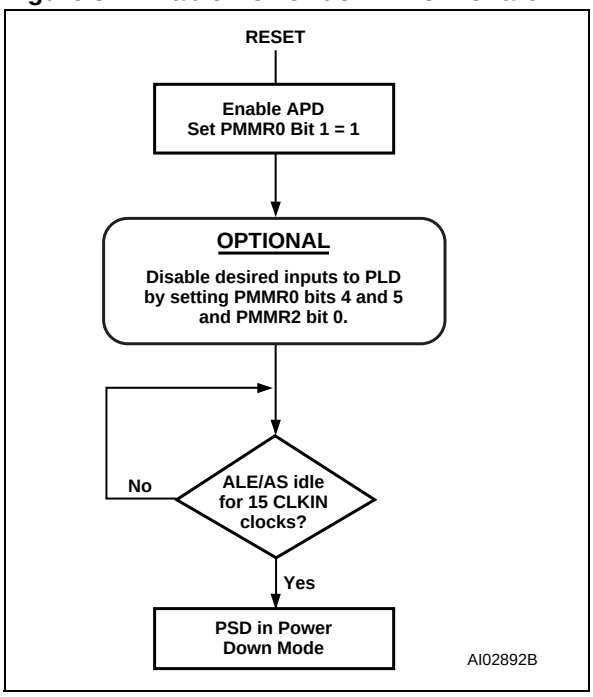
There may be a timing penalty when using PSD Chip Select Input (CSI, PD2) depending on the speed grade of the PSD that you are using. See the timing parameter t_{SLQV} in Table 64., page 93.

Input Clock

The PSD provides the option to turn off CLKIN (PD1) to the PLD to save AC power consumption. CLKIN (PD1) is an input to the PLD AND Array and the Output Macrocells (OMC).

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

Figure 32. Enable Power-down Flow Chart



Input Control Signals

The PSD provides the option to turn off the address input (A7-A0) and input control signals (CNTL0, CNTL1, CNTL2, Address Strobe (ALE/AS, PD0) and DBE) to the PLD to save AC power consumption. These 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 signals should be disabled to save AC power. They are disconnected from the PLD AND Array by setting Bits 0, 2, 3, 4, 5, and 6 to a '1' in PMMR2.

Table 46. 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)

RESET TIMING AND DEVICE STATUS AT RESET

Power-Up Reset

Upon Power-up, the PSD requires a Reset ($\overline{RESET}$) pulse of duration $t_{NLNH-PO}$ (1ms minimum) after V_{CC} is steady. During this period, the device loads internal configurations, clears some of the registers and sets the Flash memory into Operating mode. After the rising edge of Reset ($\overline{RESET}$), the PSD remains in the Reset mode for an additional period, t_{OPR} (120ns maximum), before the first memory access is allowed.

The Flash memory is reset to the READ mode upon Power-up. Sector Select (FS0-FS7 and CSBOOT0-CSBOOT3) must all be Low, Write Strobe (WR, CNTL0) High, 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 Write Strobe (WR, CNTL0). 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 pulse of a much shorter duration, t_{NLNH} (150ns minimum). The same t_{OPR} period is needed before the device is operational after warm reset. Figure 33 shows the timing of the Power-up and warm reset.

I/O Pin, Register and PLD Status at Reset

Table 47., page 75 shows the I/O pin, register and PLD status during Power-Up Reset, warm reset and Power-down mode. PLD outputs are always valid during warm reset, and they are valid in Power-Up Reset once the internal PSD Configuration bits are loaded. This loading of PSD is completed typically long before V_{CC} ramps up to operating level. Once the PLD is active, the state of the outputs are determined by the equations specified in PSDsoft.

Reset of Flash Memory Erase and Program Cycles

A Reset ($\overline{RESET}$) also resets the internal Flash memory state machine. During a Flash memory Program or Erase cycle, Reset ($\overline{RESET}$) terminates the cycle and returns the Flash memory to the READ mode within a period of t_{NLNH-A} (25µs minimum).

Figure 33. Power-Up and Warm Reset ($\overline{RESET}$) Timing

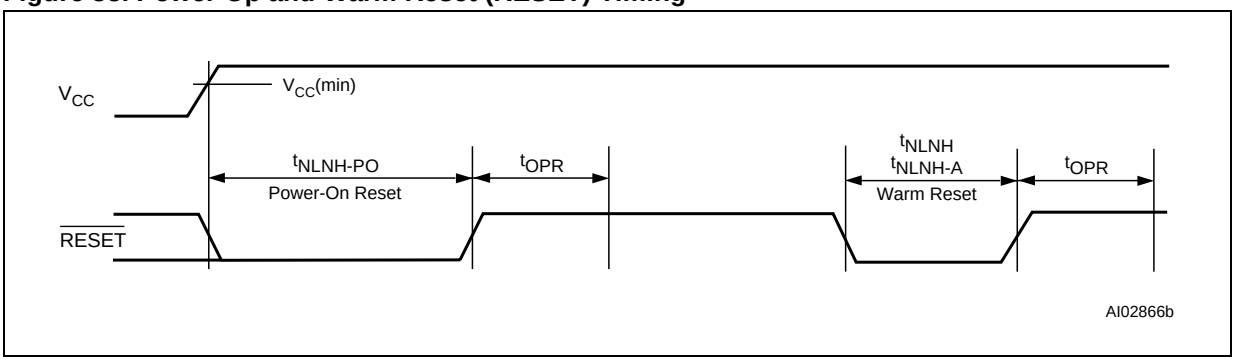


Table 47. Status During Power-Up Reset, Warm Reset and Power-down Mode

Port Configuration	Power-Up 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-Un Reset**	**Warm Reset**	**Power-down Mode**
PMMR0 and PMMR2	Cleared to '0'	Unchanged	Unchanged
Macrocells flip-flop status	Cleared to '0' by internal Power-Up Reset	Depends on .re and .pr equations	Depends on .re and .pr equations
VM Register[1]	Initialized, based on the selection in PSDsoft Configuration menu	Initialized, based on the selection in PSDsoft Configuration menu	Unchanged
All other registers	Cleared to '0'	Cleared to '0'	Unchanged

Note: 1. The SR_cod and PeriphMode Bits in the VM Register are always cleared to '0' on Power-Up Reset or Warm Reset.

PROGRAMMING IN-CIRCUIT USING THE JTAG/ISP INTERFACE

The JTAG/ISP Interface block can be enabled on Port E (see Table 48., page 77). All memory blocks (primary and secondary Flash memory), PLD logic, and PSD Configuration Register bits may be programmed through the JTAG/ISP Interface block. A blank device can be mounted on a printed circuit board and programmed using JTAG/ISP.

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

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

See Application Note *AN1153* for more details on JTAG In-System Programming (ISP).

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 JTAG serial command from an external JTAG controller device (such as FlashLINK or Automated Test Equipment). When the enabling command is received, TDO becomes an output and the JTAG channel is fully functional inside the PSD. The same command that enables the JTAG channel may optionally enable the two additional JTAG signals, TSTAT and TERR.

The following symbolic logic equation specifies the conditions enabling the four basic JTAG signals (TMS, TCK, TDI, and TDO) on their respective Port E pins. For purposes of discussion, the logic label JTAG_ON is 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 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
    20., page 22 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 E 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. See Application Note 1153
    for details. */
```

The PSD supports JTAG/ISP commands, but not Boundary Scan. The PSDsoft software tool and FlashLINK JTAG programming cable implement the JTAG/ISP commands.

JTAG Extensions

TSTAT and $\overline{\text{TERR}}$ are two JTAG extension signals enabled by an JTAG command received over the four standard JTAG signals (TMS, TCK, TDI, and TDO). They are used to speed Program and Erase cycles by indicating status on PSD signals instead of having to scan the status out serially using the standard JTAG channel. See Application Note *AN1153*.

$\overline{\text{TERR}}$ indicates if an error has occurred when erasing a sector or programming a Byte in Flash memory. This signal goes Low (active) when an Error condition occurs, and stays Low until a special JTAG command is executed or a chip Reset ($\overline{\text{RESET}}$) pulse is received after an "ISC_DISABLE" command.

TSTAT behaves the same as Ready/$\overline{\text{Busy}}$ described in the section entitled Ready/Busy (PE4), page 26. TSTAT is High when the PSD device is in READ mode (primary and secondary Flash memory contents can be read). TSTAT is Low when Flash memory Program or Erase cycles are in progress, and also when data is being written to the secondary Flash memory.

TSTAT and $\overline{\text{TERR}}$ can be configured as open-drain type signals during a JTAG command.

Security and Flash memory Protection

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

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

All primary and secondary Flash memory sectors can individually be sector protected against erasures. The Sector Protect Bits can be set in PSDsoft.

Table 48. JTAG Port Signals

Port E Pin	JTAG Signals	Description
PE0	TMS	Mode Select
PE1	TCK	Clock
PE2	TDI	Serial Data In
PE3	TDO	Serial Data Out
PE4	TSTAT	Status
PE5	$\overline{\text{TERR}}$	Error Flag

AC/DC PARAMETERS

The tables provided below 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, Flash memory, and SRAM mA/MHz specification. Figure 34 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 34. PLD I_{CC} /Frequency Consumption

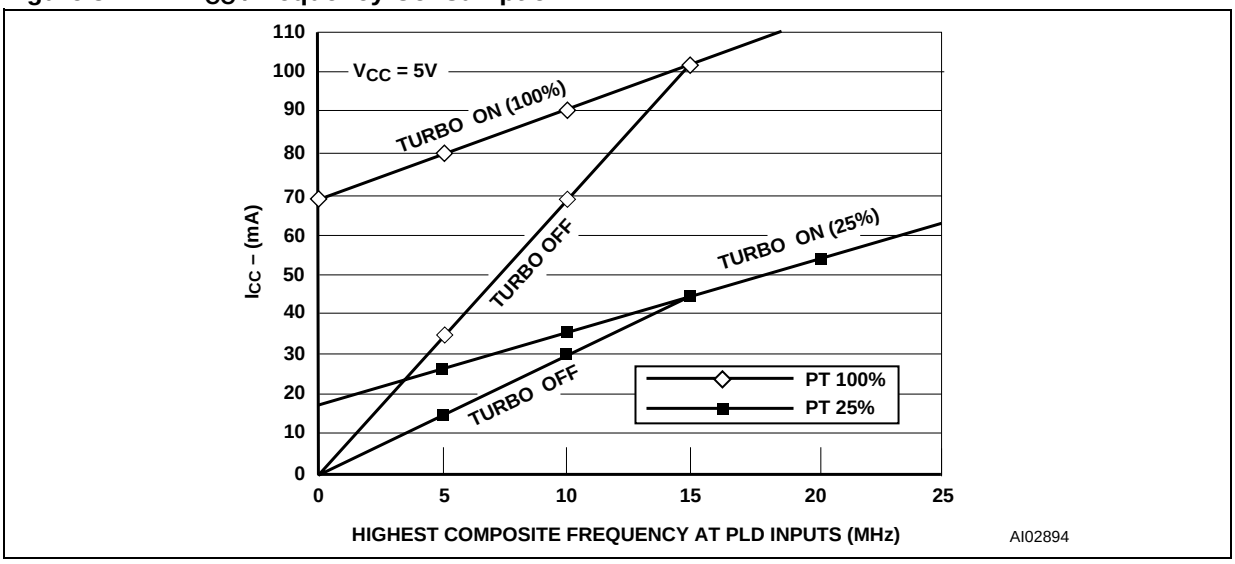


Table 49. Example of PSD Typical Power Calculation at V_{CC} = 5.0V (with Turbo Mode On)

Conditions			
Highest Composite PLD input frequency			
	(Freq PLD)	= 8 MHz	
MCU ALE frequency (Freq ALE)		= 4 MHz	
	% 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	= 45/193 = 23.3%	
	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 4 MHz	
			+ 0.15 x 1.5mA/MHz x 4 MHz
			+ 2mA/MHz x 8 MHz
			+ 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 Flash memory WRITE or Erase cycles in progress. Calculation is based on I_{OUT} = 0mA.			

Table 50. Example of PSD Typical Power Calculation at V_{CC} = 5.0V (with Turbo Mode Off)

Conditions			
Highest Composite PLD input frequency			
	(Freq PLD)	= 8 MHz	
MCU ALE frequency (Freq ALE)		= 4 MHz	
	% 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	= 45/193 = 23.3%	
	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 4 MHz	
			+ 0.15 x 1.5mA/MHz x 4 MHz
			+ 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 Flash memory WRITE or Erase cycles in progress. Calculation is based on I_{OUT} = 0mA.			

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 51. 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)

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 52. Operating Conditions

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 53. AC Signal Letters for PLD Timing

A	Address Input
C	CEout Output
D	Input Data
E	E Input
I	Interrupt Input
L	ALE Input
N	$\overline{\text{RESET}}$ Input or Output
P	Port Signal Output
Q	Output Data
R	$\overline{\text{UDS}}$, $\overline{\text{LDS}}$, $\overline{\text{DS}}$, $\overline{\text{RD}}$, $\overline{\text{PSEN}}$ Inputs
S	Chip Select Input
T	R/$\overline{\text{W}}$ Input
W	WR Input
B	V_{STBY} Output
M	Output Macrocell

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

Table 54. AC Signal Behavior Symbols for PLD Timing

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 55. 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.

Table 56. 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 35. AC Measurement I/O Waveform

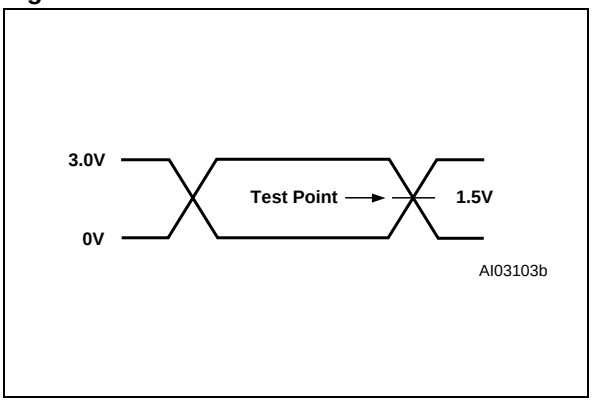


Figure 36. AC Measurement Load Circuit

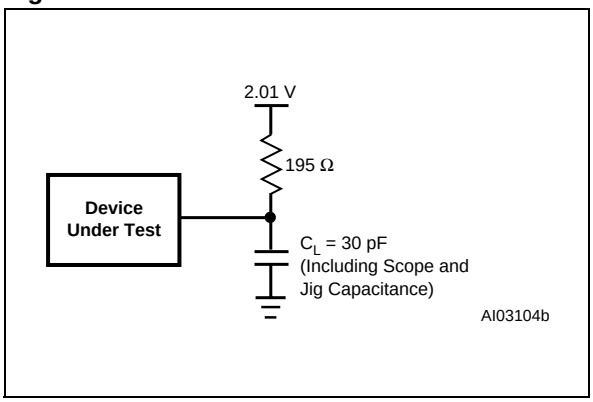


Figure 37. Switching Waveforms – Key

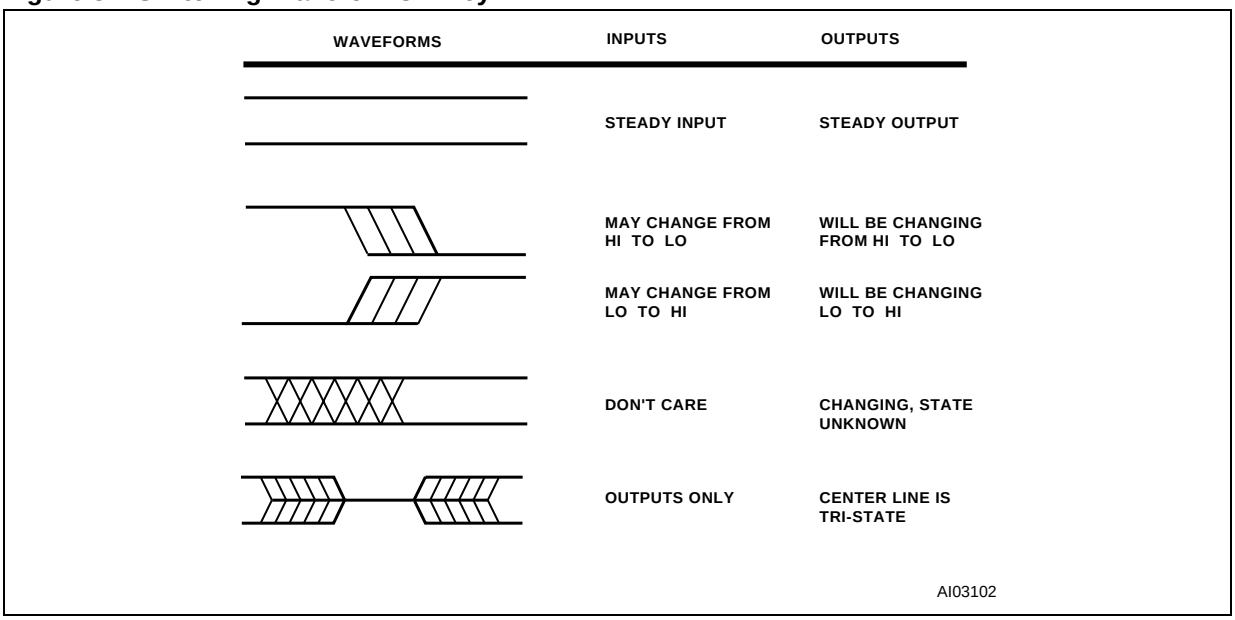


Table 57. DC Characteristics

Symbol	Parameter		Test Condition (in addition to those in Table 52., page 82)	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
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 Except V_{STBY} On		I_{OH} = –20µA, V_{CC} = 4.5V	4.4	4.49		V
			I_{OH} = –2mA, V_{CC} = 4.5V	2.4	3.9		V
V_{OH1}	Output High Voltage V_{STBY} On		I_{OH1} = –1µA	V_{STBY} – 0.8			V
V_{STBY}	SRAM Stand-by Voltage			2.0		V_{CC}	V
I_{STBY}	SRAM Stand-by Current		V_{CC} = 0V		0.5	1	µA
I_{IDLE}	Idle Current (V_{STBY} input)		V_{CC} > V_{STBY}	–0.1		0.1	µA
V_{DF}	SRAM Data Retention Voltage		Only on V_{STBY}	2			V
I_{SB}	Stand-by Supply Current for Power-down Mode		$\overline{CSI}$ > V_{CC} – 0.3V (Notes 2,3,5)		100	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_{IN} < V_{CC}	–10	±5	10	µA
I_{CC} (DC) (Note5)	Operating Supply Current	PLD Only	PLD_TURBO = Off, f = 0 MHz (Note 3)		0		µA/PT
			PLD_TURBO = On, f = 0 MHz		400	700	µA/PT
		Flash memory	During Flash memory WRITE/Erase Only		15	30	mA
			Read only, f = 0 MHz		0	0	mA
		SRAM	f = 0 MHz		0	0	mA
I_{CC} (AC) (Note5)	PLD AC Base			Figure 34 (note4)			
	Flash memory 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 34 for the PLD current calculation.
5. I_{OUT} = 0mA

Figure 38. Input to Output Disable / Enable

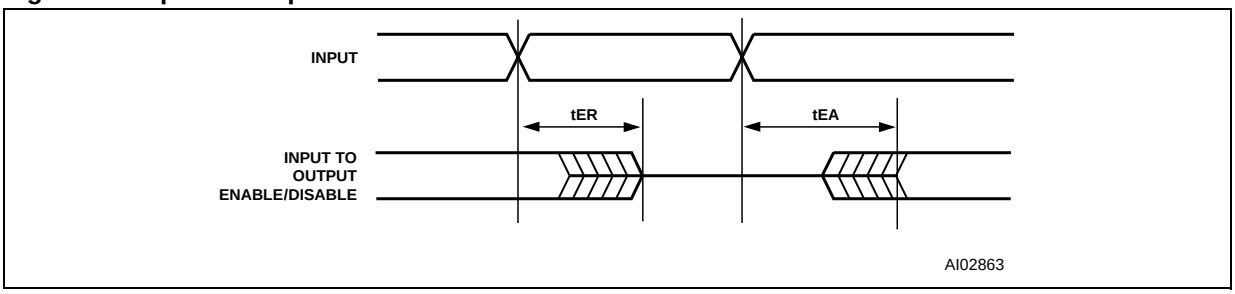


Figure 39. Combinatorial Timing – PLD

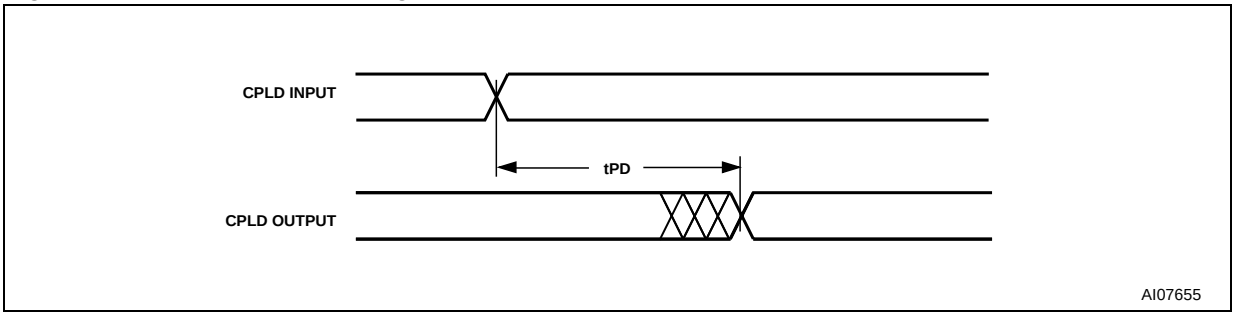


Table 58. CPLD Combinatorial Timing

Symbol	Parameter	Conditions	-70		-90		PT Aloc	Turbo Off	Slew rate[1]	Unit
			Min	Max	Min	Max				
t_{PD}	CPLD Input Pin/Feedback to CPLD Combinatorial Output			20		25	+ 2	+ 12	– 2	ns
t_{EA}	CPLD Input to CPLD Output Enable			21		26		+ 12	– 2	ns
t_{ER}	CPLD Input to CPLD Output Disable			21		26		+ 12	– 2	ns
t_{ARP}	CPLD Register Clear or Preset Delay			21		26		+ 12	– 2	ns
t_{ARPW}	CPLD Register Clear or Preset Pulse Width		10		20			+ 12		ns
t_{ARD}	CPLD Array Delay	Any macrocell		11		16	+ 2			ns

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

Table 59. CPLD Macrocell Synchronous Clock Mode Timing

Symbol	Parameter	Conditions	-70		-90		PT Aloc	Turbo Off	Slew rate[1]	Unit
			Min	Max	Min	Max				
f_{MAX}	Maximum Frequency External Feedback	$1/(t_S+t_{CO})$		34.4		30.30				MHz
	Maximum Frequency Internal Feedback (f_{CNT})	$1/(t_S+t_{CO}–10)$		52.6		43.48				MHz
	Maximum Frequency Pipelined Data	$1/(t_{CH}+t_{CL})$		83.3		50.00				MHz
t_S	Input Setup Time		14		15		+ 2	+ 12		ns
t_H	Input Hold Time		0		0					ns
t_{CH}	Clock High Time	Clock Input	6		10					ns
t_{CL}	Clock Low Time	Clock Input	6		10					ns
t_{CO}	Clock to Output Delay	Clock Input		15		18			– 2	ns
t_{ARD}	CPLD Array Delay	Any macrocell		11		16	+ 2			ns
t_{MIN}	Minimum Clock Period [2]	$t_{CH}+t_{CL}$	12		20					ns

Note: 1. Fast Slew Rate output available on Ports C and F.
2. CLKIN (PD1) $t_{CLCL} = t_{CH} + t_{CL}$.

Table 60. CPLD Macrocell Asynchronous Clock Mode Timing

Symbol	Parameter	Conditions	-70		-90		PT Aloc	Turbo Off	Slew Rate	Unit
			Min	Max	Min	Max				
f_{MAXA}	Maximum Frequency External Feedback	$1/(t_{SA}+t_{COA})$		38.4		26.32				MHz
	Maximum Frequency Internal Feedback (f_{CNTA})	$1/(t_{SA}+t_{COA}–10)$		62.5		35.71				MHz
	Maximum Frequency Pipelined Data	$1/(t_{CHA}+t_{CLA})$		47.6		37.3				MHz
t_{SA}	Input Setup Time		6		8		+ 2	+ 12		ns
t_{HA}	Input Hold Time		7		12					ns
t_{CHA}	Clock Input High Time		9		12			+ 12		ns
t_{CLA}	Clock Input Low Time		12		15			+ 12		ns
t_{COA}	Clock to Output Delay			21		30		+ 12	– 2	ns
t_{ARDA}	CPLD Array Delay	Any macrocell		11		16	+ 2			ns
t_{MINA}	Minimum Clock Period	$1/f_{CNTA}$	16		28					ns

Figure 40. Synchronous Clock Mode Timing – PLD

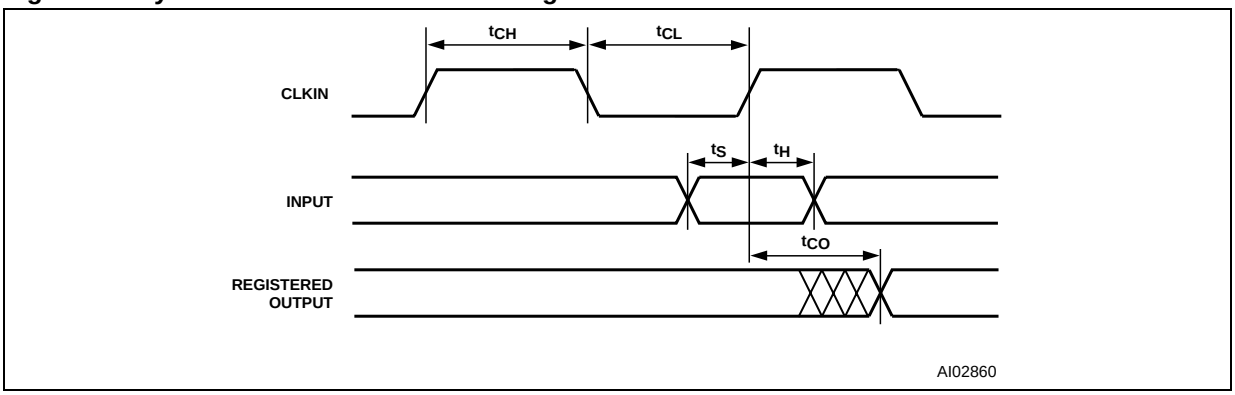


Figure 41. Asynchronous Reset / Preset

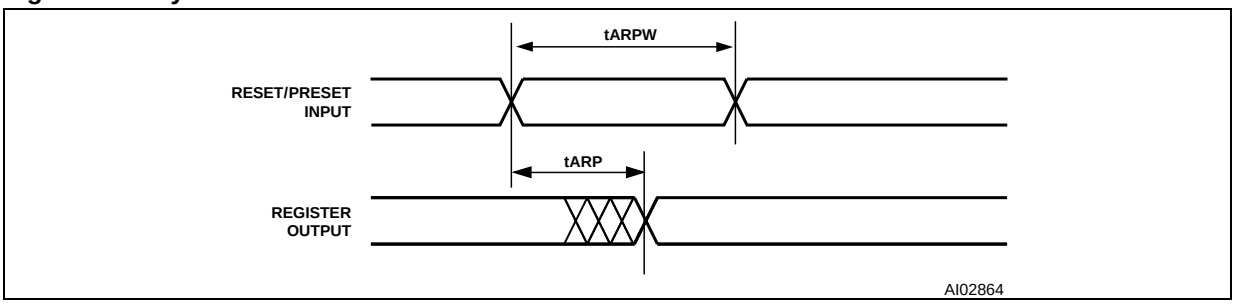


Figure 42. Asynchronous Clock Mode Timing (Product Term Clock)

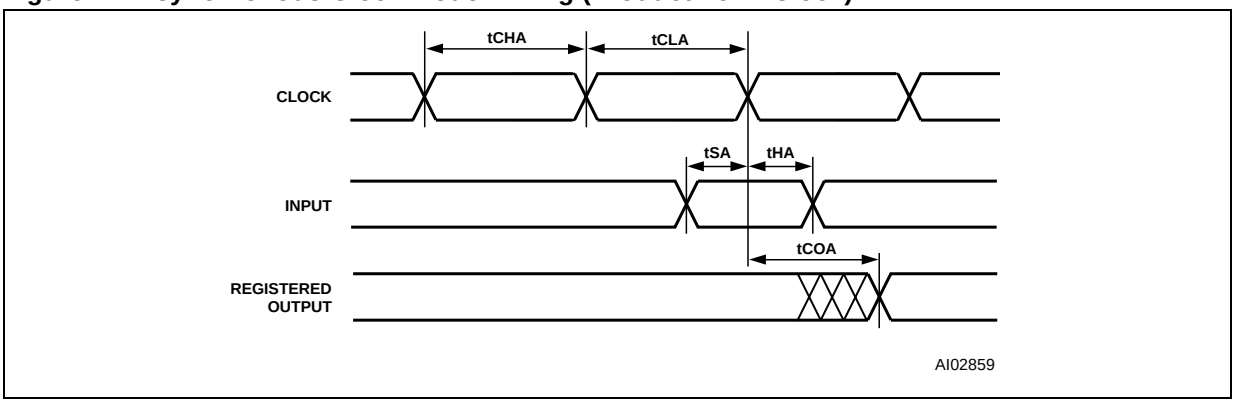


Figure 43. Input Macrocell Timing (Product Term Clock)

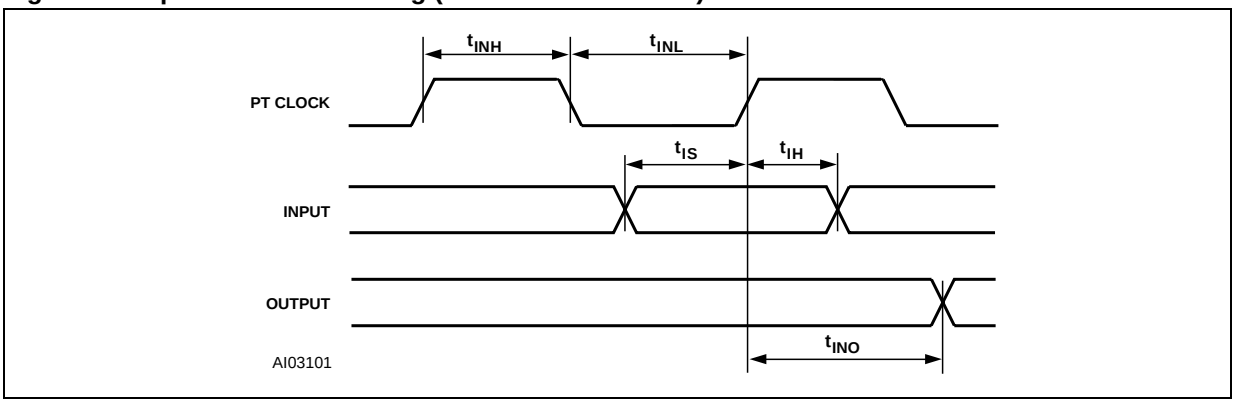
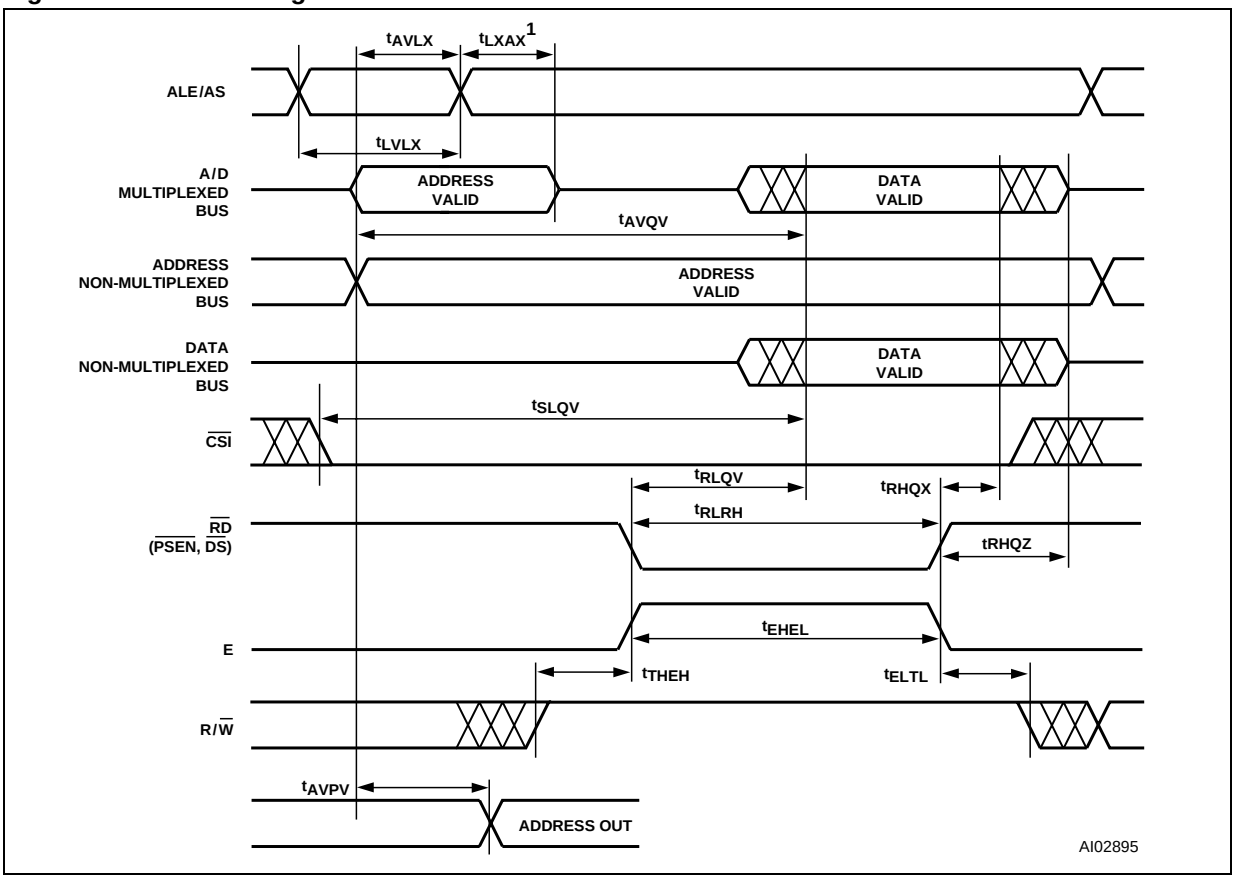


Table 61. Input Macrocell Timing

Symbol	Parameter	Conditions	-70		-90		PT Aloc	Turbo Off	Unit
			Min	Max	Min	Max			
t_{IS}	Input Setup Time	(Note 1)	0		0				ns
t_{IH}	Input Hold Time	(Note 1)	15		20			+ 12	ns
t_{INH}	NIB Input High Time	(Note 1)	9		12				ns
t_{INL}	NIB Input Low Time	(Note 1)	9		12				ns
t_{INO}	NIB Input to Combinatorial Delay	(Note 1)		34		46	+ 2	+ 12	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}.

Figure 44. READ Timing



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

Table 62. READ Timing

Symbol	Parameter	Conditions	-70		-90		Turbo Off	Unit
			Min	Max	Min	Max		
t_{LVLX}	ALE or AS Pulse Width		15		20			ns
t_{AVLX}	Address Setup Time	(Note 3)	4		6			ns
t_{LXAX}	Address Hold Time	(Note 3)	7		8			ns
t_{AVQV}	Address Valid to Data Valid	(Note 3)		70		90	+ 12	ns
t_{SLQV}	CS Valid to Data Valid			75		100		ns
t_{RLQV}	$\overline{RD}$ to Data Valid 8-Bit Bus	(Note 5)		24		32		ns
	$\overline{RD}$ or $\overline{PSEN}$ to Data Valid 8-Bit Bus, 8031, 80251	(Note 2)		31		38		ns
t_{RHQX}	$\overline{RD}$ Data Hold Time	(Note 1)	0		0			ns
t_{RLRH}	$\overline{RD}$ Pulse Width	(Note 1)	27		32			ns
t_{RHQZ}	$\overline{RD}$ to Data High-Z	(Note 1)		20		25		ns
t_{EHEL}	E Pulse Width		27		32			ns
t_{THEH}	R/$\overline{W}$ Setup Time to Enable		6		10			ns
t_{ELTL}	R/$\overline{W}$ Hold Time After Enable		0		0			ns
t_{AVPV}	Address Input Valid to Address Output Delay	(Note 4)		20		25		ns

Note: 1. $\overline{RD}$ timing has the same timing as $\overline{DS}$ 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 $\overline{DS}$ signal.

Figure 45. WRITE Timing

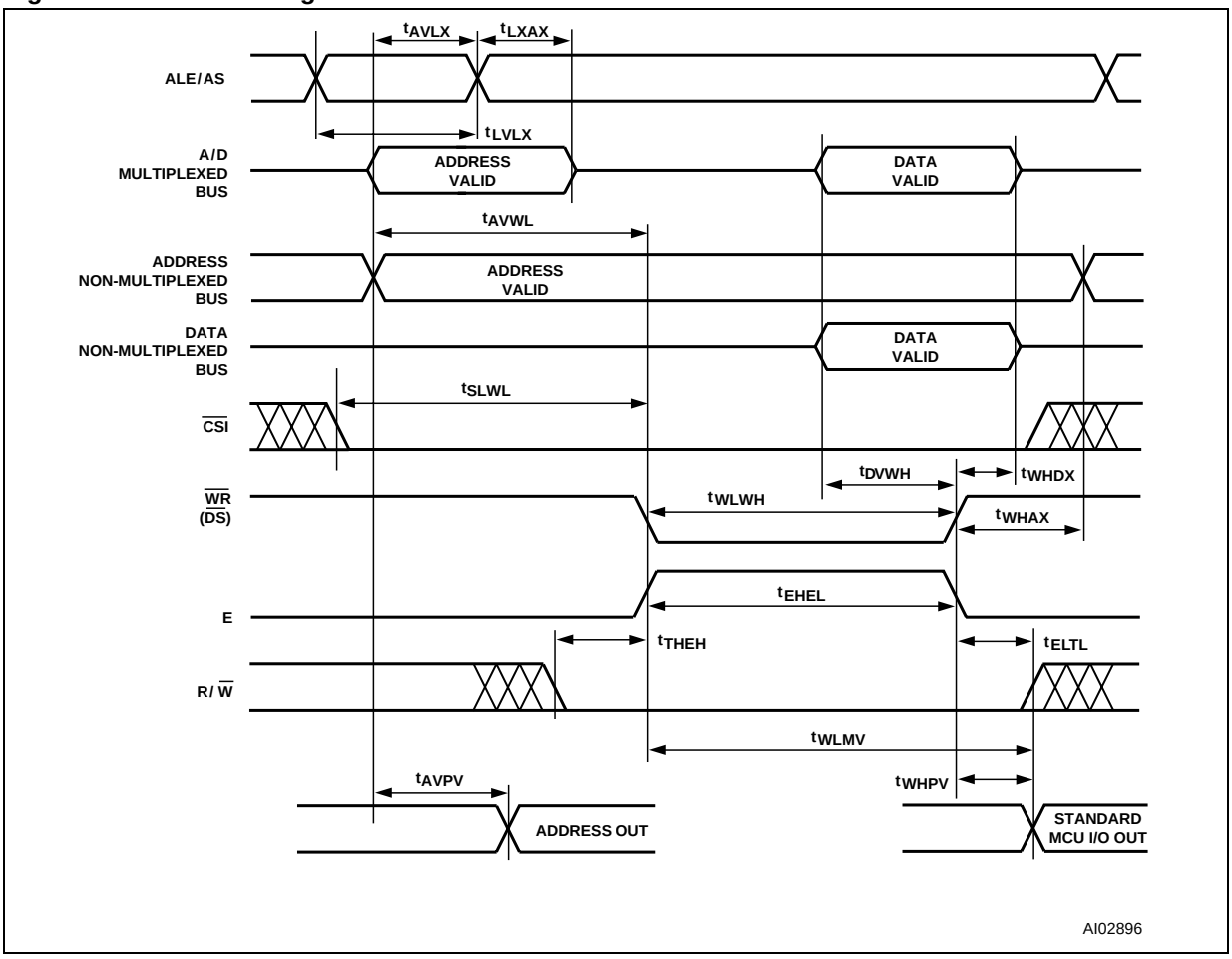


Table 63. WRITE Timing

Symbol	Parameter	Conditions	-70		-90		Unit
			Min	Max	Min	Max	
t_{LVLX}	ALE or AS Pulse Width		15		20		ns
t_{AVLX}	Address Setup Time	(Note 1)	4		6		ns
t_{LXAX}	Address Hold Time	(Note 1)	7		8		ns
t_{AVWL}	Address Valid to Leading Edge of $\overline{WR}$	(Notes 1,3)	8		15		ns
t_{SLWL}	$\overline{CS}$ Valid to Leading Edge of $\overline{WR}$	(Note 3)	12		15		ns
t_{DVWH}	$\overline{WR}$ Data Setup Time	(Note 3)	25		35		ns
t_{WHDX}	$\overline{WR}$ Data Hold Time	(Note 3,7)	4		5		ns
t_{WLWH}	$\overline{WR}$ Pulse Width	(Note 3)	28		35		ns
t_{WHAX1}	Trailing Edge of $\overline{WR}$ to Address Invalid	(Note 3)	6		8		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)		27		30	ns
t_{DVMV}	Data Valid to Port Output Valid Using Macrocell Register Preset/Clear	(Notes 3,5)		42		55	ns
t_{AVPV}	Address Input Valid to Address Output Delay	(Note 2)		20		25	ns
t_{WLMV}	$\overline{WR}$ Valid to Port Output Valid Using Macrocell Register Preset/Clear	(Notes 3,4)		48		55	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 and $\overline{DS}$ 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.
7. t_{WHDX} is 6ns when writing to Output Macrocell Registers AB and BC.

Figure 46. Peripheral I/O Read Timing

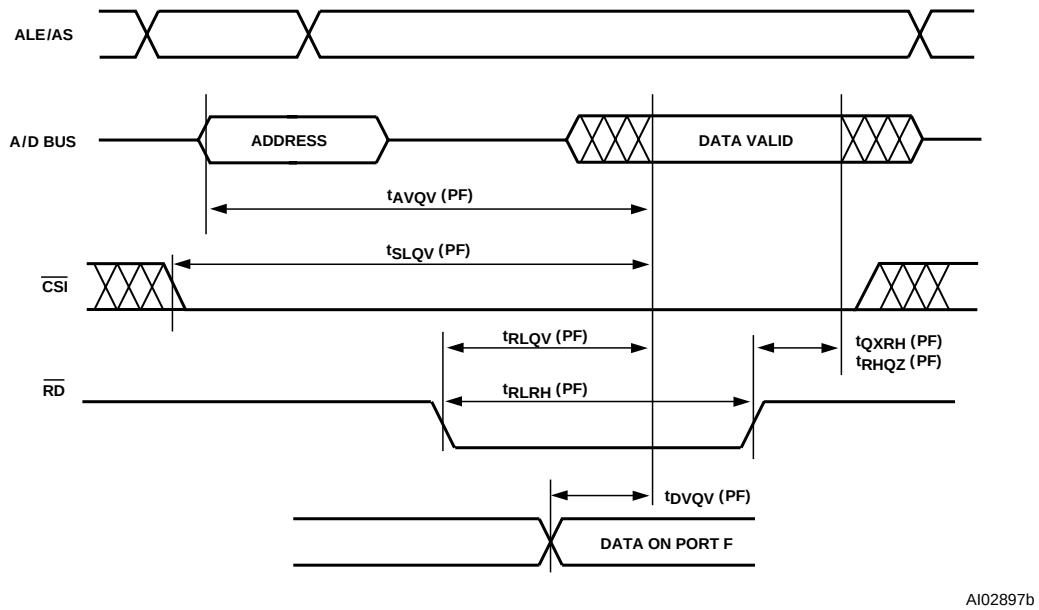


Table 64. Port F Peripheral Data Mode Read Timing

Symbol	Parameter	Conditions	-70		-90		Turbo Off	Unit
			Min	Max	Min	Max		
$t_{AVQV-PF}$	Address Valid to Data Valid	(Note 3)		30		35	+ 12	ns
$t_{SLQV-PF}$	$\overline{CSI}$ Valid to Data Valid			25		35	+ 12	ns
$t_{RLQV-PF}$	$\overline{RD}$ to Data Valid	(Notes 1,4)		21		32		ns
	$\overline{RD}$ to Data Valid 8031 Mode			31		38		ns
$t_{DVQV-PF}$	Data In to Data Out Valid			22		30		ns
$t_{QXRH-PF}$	$\overline{RD}$ Data Hold Time		0		0			ns
$t_{RLRH-PF}$	$\overline{RD}$ Pulse Width	(Note 1)	27		32			ns
$t_{RHQZ-PF}$	$\overline{RD}$ to Data High-Z	(Note 1)		23		25		ns

Figure 47. Peripheral I/O Write Timing

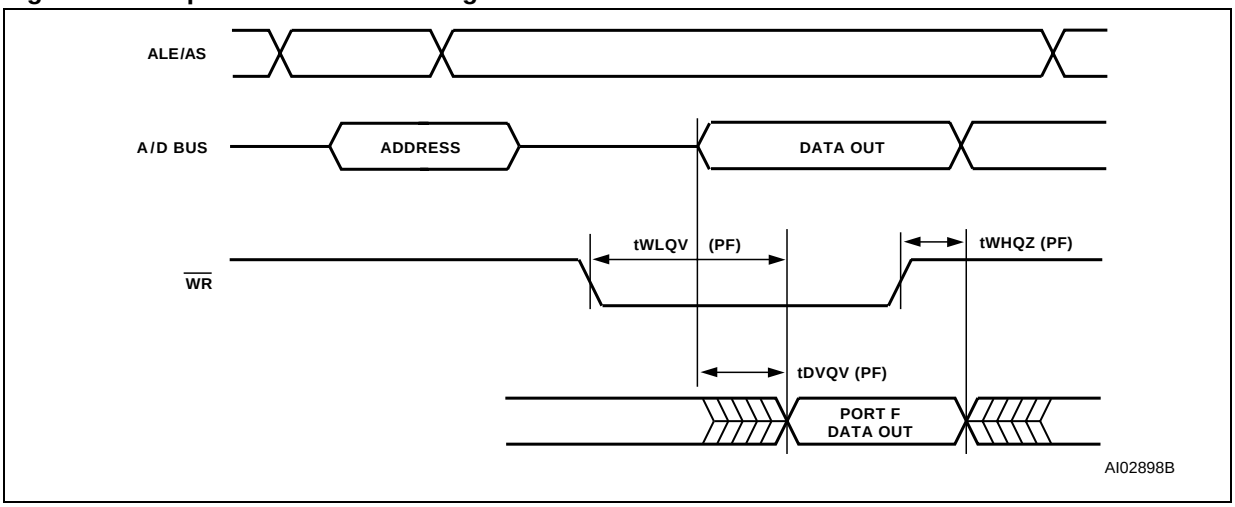


Table 65. Port F Peripheral Data Mode Write Timing

Symbol	Parameter	Conditions	-70		-90		Unit
			Min	Max	Min	Max	
$t_{WLQV-PF}$	$\overline{WR}$ to Data Propagation Delay	(Note 2)		25		35	ns
$t_{DVQV-PF}$	Data to Port A Data Propagation Delay	(Note 5)		22		30	ns
$t_{WHQZ-PF}$	$\overline{WR}$ Invalid to Port A Tri-state	(Note 2)		20		25	ns

Note: 1. $\overline{RD}$ has the same timing as $\overline{DS}$ and $\overline{PSEN}$.
2. $\overline{WR}$ has the same timing as the E and $\overline{DS}$ signals.
3. Any input used to select Port F Data Peripheral mode.
4. Data is already stable on Port F.
5. Data stable on ADIO pins to data on Port F.

Table 66. Program, Write and Erase Times

Symbol	Parameter	Min.	Typ.	Max.	Unit
	Flash Program		8.5		s
	Flash Bulk Erase[1] (pre-programmed to “00”)		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 to “00”)		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 tQ7VQV time units before the data byte, DQ0-DQ7, is valid for reading.

Table 67. Power-down Timing

Symbol	Parameter	Conditions	-70		-90		Unit
			Min	Max	Min	Max	
t_{LVDV}	ALE Access Time from Power-down			80		90	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).

Figure 48. Reset ($\overline{\text{RESET}}$) Timing

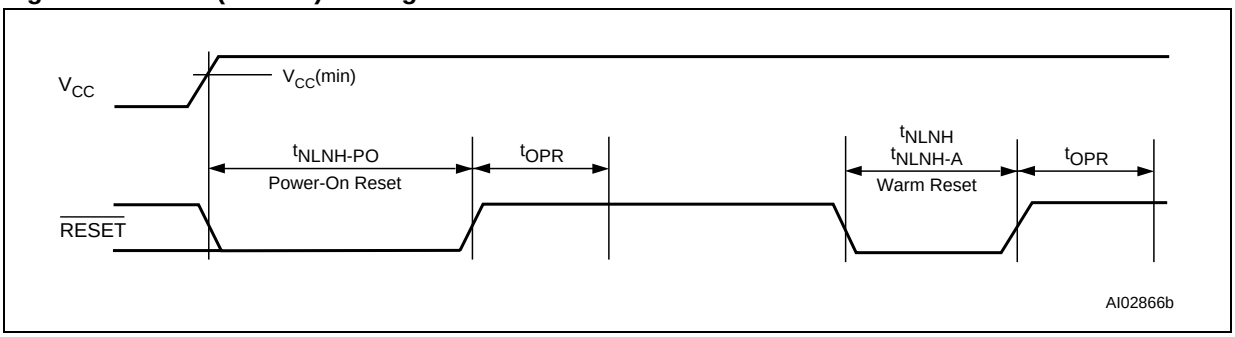


Table 68. Reset ($\overline{\text{Reset}}$) Timing

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_{NLNH-A}	Warm Reset [2]		25		µs
t_{OPR}	RESET High to Operational Device			120	ns

Note: 1. Reset ($\overline{\text{RESET}}$) 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 69. V_{STBYON} Timing

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
t_{BVBH}	V_{STBY} Detection to V_{STBYON} Output High	(Note [1])		20		µs
t_{BXBL}	V_{STBY} Off Detection to V_{STBYON} Output Low	(Note [1])		20		µs

Note: 1. V_{STBYON} timing is measured at V_{CC} ramp rate of 2 ms.

Figure 49. ISC Timing

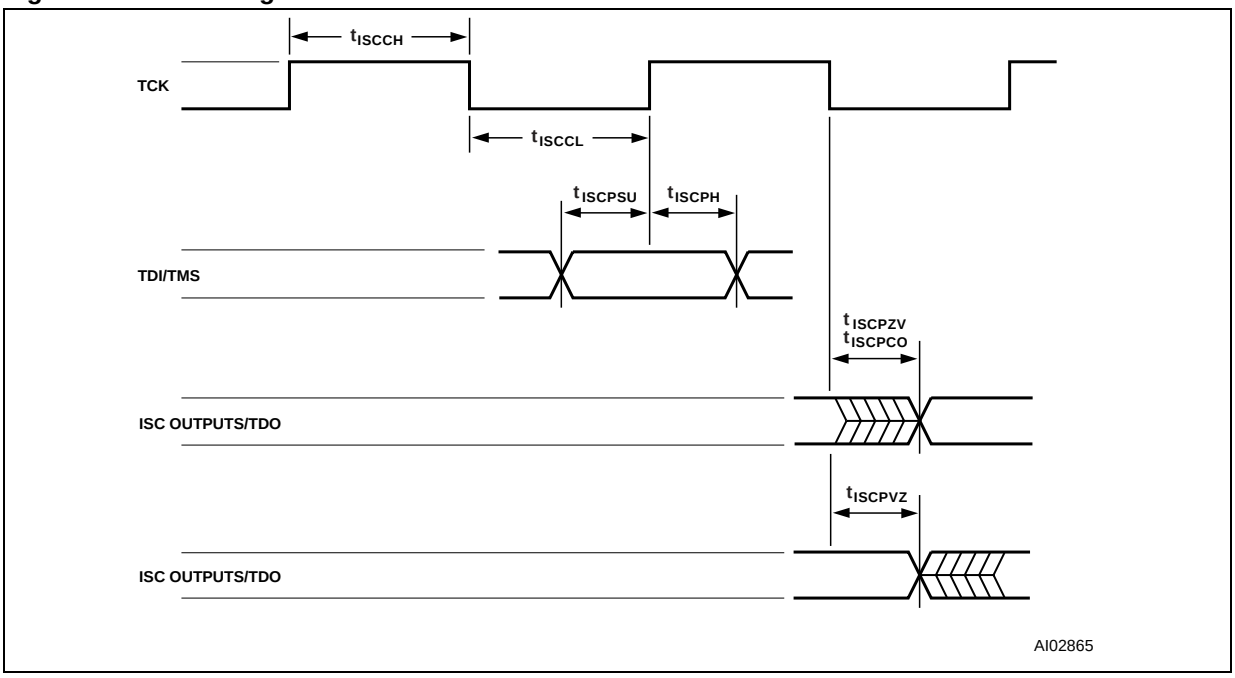


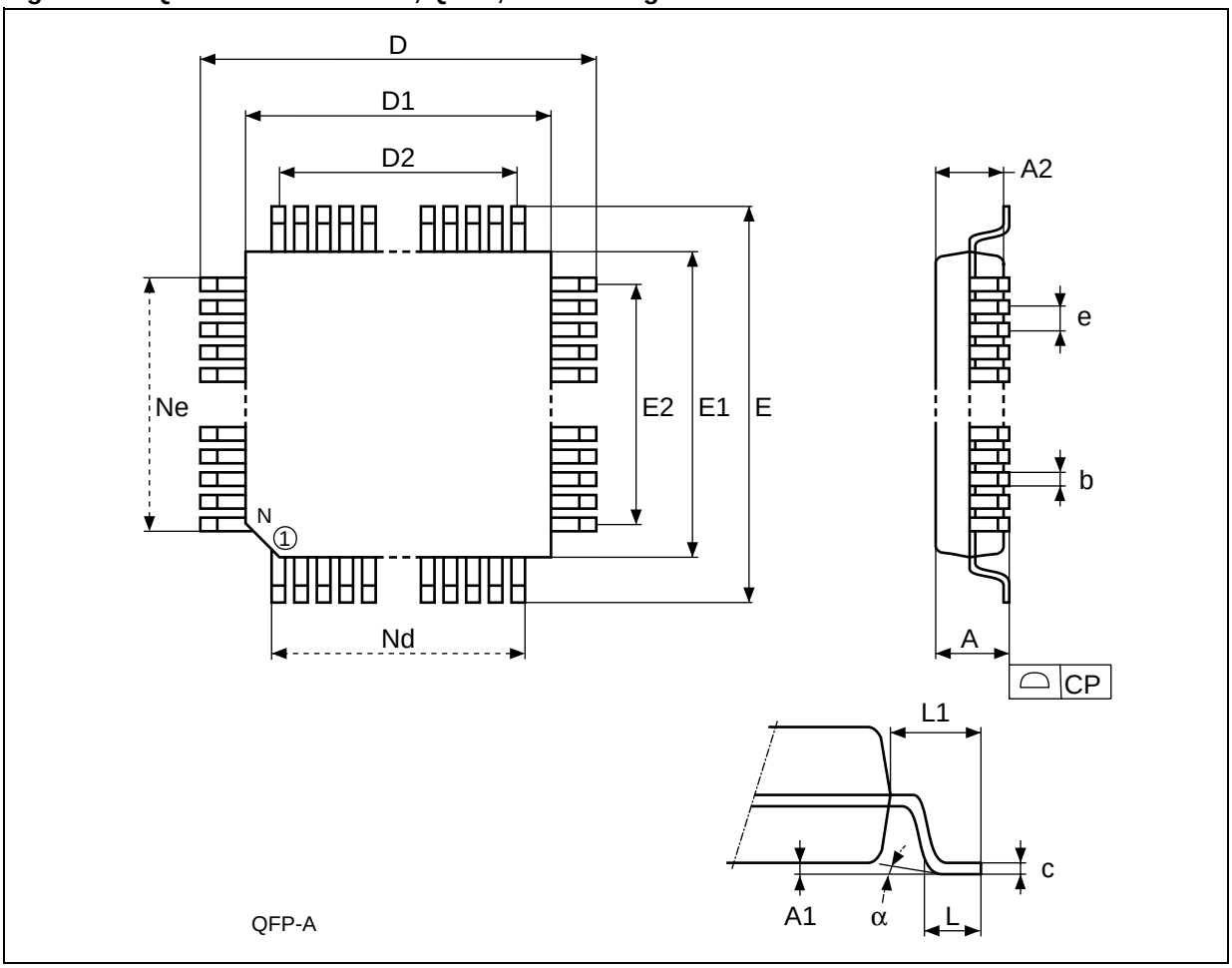
Table 70. ISC Timing

Symbol	Parameter	Conditions	-70		-90		Unit
			Min	Max	Min	Max	
t_{ISCCF}	Clock (TCK, PC1) Frequency (except for PLD)	(Note 1)		20		18	MHz
t_{ISCCH}	Clock (TCK, PC1) High Time (except for PLD)	(Note 1)	23		26		ns
t_{ISCCL}	Clock (TCK, PC1) Low Time (except for PLD)	(Note 1)	23		26		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		6		8		ns
t_{ISCPH}	ISC Port Hold Up Time		5		5		ns
t_{ISCPCO}	ISC Port Clock to Output			21		23	ns
t_{ISCPZV}	ISC Port High-Impedance to Valid Output			21		23	ns
t_{ISCPVZ}	ISC Port Valid Output to High-Impedance			21		23	ns

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

PACKAGE MECHANICAL

Figure 50. TQFP80 - 80 lead Thin, Quad, Flat Package Outline



Note: Drawing is not to scale.

Table 71. TQFP80 - 80 lead Thin, Quad, Flat Package Mechanical Data

Symb.	mm			inches		
	Typ.	Min.	Max.	Typ.	Min.	Max.
A			1.20			0.047
A1		0.05	0.15		0.002	0.006
A2		0.95	1.05		0.037	0.041
b	0.22	0.17	0.27	0.009	0.007	0.011
c		0.09	0.20		0.004	0.008
D	14.00			0.551		
D1	12.00			0.472		
D2	9.50	—	—	0.374	—	—
E	14.00			0.551		
E1	12.00			0.472		
E2	9.50	—	—	0.374	—	—
e	0.50	—	—	0.020	—	—
L	0.60	0.45	0.75	0.024	0.018	0.030
L1	1.00			0.039		
CP	0.08			0.003		
α	3.5°	0.0°	7.0°	3.5°	0.0°	7.0°
N	80			80		
Nd	20			20		
Ne	20			20		

PART NUMBERING

Table 72. Ordering Information Scheme

Example: PSD8 3 5 G 2 – 90 U I T

Device Type

PSD8 = 8-bit PSD with Register Logic

SRAM Size

3 = 64 Kbit

Flash Memory Size

5 = 4 Mbit (512 Kb x8)

I/O Count

G = 52 I/O

2nd Flash Memory

2 = 256 Kbit (32 Kb x8) Flash Memory

Operating Voltage

blank = V_{CC} = 4.5 to 5.5V

$V^{(1)}$ = V_{CC} = 3.0 to 3.6V

Speed

70 = 70ns

90 = 90ns

Package

U = TQFP80

Temperature Range

blank = 0 to 70°C (Commercial)

I = –40 to 85°C (Industrial)

Shipping Option

T = Tape & Reel Packing

Note: 1. The 3.3V ±10% devices are not covered by this datasheet, but by the PSD835G2V datasheet.

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

APPENDIX A. PIN ASSIGNMENTS

Table 73. PSD835G2 TQFP80

Pin No.	Pin Assignments	Pin No.	Pin Assignments	Pin No.	Pin Assignments	Pin No.	Pin Assignments
1	PD2	21	PG0	41	PC0	61	PB0
2	PD3	22	PG1	42	PC1	62	PB1
3	AD0	23	PG2	43	PC2	63	PB2
4	AD1	24	PG3	44	PC3	64	PB3
5	AD2	25	PG4	45	PC4	65	PB4
6	AD3	26	PG5	46	PC5	66	PB5
7	AD4	27	PG6	47	PC6	67	PB6
8	GND	28	PG7	48	PC7	68	PB7
9	V_{CC}	29	V_{CC}	49	GND	69	V_{CC}
10	AD5	30	GND	50	GND	70	GND
11	AD6	31	PF0	51	PA0	71	PE0
12	AD7	32	PF1	52	PA1	72	PE1
13	AD8	33	PF2	53	PA2	73	PE2
14	AD9	34	PF3	54	PA3	74	PE3
15	AD10	35	PF4	55	PA4	75	PE4
16	AD11	36	PF5	56	PA5	76	PE5
17	AD12	37	PF6	57	PA6	77	PE6
18	AD13	38	PF7	58	PA7	78	PE7
19	AD14	39	$\overline{\text{RESET}}$	59	CNTL0	79	PD0
20	AD15	40	CNTL2	60	CNTL1	80	PD1

REVISION HISTORY

Table 74. Document Revision History

Date	Version	Description of Revision
01-Mar-2000	1.0	PSD835G2: Document written in the WSI format. Initial release.
30-Nov-2000	1.1	Turbo Off changed from +10 to +12 in Table 58, CPLD Combinatorial Timing, Table 59, CPLD Macrocell Synchronous Clock Mode Timing, Table 60, CPLD Macrocell Asynchronous Clock Mode Timing, Table 61, Input Macrocell Timing, Table 62, Read Timing, Table 64, Port F Peripheral Data Mode Read Timing. t_{CO} max for 70ns speed class changed from 13 to 15 in Table 59. t_{HA} min and t_{CLA} min for 70ns speed class changed from 5 to 7 and 9 to 12, respectively and t_{CLA} min for 90ns speed class changed from 12 to 15 in Table 60. t_{LXAX} min changed for 70ns speed class changed from 5 to 7 in Table 62. t_{LXAX} min changed for 70ns speed class changed from 5 to 7, t_{DVWH} min for 70ns speed class changed from 12 to 25, t_{WLWH} min for 70ns speed class changed from 25 to 28 in Table 63.
31-Jan-2002	1.2	PSD835G2: Configurable Memory System on a Chip for 8-bit Microcontrollers. Front page and back two pages in ST format added to the PDF file. Any references to Waferscale, WSI, EasyFLASH and PSDsoft 2000 updated to ST, ST, Flash+PSD and PSDsoft.
11-Sep-2002	2.0	Document reformatted. No parameters changed.
03-Mar-04	3.0	Document reformatted; mechanical dimensions corrected (Table 71)