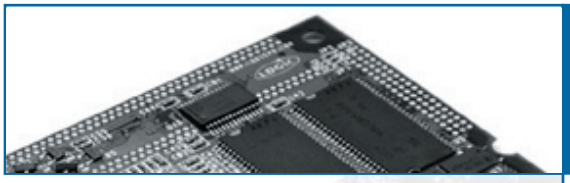
LOGIC PRODUCT DEVELOPMENT WWW.LOGICPD.COM



Zoom™

Card Engine

LH79520-10 Card Engine
IO Controller Specification

REVISION HISTORY

REV	EDITOR	REVISION DESCRIPTION	CPLD REV	APPROVAL	DATE
A	Colette O'Brien	Release	--	C.O.	1/27/03
B	Colette O'Brien	Release	--	C.O.	2/20/03
C	Colette O'Brien	Release	--	C.O.	4/23/03
D	James Wicks, Michael Anderson	SPI Section Edit; GPIO Section Edit; Product Brief Update	--	J.W.	12/31/03
E	Jed Anderson	Added note to reference App Note 303 in Section 3	--	JCA	12/30/05
F	Jed Anderson	Added Appendix A: CPLD Revision History; Added CPLD Rev column in Rev History	2.0B	JCA	2/9/06

Please check www.logicpd.com for the latest revision of this manual, errata's, and additional application notes.

Table of Contents

Table of Figures

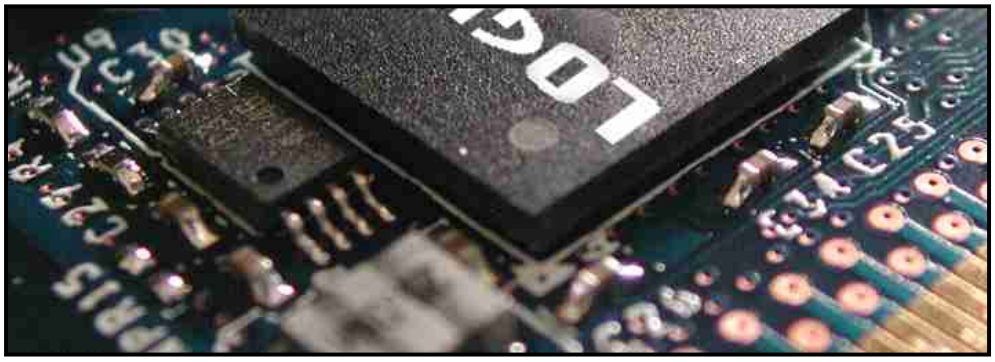
LH79520 I/O CONTROLLER

Developing products is as simple as

A B C

- A Application Development Kits
- B Board Support Packages
- C Card Engines

Logic offers production-ready I/O controller devices and design packages for customers creating custom Card Engine designs and CPLD code for Logic's Card Engines. Logic has optimized the VHDL code to fit in the smallest possible programmable logic device. This results in an embedded product development cycle with **less time, less cost, less risk ... more innovation.**



- **I/O Controller was written in VHDL and contains the following:**
  - I2S (includes DMA)
  - SPI (parallel to SPI interface)
  - ISA-like bus interface
  - SMSC LAN91C111 wired LAN bus interface and power control logic
  - Buffer control logic
  - Chip select decoder logic
  - Interrupt encoder logic
  - Flash program control logic
  - Processor mode control logic
  - IC code revision register
- **Configuration Options**
  - I2S interface to the CODEC can be set to 10 or 12-bit audio
- **Source Code**
  - Includes all VHDL code (licensable .vhd source code files)
- **Support**
  - VHDL IP Core Source Code Design Package includes the Bronze level support package

CUSTOMER SUPPORT

Logic provides technical support for Application Development Kits. Various support packages are available; contact us for more information.

CONTACT

For more information on our Embedded Platform Solutions, please contact Logic Sales at www.logicpd.com or 612.672.9495.

 411 WASHINGTON AVE N SUITE 101 MINNEAPOLIS MN 55401 TEL: 612.672.9495 FAX: 612.672.9489 WWW.LOGICPD.COM

PN: 70000015 Rev. A

1.2 Acronyms

BALE	Buffered Address Latch Enable
CF	CompactFlash
CODEC	Coder Decoder
CS	Chip Select
EEPROM	Electrically Erasable Programmable Read Only Memory
EPROM	Electrically Programmable Read Only Memory
GPIO	General Purpose Input Output
IO	Input Output
IRQ	Interrupt Request
ISA	Industry Standard Architecture
LAN	Local Area Network
LED	Light Emitting Diode
MB	Megabyte (2^20 bytes)
SPI	Serial Peripheral Interface

1.3 Technical Specifications

Please refer to the following component specifications and data sheets.

Altera MAX 7000A PLD data sheet (EPM7256AE)
Altera Device Package Information data sheet
Altera Ordering Information
Texas Instruments TLV320DAC23 data manual
Texas Instruments (Burr-Brown) ADS7843 data sheet

1.4 IO Controller Advantages

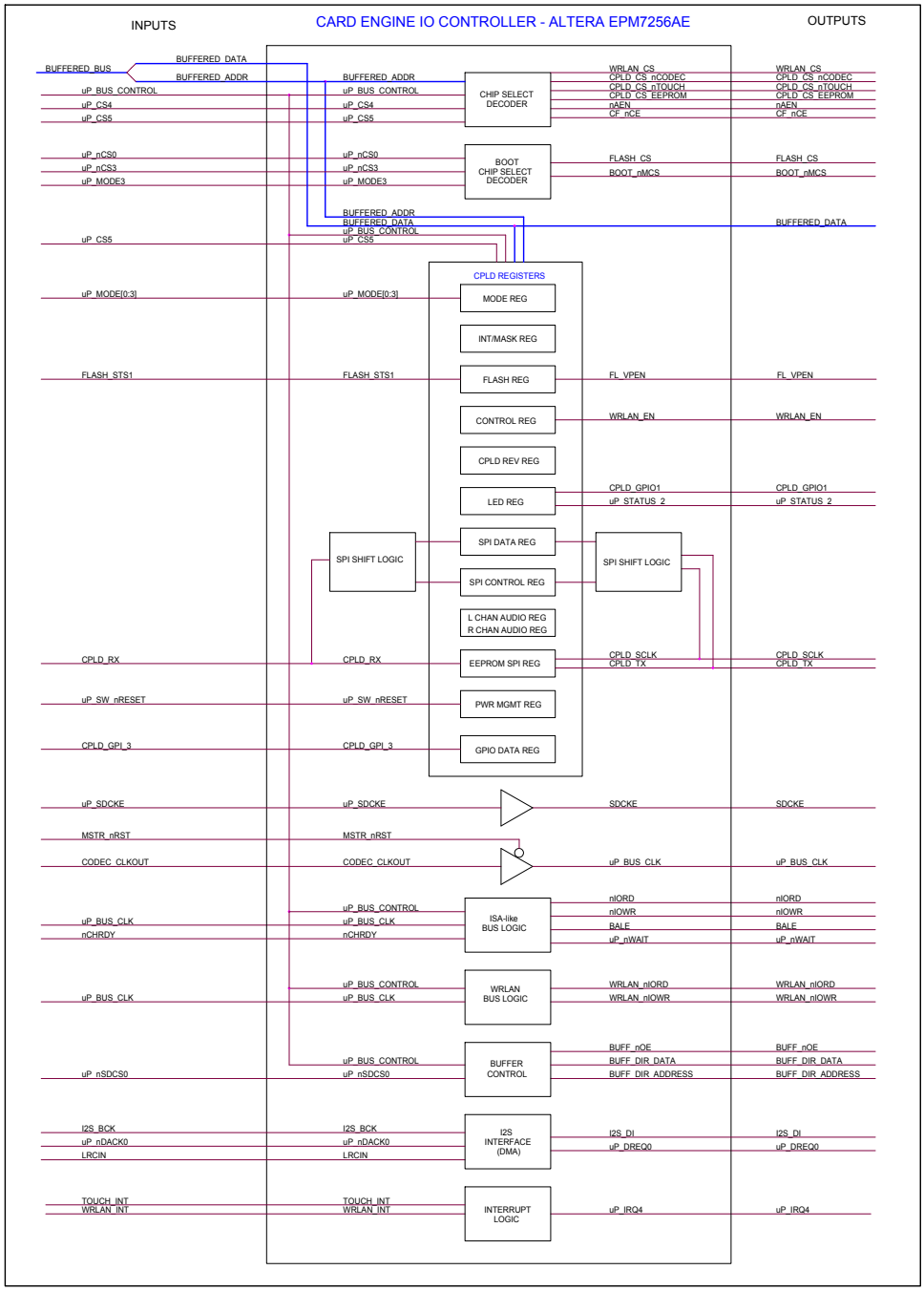
Some of the key features in the IO Controller include:

- Parallel to I2S Audio Interface with DMA
- Parallel to SPI Interface
- Chip Select Decoder
- Interrupt Decoder
- ISA-Like Bus Interface
- Bus Control Logic
- Programmable Register Control
- GPIO Interface
- In-System Programmability via JAM Player

The IO Controller VHDL source code is available for purchase. Contact Logic for more information.

1.5 IO Controller Block Diagram

Figure 1.1: IO Controller Block Diagram



2 IO Controller Address and Register Definitions

Address Range	Memory Block Description	Size
0x5400 0000 – 0x57FF FFFF	Fast Peripherals Chip Select 5 (CS5)	64MB
0x5000 0000 – 0x53FF FFFF	Slow Peripherals Chip Select 4 (CS4)	64MB

2.1 Fast Peripherals Chip Select 5 (CS5)

Address Range	Memory Block Description	Size
0x5400 0000 – 0x541F FFFF	Wired LAN Chip Select	2MB
0x5420 0000 – 0x543F FFFF	Card Engine Control Reg	2MB
0x5440 0000 – 0x545F FFFF	CODEC I2S Audio	2MB
0x5460 0000 – 0x547F FFFF	SPI Data Reg	2MB
0x5480 0000 – 0x549F FFFF	SPI Control Reg	2MB
0x54A0 0000 – 0x54BF FFFF	EEPROM SPI Reg	2MB
0x54C0 0000 – 0x54DF FFFF	Interrupt/Mask Reg	2MB
0x54E0 0000 – 0x54FF FFFF	Mode Reg	2MB
0x5500 0000 – 0x551F FFFF	Flash Reg	2MB
0x5520 0000 – 0x553F FFFF	Power Management Reg	2MB
0x5540 0000 – 0x555F FFFF	IO Controller Code Revision Reg	2MB
0x5560 0000 – 0x557F FFFF	Extended GPIO Reg	2MB
0x5580 0000 – 0x559F FFFF	GPIO Data Reg	2MB
0x55A0 0000 – 0x55FF FFFF	Reserved - On-Board Expansion	2MB (X3)
0x5600 0000 – 0x56FF FFFF	Reserved - Off-Board Expansion	1MB (X16)
0x5700 0000 – 0x57FF FFFF	Open – Available for User	1MB (X16)

Each memory block for chip select 5 is described below. The register definitions include bit descriptions, read/write access allowed, and the initial value upon reset.

2.1.1 Wired LAN Chip Select

Address Range: 0x5400 0000 – 0x541F FFFF

This area of memory is used when accessing the Wired LAN chip (internal registers/memory).

2.1.2 Card Engine Control Register

Address Range: 0x5420 0000

This register holds control bits for the card engine.

7	6	5	4	3	2	1	0	
-	-	-	-	-	-	-	WLPEn	
-	-	-	-	-	-	-	0	reset
-	-	-	-	-	-	-	R/W	R/W

WLPEn: Wired LAN power enable signal. This bit enables/disables power to the on-board Wired LAN chip.

0 = Wired LAN enabled
1 = Wired LAN disabled

2.1.3 CODEC I2S Audio

The I2S core, uses this area of memory. The I2S core is used when interfacing to the on-board CODEC, and supports AUDIO OUT only. See Section 3.7 for more information regarding the CODEC I2S Interface.

Left channel audio data register:

Address Range: 0x54400000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
D15 (MSB) …. D6 (LSB)										-	-	-	-	-	-	
0x00										-	-	-	-	-	-	reset
R/W										-	-	-	-	-	-	R/W

Right channel audio data register:

Address Range: 0x54400002

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
D15 (MSB) …. D6 (LSB)										-	-	-	-	-	-	
0x00										-	-	-	-	-	-	reset
R/W										-	-	-	-	-	-	R/W

2.1.4 SPI Data Register

Address Range: 0x5460 0000

This register holds the SPI data for transmit or receive. SPI is the interface to the touch chip and is the interface used to configure the CODEC. See Section 3.3 for a detailed description of the SPI interface and how to use it.

7	6	5	4	3	2	1	0	
8-bit SPI data								
0	0	0	0	0	0	0	0	reset
R/W								R/W

2.1.5 SPI Control Register

Address Range: 0x5480 0000

This register controls the transmission and reception of SPI data, to/from the 8-bit SPI data register.

7	6	5	4	3	2	1	0	
-	-	SPLD	SPST	SPDN	SPRW	STCS	SCCS	
		0	0	0	0	0	0	reset
-	-	R	R/W	R	R/W	R/W	R/W	R/W

SPLD: SPI load.
0 = SPI data register has not been loaded and shift count has not been reset
1 = SPI data register has been loaded and shift count has been reset
SPST: SPI start.
0 = don't load SPI data register and reset shift count
1 = ready to load SPI data register and reset shift count
SPDN: SPI done.
0 = not done
1 = SPI access done
SPRW: SPI read/write.
0 = SPI write access
1 = SPI read access
STCS: SPI touch chip select.
0 = not selected
1 = touch chip selected
SCCS: SPI CODEC chip select.
0 = not selected
1 = CODEC chip selected

2.1.6 EEPROM SPI Interface Register

Address Range: 0x54A0 0000

This register holds SPI data during a read/write between the processor and on-board EEPROM. The SPI interface timing used for EEPROM communications is implemented by the processor, and not the IO Controller (as in the case of the CODEC and touch chip).

7	6	5	4	3	2	1	0	
-	-	-	-	EECS	EECK	EETX	EERX	
-	-	-	-	0	0	0	0	reset
-	-	-	-	R/W	R/W	R/W	R	R/W

EECS: EEPROM chip select.
0 = not selected
1 = EEPROM chip selected
EECK: EEPROM SPI clock.
EETX: EEPROM SPI data transmit.
EERX: EEPROM SPI data receive.

2.1.7 Interrupt/Mask Register

Address Range: 0x54C0 0000

This register contains the information used by the IO Controller to generate an interrupt to the processor.

7	6	5	4	3	2	1	0	
-	-	-	PIRQ	TMSK	WMSK	TIRQn	WIRQn	
-	-	-	1	0	0	1	1	reset
-	-	-	R/W	R/W	R/W	R	R	R/W

PIRQ: touch chip pen interrupt request (IRQ). The touch chip pen IRQ pin on the IO Controller initially is an input. When the pen IRQ signal goes low (on an interrupt), the IO Controller then drives the pen IRQ low as an output pin. This is the correct interrupt procedure from the component datasheet. When bit(4) of the interrupt/mask register is set to a 1, the pen IRQ pin again becomes an input. Bit(4) will need to be set back to a 0 in order to enable the pen IRQ to be an output again (and allow for another interrupt).

0 = enable pen IRQ to be an output
1 = make pen IRQ an input

TMSK: touch chip interrupt mask.
0 = interrupt not masked
1 = interrupt masked
WMSK: Wired LAN chip interrupt mask.
0 = interrupt not masked
1 = interrupt masked
TIRQn: touch chip interrupt request (IRQ).
0 = interrupt
1 = no interrupt
WIRQn: Wired LAN chip interrupt request (IRQ).
0 = interrupt
1 = no interrupt

2.1.8 Mode Register

Address Range: 0x54E0 0000

This register holds the values of the mode pins.

7	6	5	4	3	2	1	0	
-	-	-	-	MDP3	MDP2	MDP1	MDP0	
-	-	-	-	-	-	-	-	reset
-	-	-	-	R	R	R	R	R/W

MDP3: mode pin 3. Mode pin 3 selects between on-board and off-board boot device. See Section 3.2 for detailed information on mode pin 3. (Note: The only option for the LH79520 processor is a 16-bit bank 0 at boot. The only option for an off-board boot device is a 16-bit device. If setting mode pin 3 low, the user must ensure that the off-board boot device is a 16-bit device.)

0 = off-board boot device (user must ensure a 16-bit device)
1 = on-board boot device (flash)

MDP2: mode pin 2. Mode pin 2 represents the endian setting for the processor. (Note: Regardless of the value read in bit MDP2, the only option for the LH79520 processor is little endian. This cannot be modified for the LH79520 processor.)

0 = big endian
1 = little endian

MDP1, MDP0: mode pin 1 and mode pin2. These mode pins represent the bus width at boot. (Note: Regardless of the value read in bits MDP1 and MDP0, the only option for the LH79520 processor is a 16-bit bank 0. This cannot be modified for the LH79520 processor.)

00 = reserved
01 = 8 bit
10 = 16 bit
11 = 32 bit

2.1.9 Flash Register

Address Range: 0x5500 0000

This register holds status information for the flash.

7	6	5	4	3	2	1	0	
-	-	-	-	FPOPn	-	FST1	FPEN	
-	-	-	-	1	-	1	0	reset
-	-	-	-	R/W	-	R	R/W	R/W

FPOPn: flash populated bit.

0 = flash populated
1 = flash not populated

Note: This bit is used to generate the flash chip select, when mode pin 3 is low. This bit is ignored when mode pin 3 is high. See Section 3.2 for detailed information on mode pin 3.

FST1: flash status pin. This is the flash RY/BY# pin.

0 = flash busy
1 = flash ready

FPEN: flash program enable.

0 = normal flash operations
1 = program flash enabled

2.1.10 Power Management Register

Address: 0x5520 0000

This register holds the value of the uP_SW_nRESET input signal to the CPLD.

7	6	5	4	3	2	1	0	
-	-	-	-	-	-	SWRS	-	
-	-	-	-	-	-	-	-	reset
-	-	-	-	-	-	R	-	R/W

SWRS: value of the uP_SW_nRESET input signal to the CPLD. The uP_SW_nRESET signal has a pull-up resistor on the card engine.

0 = uP_SW_nRESET signal is low

1 = uP_SW_nRESET signal is high

Note: When uP_SW_nRESET is low, an interrupt signal to the processor will be generated. There is no interrupt mask bit for the uP_SW_nRESET signal.

2.1.11 IO Controller Code Revision Register

Address Range: 0x5540 0000

This register holds the IO Controller code revision number.

7	6	5	4	3	2	1	0	
8-bit revision number								
8-bit revision number								reset
R								R/W

2.1.12 Extended GPIO Register

Address Range: 0x5560 0000

This register controls extended general purpose signals.

7	6	5	4	3	2	1	0	
-	-	-	-	-	-	uP_STATUS_2	CPLD_GPIO_1	
-	-	-	-	-	-	1	1	reset
-	-	-	-	-	-	R/W	R/W	R/W

uP_STATUS_2: general purpose output only bit. Typically used to indicate microprocessor status in conjunction with uP_STATUS_1 signal.
0 = Set pin low
1 = Set pin high
CPLD_GPIO_1: general purpose output-only bit.
0 = Set pin low
1 = Set pin high

2.1.13 GPIO Data Register

Address: 0x5580 0000

This register holds data for the CPLD general purpose input pin. The external CPLD signal is labeled MFP23 - CPLD_GPI_3.

7	6	5	4	3	2	1	0	
-	-	-	-	-	-	-	CPLD_GPI_3	
-	-	-	-	-	-	-	-	reset
-	-	-	-	-	-	-	R	R/W

CPLD_GPI_3: external CPLD signal MFP23 - CPLD_GPI_3, general purpose input.
0 = external CPLD signal MFP23 - CPLD_GPI_3 is low
1 = external CPLD signal MFP23 - CPLD_GPI_3 is high

2.1.14 Reserved On-Board Memory Blocks

Address Range: 0x55A0 0000 – 0x55FF FFFF

These memory blocks are reserved for future on-board expansion.

2.1.15 Reserved Off-Board Memory Blocks

Address Range: 0x5600 0000 – 0x56FF FFFF

These memory blocks are reserved for off-board IO controller expansion.

2.1.16 Open Memory Blocks – Available for User

Address Range: 0x5700 0000 – 0x57FF FFFF

These memory blocks are open and available for the user to utilize.

2.2 Slow Peripherals Chip Select 4 (CS4)

Address Range	Memory Block Description	Size
0x5000 0000 – 0x501F FFFF	reserved	2MB
0x5020 0000 – 0x503F FFFF	CF Chip Select	2MB
0x5040 0000 – 0x505F FFFF	ISA-like Bus Chip Select	2MB
0x5060 0000 – 0x51FF FFFF	reserved - On-Board Expansion	2MB (X13)
0x5200 0000 – 0x52FF FFFF	reserved - Off-Board Expansion	1MB (X16)
0x5300 0000 – 0x53FF FFFF	open – Available for User	1MB (X16)

Each memory block for chip select 4 is described below. The register definitions include bit descriptions, read/write access allowed, and the initial value upon reset.

2.2.1 CompactFlash (CF) Chip Select

Address Range: 0x5020 0000 – 0x503F FFFF

This area of memory is used when accessing the off-board memory mapped CompactFlash Type 1 Memory Only slot.

2.2.2 ISA-like Bus Chip Select

Address Range: 0x5040 0000 – 0x505F FFFF

The ISA-like bus is similar to the ISA bus standard, but does not meet every requirement within the standard. This area of memory is used when accessing off-board components on the “ISA-like” bus. See Section 4 for read and write timing diagrams.

2.2.3 Reserved On-Board Memory Blocks

Address Range: 0x5060 0000 – 0x51FF FFFF

These memory blocks are reserved for future on-board expansion.

2.2.4 Reserved Off-Board Memory Blocks

Address Range: 0x5200 0000 – 0x52FF FFFF

These memory blocks are reserved for off-board IO controller expansion.

2.2.5 Open Memory Blocks – Available for User

Address Range: 0x5300 0000 – 0x53FF FFFF

These memory blocks are open and available for the user to utilize.

3 IO Controller Functions

This section describes in detail the different IO Controller function blocks. See Section 1.5 for the IO Controller block diagram.

Note: A specific software protocol must be followed to access IO devices on Sharp Card Engines. Please see Logic's Application Note 303: *Interfacing to IO Devices via the Static Memory Controller on LH7xxxx Card Engines* for examples of the protocol when accessing registers within the CPLD. This document can be found at: https://www.logicpd.com/auth/.

3.1 Chip Select Decoder Logic

This logic decodes processor chip selects 4 and 5 into smaller segments of memory. See Section 2.2 for the fast peripherals chip section and Section 2.2 for the slow peripherals chip section.

3.2 Boot Chip Select Decoder Logic

IMPORTANT NOTE: Mode pin 3 selects between on-board and off-board boot device. (Note: The only option for the LH79520 processor is a 16-bit bank 0 at boot. The only option for an off-board boot device is a 16-bit device. If setting mode pin 3 low, the user must ensure that the off-board boot device is a 16-bit device.)

The card engine can boot from a 16-bit on-board flash or a 16-bit off-board memory device. The boot device is determined by a jumper setting (mode pin 3) on the application board. When mode pin 3 is high, the on-board flash is selected for boot (area 0, CS0), and when mode pin 3 is low, the off-board memory device is selected for boot. This logic implements the following table.

Flash register bit (3) is used to generate the flash chip select, when mode pin 3 is low. This bit is ignored when mode pin 3 is high. See Section 2.1.9 for more information on the flash register.

Flash (on-board)	Off-board memory	Mode Pin 3	Flash Reg (3)	Function
CS0 (area 0)	CS3 (area 3)	1	ignored	boot from flash in bank 0, off-board memory device is in bank 3
CS3 (area 3)	CS0 (area 0)	0	0	boot from off-board memory device in bank 0, flash is in bank 3
CS3 (area 3)	CS0 (area 0)	0	1	boot from off-board memory device in bank 0, (flash not populated) bank 3 is open

The chip selects for area 0 and 3 are routed externally to the flash and off-board memory device by signals FLASH_CS and BOOT_nMCS.

3.3 SPI Interface

3.3.1 CODEC SPI Interface

In order to follow a procedure to configure the CODEC via SPI, see Logic's "CODEC Configuration via SPI" Application note. For complete command set and registers, see the Texas Instruments TLV320DAC23 data manual.

3.4 ISA-like Bus Logic (CompactFlash and ISA peripherals in area 4)

The ISA-like bus is similar to the ISA bus standard, but does not meet every requirement within the standard. This logic outputs the ISA chip select, CompactFlash chip select, BALE, read (nIORD), and write (nIOWR) signals. It also creates two timing delays in the ISA-like bus timing: first, the delay between the falling edge of the chip select (CompactFlash or ISA) and falling edge of read (nIORD) or write (nIOWR) signal, and second, the delay between the rising edge of the read or write signal and rising edge of the chip select.

The first delay is created by shifting the falling edge of the read (nIORD) or write (nIOWR) signal to create a delay from the chip select. The rising edge of the read and write signals follow the rising edge of the processor read and write signals. See Section 4 for sample read and write ISA-like timing diagrams.

The ISA device chip select is created when an access to address range 0x5040 0000 – 0x505F FFFF is made, and the CompactFlash chip select is created when an access to address range 0x5020 0000 – 0x503F FFFF is made.

The ISA-like bus timing is shown with a number of internal wait states programmed for the processor. This is shown in order to meet the CompactFlash timing. The user can verify/change these values by modifying the WST1 and WST2 fields in the processor's Static Memory Controller Bus Configuration Register, for area 4.

The nCHRDY input signal to the CPLD is shown in the ISA-like bus timing diagrams. It can be asserted by CompactFlash or an ISA device. When pulled low, the nCHRDY signal generates a low on the uP_nWAIT signal to the processor, extending the length of the current cycle beyond the programmed internal wait states. The nCHRDY low pulse width for CompactFlash is a maximum of 350ns, and an example of this signal is shown in the timing diagrams. Not all CompactFlash cards or ISA devices will assert the nCHRDY signal. Therefore, the other signals in the read and write timing diagrams are shown assuming the nCHRDY signal was not pulled low.

3.5 Wired LAN Bus Logic

This logic creates read and write output signals to the Wired LAN chip by shifting the falling edge of the read and write signals from the processor, to meet the required Wired LAN timing. The rising edge of the Wired LAN read and write signals are not shifted.

3.6 Buffer Control Logic

This logic controls the output enable and direction of the on-board buffers.

3.7 CODEC I2S Interface

See Logic's "Application Note: Communicating to the CODEC via I2S" in order to communicate to the CODEC via I2S. See the Texas Instruments TLV320DAC23 data sheet for the complete command set and registers.

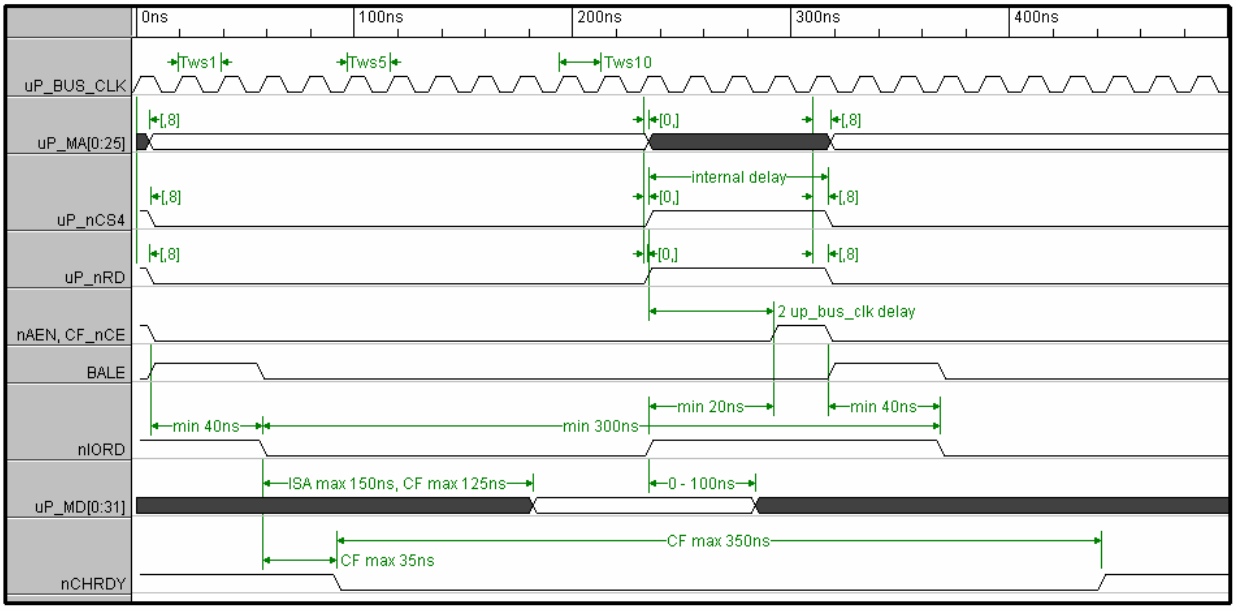
3.8 Interrupt Logic

This logic generates the processor's IRQ4, from information in the Interrupt/Mask register. See Section 2.1.7 for a detailed description of the Interrupt/Mask register.

4 ISA Timing Diagrams

4.1 ISA-like Bus, Read Cycle Timing Diagram

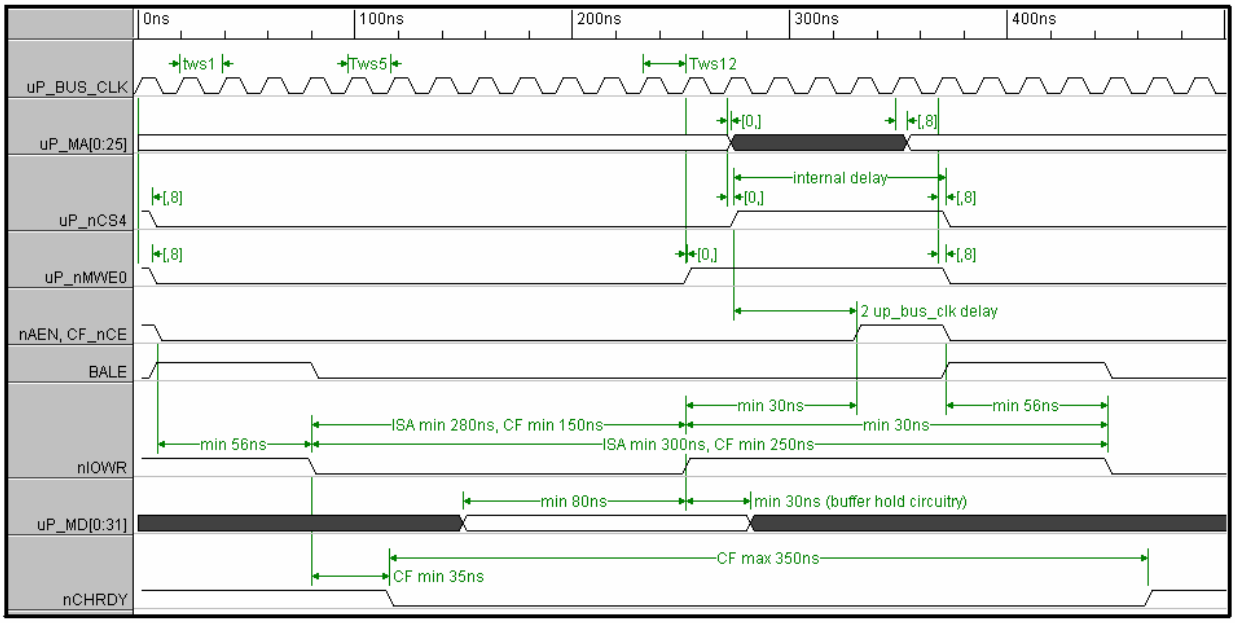
Figure 4.1: ISA-like Bus, Read Cycle Timing



Note: All timing parameters shown in nanoseconds (ns).

4.2 ISA-like Bus, Write Cycle Timing Diagram

Figure 4.2: ISA-like Bus, Write Cycle Timing



Note: All timing parameters shown in nanoseconds (ns).

5 IO Controller Pin Information

Pin	Signal Name	Input/Output
C1	uP_nCS0	Input
F1	uP_nCS3	Input
A2	uP_nCS2	Input
J4	uP_CS3	Input
B3	uP_nCS4	Input
F2	uP_nCS5	Input
A4	WRLAN_CS	Output
B4	CPLD_CS_nCODEC	Output
C4	CPLD_CS_nTOUCH	Output
C5	CPLD_CS_EEPROM	Output
B1	BOOT_nMCS	Output
B2	FLASH_CS	Output
A3	uP_AUX_CLK	Input
F4	CPLD_SCLK	Output
E2	CPLD_TX	Output
E1	CPLD_RX	Input
E3	MFP34 - CODEC_CLKOUT	Input
J4	uP_BUS_CLK	Input
E4	UP_MA25	Input
D2	UP_MA24	Input
D1	UP_MA23	Input
D3	UP_MA22	Input
C2	UP_MA21	Input
F10	UP_MA1	Input
K1	uP_MODE3	Input
J1	uP_MODE2	Input
H1	uP_MODE1	Input
H2	uP_MODE0	Input
G2	TOUCH_BUSY	Input
G1	FL_VPEN	Output
E8	FLASH_STS1	Input
E7	uP_nSDCS0	Input
G3	WRLAN_ENABLE	Output
K5	uP_IRQ4	Output
J5	TOUCH_INT	Input/Output
H5	WRLAN_INT	Input
K4	uP_STATUS_2	Output
J3	uP_SW_nRESET	Input
K3	MFP29 - SDCKE	Output
J2	uP_SDCKE	Input
K2	CPLD_GPIO_1	Output
K6	UP_MD0	Input/Output

J6	UP_MD1	Input/Output
H6	UP_MD2	Input/Output
K7	UP_MD3	Input/Output
J7	UP_MD4	Input/Output
H7	UP_MD5	Input/Output
J8	UP_MD6	Input/Output
K8	UP_MD7	Input/Output
J10	UP_MD8	Input/Output
H10	UP_MD9	Input/Output
H9	UP_MD10	Input/Output
J9	UP_MD11	Input/Output
G9	UP_MD12	Input/Output
G10	UP_MD13	Input/Output
G8	UP_MD14	Input/Output
F9	UP_MD15	Input/Output
D9	uP_DREQ0	Output
D10	uP_nDACK0	Input
H4	uP_DACK1	Input
D8	uP_nWR	Input
B6	uP_nRD	Input
A6	uP_nMWE0	Input
C6	uP_nWAIT	Output
E10	LRCIN	Input
F7	I2S_BCK	Input
E9	I2S_DI	Output
C9	BUFF_nOE	Output
C10	BUFF_DIR_ADDRESS	Output
B10	BUFF_DIR_DATA	Output
B9	WRLAN_nIOWR	Output
A9	WRLAN_nIORD	Output
A8	nIOWR	Output
B8	nIORD	Output
A7	BALE	Output
B7	nCHRDY	Input
C7	nAEN	Output
B5	MSTR_nRST	Input
K9	CF_nCE	Output
K10	RSVD_1	Input
A1	CPLD_TDI	JTAG
F3	CPLD_TMS	JTAG
F8	CPLD_TCK	JTAG
A10	CPLD_TDO	JTAG
D5,G6	VCCINT	POWER
C8,D4,E6,F5,G7,H3	VCCIO	POWER
C3,D7,E5,F6,G4,H8	GNDIO	GND
D6,G5	GNDINT	GND

Appendix A: CPLD Revision History

CPLD revision history from the VHDL source file header.

```
-------------------------------------------------------------------------------------------------------------------
-- Logic Product Development
--
--  Filename:       llh79520_10.vhd
--  Description:    Top-Level Design for the Sharp 9520 Card Engine CPLD
--  Project:        EPS
--  Language:       VHDL
--  Target Part:    Altera CPLD MAX7256AEFC100-10
--  File Type:      Product Code
--  Engineers:      Colette O'Brien (CRO) - Logic Product Development
--                  Matthew Tilstra (MWT) - Logic Product Development
--
--
-- Revision         Date       Engineer  Notes
-------------------------------------------------------------------------------------------------------------------
--  02.0B           10/31/03   MWT       delay DREQ generation by one bitclk in order to fix timing data corruption
--                                              at slower audio playback frequencies.  Also change audio resolution to
--                                              16bit
--  02.0A           06/03/03   CRO       changed WRLAN_CS from inout to out
--  02.09           05/29/03   CRO       changed (CF_nCE and nAEN and WRLAN_CS) to (uP_nCS4 and uP_nCS5)
--                                              for IDK application board peripherals
--  02.08           05/15/03   CRO       added (CF_nCE and nAEN and WRLAN_CS) to first ff of nIOWR, nIORD
--                                              generation
--  02.07           05/15/03   CRO       changed isa, cf, wrlan read/write generation to match 7727 cpld code rev 1.3
--                                              (HW rev 3)
--  02.06           05/14/03   CRO       changed uP_nMWE0 to uP_nWR to generate nIOWR and WRLAN_nIOWR
--                                       changed uP_nMWE0 to uP_nWR in "write registers" process
--                                       changed uP_nMWE0 to uP_nWR in writing spi_data_reg
--  02.05           05/12/03   CRO       reduced cs extend by 1 bus clk for isa and cf
--  02.04           05/07/03   CRO       added bus_clk gets 'Z' when mstr_rst goes high
--  02.03           04/16/03   CRO       documentation changes (comments and notes)
--                                       changed rev code from 00_22 to 02_03
--                                       made all writable registers also readable
--                                       updated "read registers" sensitivity list
--                                       added bit 3 to FLASH_REG
--                                       added CPLD_GPI_3 to gpio data register (input only) in "read registers"
--                                       changed audio back to 10b (as it was for rev B hardware)
--  2.2             04/07/03   CRO       added code to extend slow chip select
--                                       added code to disable BUFF_nOE while extend slow chip select
--                                       added "or uP_nMWE0" and "or uP_nRD" to ISA/WRLAN bus logic
--                                       added ISA BALE signal
--                                       added PWRMGMT_REG (bit 1: sw_reset)
--                                       added code so that sw_reset generates uP_IRQ4 (low)
--                                       changed net names to match schematic net names
--                                       updated chip from MAX7128AFC100-10 to MAX7256AEFC100-10
--  2.1             03/31/03   CRO       chg uP_WR signal to uP_nMWE0 (on global pin)
--                                       chg cpld rev reg from 8b to 16b
--                                       connected input mode pins to MODE_REG
--                                       uP_WAIT code to include "Z" state
--                                       removed SDRAM REG (BUS_CLK provided during RESET)
--  2.0             02/18/03   CRO       Started changes for Rev C card engine hw changes
--                                       1 bus clk (input/output) pin
--                                       update buffer control to SDRAM chip select only
--                                       combine ISA and WRLAN bus control logic
--                                       dropped audio to 8b
--  1.1             02/12/03   CRO       Added fix for bus_clk during sdram refresh
--  1.0             02/05/03   CRO       Code for release
--                                       Removed old commented code
--                                       Fixed UP_IRQ4 (wrlan interrupt was not'd 2 times)
--                                       Made interrupt mask bits read/write
--                                       Connect nCHRDY to uP_WAIT
--                                       Dropped audio from 12b to 10b to gain macrocells
--  0.6             02/01/03   CRO       Added fix for wired lan read and write signals
--  0.5             01/23/03   CRO       Added code to connect EEPROM via SPI
--  0.4             01/14/03   CRO       Made pin changes for Rev B boards
```



```
-- 0.3          01/14/03  CRO    Added sdcke code
--                               Added sw reset interrupt to the interrupt/mask register
--                               Added eeprom spi register
-- 0.2          01/08/03  CRO    Added gate so that INT_REG bits 0 and 1 would read "0" when interrupt and
--                                       "1" when no interrupt.
--                               Added application board LED access
-- 0.1          01/02/03  MWT    Modified design to support 12-bit audio
--                               jumpered uP_MA1 to CPLD_GPIO_2 on EVB
--                               left and right playback registers are now on adjacent 16-bit address boundaries
--                                       instead of sharing an address location.  This should allow the DMAC to
--                                       automatically perform a 32-bit to 16-bit location DMA transfer and get
--                                       the audio data in the right places
-- 0.0          12/23/02  CRO    Added pen IRQ stuff
--           12/23/02  MWT    Modified SPI code to "shift in place" code had two registers, one for uP write
--                                       interface (busclk) and another to shift data out (spiclk) since clocks
--                                       were asynchronous, these registers had to be separate.  By switching
--                                       to a spi clk enable, these clocks are made synchronous and the
--                                       registers can be merged.  So the uP write register is used to shift the
--                                       spi data out
--           12/20/02  MWT    Changed SPI timing generation to use a clock enable instead of a divided
--                                       clock.
--           12/20/02  MWT    Updated uP register read and write interfaces to improve readability
--           12/10/02  CRO    revA (include spi) of document
--           12/04/02  CRO    revA (no spi) of document
--           10/24/02  CRO    rev2a of document
--           10/23/02  CRO    inital rev of document
```

