IM13042 CPLDs

Altera EPM2210F256
Revision 0.2
Sept 20, 2013

1.0 Overview

The IM13042 board uses an Altera MAX II CPLD to allow the main CPU to control the basic functionality of the board. Some of the features controlled by the CPLD are: Resets, Power Sequencing, extended serial bus, I2C and AVR interfaces.

2.0 Internal Registers

The address map of the CPLD internal registers is shown in the table below.
The registers are accessed from the Host CPU thru the LPC interface.
They are also accessible from a test port thru the SPI protocol.

The CPLD responds to the LPC I/O address range 0x0500 - 0x05FF

Table 1. CPLD Registers Address Map

Register Name	Address Offset	Host Access
General Registers		
S_CPLD_SIG_L	0	RO
S_CPLD_SIG_H	1	RO
S_CPLD_REV	2	RO
reserved	3	-
RST_CTL	4	R/W
MISC_CSR	5	R/W
GPIO_R	6	R/W
reserved	7 - 10	-
INT_CFG	11 (0x0B)	R/W
INT_STAT	12 (0x0C)	RO
INT_EN	13 (0x0D)	R/W
LED_CTL	14 (0x0E)	R/W
reserved	15 - 23	-
P_CTL	24 (0x18)	R/W
TEST_R0	25 (0x19)	RO
reserved	26 - 27	-
BA_CTL	28 (0x1C)	R/W
reserved	29 - 30	-
I2C_BSL	31 (0x1F)	R/W
reserved	32 - 39	-
SPI Interface, master		
SPI_CMD	40 (0x28)	R/W
SPI_CFG	41 (0x29)	R/W
SPI_DATA_0 / RM_DATA / B_CNT	42 (0x2A)	R/W
SPI_DATA_1	43 (0x2B)	R/W
SPI_ADDR_0 / RM_ADDR	44 (0x2C)	R/W
SPI_ADDR_1 / RM_SEL	45 (0x2D)	R/W
SPI_ADDR_2	46 (0x2E)	R/W
SPI_ADDR_3 / SPI_MCTL	47 (0x2F)	R/W
MDIO / MDC Interface		
MI_CSR	48 (0x30)	R/W
MI_MCTL	49 (0x31)	R/W

MI_DATA_0	50 (0x32)		R/W
MI_DATA_1	51 (0x33)		R/W
MI_DEVAD	52 (0x34)		R/W
MI_PRTAD	53 (0x35)		R/W
reserved	54 - 55		-
I2C Interface			
I2C_CSR	56 (0x38)		R/W
I2C_DA	57 (0x39)		R/W
I2C_DATA	58 (0x3A)		R/W
I2C_STAT	59 (0x3B)		R/W
I2C_ADDR_0	60 (0x3C)		R/W
I2C_ADDR_1	61 (0x3D)		R/W
I2C_CTL	62 (0x3E)		R/W
reserved	63		-
User Flash Memory Interface			
	SPI Debug Port Offset	**Express Module Offset**	
UFM_CTL	64 (0x40)	128 (0x80)	R/W
UFM_ADDR	65 (0x41)	129 (0x81)	R/W
UFM_DATAIN_L	66 (0x42)	130 (0x82)	R/W
UFM_DATAIN_H	67 (0x43)	131 (0x83)	R/W
UFM_DATAOUT_L	68 (0x44)	132 (0x84)	RO
UFM_DATAOUT_H	69 (0x45)	133 (0x85)	RO
reserved	70 – 127	134 - 256	-

S_CPLD_SIG_L – CPLD Signature Low Byte, read only
Returns the value 0x94.

S_CPLD_SIG_H – CPLD Signature High Byte, read only
Returns the value 0.

S_CPLD_REV – CPLD Revision, read only

7	6 4	3 0
reserved	BRD_REV[2:0]	CPLD_REV[3:0]

bit 7 reserved

bits 6:4 BRD_REV[2:0] – Returns the Board Revision Level

bits 3:0 CPLD_REV[3:0] – Returns the CPLD Revision Level

RST_CTL – Reset Control
All implemented bits are cleared upon Reset.

7	6	5	4	3	2	1	0
res	res	res	res	res	res	PCIE_RST	CPU

bits 7:2 reserved

bit 1 PCIE_RST – PCIE Reset.
When the MASTER_GNT_N signal is asserted and
this bit is set to 1, the PCIE_RST_N signal is asserted.
It must be set to 0 for normal operation.

bit 0 CPU Reset
Writing a 1 into this bit position resets the CPU
This bit is automatically cleared by the hardware.

MISC_CSR – Miscellaneous Control and Status
All implemented bits are cleared upon Reset.

7	6	5	4	3	2	1	0
MD[3]	MD[2]	MD[1]	MD[0]	res	res	LED_C	FAN_MS

bits 7:4 MD[3:0] – Mode Select, read-only
These bits return the levels of the MD[3:0] pins

bits 3:2 reserved

bit 1 LED_C – LED Control
When this bit is set to 1, the LED is controlled by the LED_CTL register.
When it is 0, the LED is controlled by the GBE_LNK and GBE_ACT signals.

bit 0 FAN_MS – Fan Max Speed Control
When set to 1, the fan maximum speed is enabled.
Upon power up, this bit is set to 1.

GPIO_R – GPIO Control
All implemented bits are cleared upon Reset.

7	6	5	4	3	2	1	0
GPO[3]	GPO[2]	GPO[1]	GPO[0]	GPI[3]	GPI[2]	GPI[1]	GPI[0]

bits 7:4 GPO[3:0] – GP Signal Inputs, read-only
These bits return the levels of the GPO[3:0] signals from the Express Module

bits 3:0 GPI[3:0] – GP Signal Outputs
These bits drive the GPI[3:0] signals to the Express Module.
Upon Reset, these bits are cleared.

INT_CFG – Interrupt Configuration

7	6	5	4	3	2	1	0
res	res	res	SIRQ[4]	SIRQ[3]	SIRQ[2]	SIRQ[1]	SIRQ[0]

bits 7:5 reserved

bits 4:0 SIRQ[4:0] – Serial Interrupt Frame Number
The data frames are shown in the table below.

Data Frame #	**Interrupt**	**Clock Past Start Frame**	**Comment**
1	IRQ0	2	Ignored. IRQ0# can only be generated via the internal 8524
2	IRQ1	5	
3	SMI#	8	Causes SMI# if low. Sets the SERIRQ_SMI_STS bit
4	IRQ3	11	
5	IRQ4	14	
6	IRQ5	17	
7	IRQ6	20	

8	IRQ7	23	
9	IRQ8	26	Ignored. IRQ8# can only be generated internally
10	IRQ9	29	
11	IRQ10	32	
12	IRQ11	35	
13	IRQ12	38	
14	IRQ13	41	Ignored. IRQ13# can only be generated from FERR#
15	IRQ14	44	
16	IRQ15	47	
17	IOCHCK#	50	
18	PCI_INTA#	53	
19	PCI_INTB#	56	
20	PCI_INTC#	59	
21	PCI_INTD#	62	

INT_STAT – Interrupt Status

7	6	5	4	3	2	1	0
res	res	res	CARD	FAN_EV	res	res	res

bits 7:5 reserved

bit 4 CARD – Card Interrupt Status, read-only
This bit returns a 1 when the switch board interrupt signal is asserted.

bit 3 FAN_EV – Card Interrupt Status
Returns a 1 when the SMB_ALERT from the fan controller is asserted.

bits 2:0 reserved

INT_EN – Interrupt Enable Register

7	6	5	4	3	2	1	0
TEST	res	res	IE[4]	IE[3]	res	res	res

bits 7 TEST –
Setting this bit to 1 generates an interrupt.
Used to test the serial interrupt.

bits 6:5 reserved

bits 4:3 IE[4:3] – Interrupt Enable bits
When any of these bits is 1 and the corresponding status bit in the INT_STAT register is also 1, a serial CPU interrupt is generated.
The interrupt priority level is determined by the contents of the INT_CFG register.

bits 2:0 reserved

LED_CTL – Status LED Control Register
This register controls the status LED when bit 7 of the MISC_CSR register is set to 1.

7	6	5	4	3	2	1	0

res	res	res	res	COL[1]	COL[0]	LC[1]	LC[0]

bits 7:4 reserved

bit 3:2 COL [1:0] – LED Color
These bits control the Link Speed LED color as shown in the table below.

COL [1:0]	Link Speed LED Color
00	res
01	Yellow
10	res
11	res

bits 1:0 LC[1:0] – LED ON/OFF Control
These bits control the LED state as shown in the table below.

LC[1:0]	LED ON/OFF State
00	OFF
01	ON
10	reserved
11	BLINKING

The following register is used for testing the power regulators.

P_CTL – Power Control
This register controls all the power regulators when the Byte Blaster is connected to the SPI debug port.
If bit 0 is set to 1, the CPLD will perform the power up sequence.
If it is 0, bits 1 to 4 control the enable pins of the individual power regulators.

bit 7 reserved
bit 6 reserved
bit 5 reserved
bit 4 en_1V5
bit 3 en_3V3
bit 2 en_5V0
bit 1 en_12V
bit 0 en_pwr_seq

TEST_R0 – Test Register 0, read-only
This register is used for testing.
It returns the state of the following signals.

bit 7 smb_alrt_N
bit 6 cb_rst_N
bit 5 wdt
bit 4 thrmtrip
bit 3 sus_stat
bit 2 sus_s[3]
bit 1 sus_s[3]
bit 0 sus_s[3]

BA_CTL – Bus Allocation Control
This register controls the association of the Extended Serial Bus to the SPI, MDIO/MDC, I2C modules

and debug SPI port.

7	6	5	4 0
EB_EN	B_MD[1]	B_MD[0]	SA_SL[4:0]

bit 7 EB_EN – Extended Bus Enable
When set to 1, the extended serial bus is enabled and it is controlled by one of the FPGA modules, or by the SPI debug port if bypass mode is selected.
When set to 0, the extended serial bus is disabled and the FPGA modules are used to access peripherals on the master board.

bits 6:5 B_MD[1:0] – Bus Mode
These bits determine which bus (SPI, I2C or MDIO/MDC) is associated with the Extended bus or with the debug SPI port in bypass mode.

SPI_SLN, SPI_FS	EB_EN	B_MD[1:0]	Access Initiator	Target Accessed by Serial Bus
1x	0	xx	Any FPGA Module	All Peripherals on Carrier board
1x	1	00	CPLD SPI Module	Slave Card CPLD Registers
1x	1	01	CPLD SPI Module	Slave Card SPI Bus
1x	1	10	CPLD I2C Module	Slave Card I2C Bus
1x	1	11	CPLD MI Module	Slave Card MDIO/MDC Bus
00	x	xx	Debug SPI Port	CPLD Registers
01	0	00	Debug SPI Port	none
01	0	01	Debug SPI Port	none
01	0	10	Debug SPI Port	I2C Peripherals on Carrier board
01	0	11	Debug SPI Port	none
01	1	xx	Debug SPI Port	Slave Card Peripherals

bits 4:0 SA_SL[4:0] – Slave Agent Select
Select the slave Agent to be accessed via the serial bus

I2C_BSL – I2C Bus Select

7	6	5	4	3	2	1	0
res	res	res	res	res	res	I2C_SL[1]	I2C_SL[0]

bits 7:2 reserved

bits 1:0 I2C_SL[1:0] – I2C Bus select

I2C_SL[1:0]	I2C Bus Selected
00	Fan Controller
01	SMB
01	I2CB
11	reserved

SPI Interface

This module provides all the capabilities to allow the host to access the SPI Peripheral's internal registers through the Serial interface.

The following modes of operation are supported:

Manual Mode
This mode is selected when the RATE[1:0] field in the SPI_CFG register is set to 0.
In this mode the host software generates the output bit stream by bit-banging the MOSI and SCLK bits, and reads the input bit stream through the MISO bit.

Direct Access Mode
This mode is selected when the RATE[1:0] field in the SPI_CFG register is not set to 0.
For one or two-byte accesses, data registers are provided to hold the data.
For multi-byte accesses, two 256-byte (**Note**) buffer memories, TX_RAM and RX_RAM, are used to hold the data to be sent to or received from the serial interface.

Note: In the Niagara 615 the buffer memories are only 4 bytes deep since the CPLD has no memory and therefore these buffers are implemented with registers and multiplexers.

The host software programs the appropriate registers, and then sets the OPCODE field and the ST bit in the SPI_CMD register to start the access operation.
For a write cycle it also loads the data to be written into the data registers or the TX_RAM.
The CPLD handles all the data serialization/de-serialization as required by the SPI interface protocol.
Upon completion it clears the ST bit.

SPI_CMD – SPI Command Register
This register is used to start an SPI read or a write cycle.

7	6	5	4	3	2	1	0
OP[3]	OP[2]	OP[1]	OP[0]	ADDR_W[1]	ADDR_W[0]	res	ST

bits 7:4 OP[3:0] - Operation Code
This field specifies the operation to be performed.
Encodings are given in the table below.
Upon reset, all bits are set to 0.

Command Encodings

OP[3:0]	CPLD Operation
0000	**SPI 8-bit Write** – Write an 8-bit data byte to the SPI interface.
0001	**SPI 16-bit Write** – Write a 16-bit data word to the SPI interface.
0010	**SPI Stream Write** – Write multiple data bytes to the SPI interface. The data to be written to the SPI peripheral is stored into the TX_RAM buffer. The number of bytes is determined by the B_CNT register content.
0011	**SPI Address Write –** Write only the address to the SPI interface. No data is shifted out.
0100	**SPI 8-bit Read** – Read an 8-bit data byte from the SPI interface.
0101	**SPI 16-bit Read** – Read a 16-bit data word from the SPI interface.
0110	**SPI Stream Read** - Read multiple data bytes from the SPI interface. The data read from the SPI peripheral is stored into the RX_RAM buffer. The number of bytes is determined by the B_CNT register content.
0111 to 1110	**reserved**
1111	**RAM Access** – Allows the host software to access the SPI module TX_RAM or RX_RAM buffer. The RAM is accessed by performing a write or read access to / from the SPI_DATA_0 register. After each access the RAM address stored into the SPI_ADDR_0 register is incremented. **Note:** The ST bit in the SPI_CMD register must not be set when this op-code is

	used

bit 3:2 ADDR_W[1:0] – Address Width
Selects the SPI address width
These bits are only written when a SPI_CMD register write is performed and the ST bit is 1.

ADDR_W[1:0]	Address Width
00	1 Byte
01	2 Bytes
10	3 Bytes
11	4 Bytes

bit 1 Reserved

bit 0 ST – Operation Start
Writing a 1 into this bit position starts an SPI access cycle.
Cleared automatically when the access operation is completed.
Writing a zero into this bit position has no effect.

Note: When an operation is started, the host software must wait until the operation is completed (ST bit cleared) before reading any result or before starting another operation.

SPI_CFG – SPI Configuration Register
This register is used to configure the SPI interface.
Upon reset, all implemented bits are set to 0.

7 6	5 3	2	1	0
RATE[1:0]	res	LAB	IB[1]	IB[0]

bits 6:4 RATE[1:0] – Bit Rate Select
These bits select the bit rate of the SPI interface.
When these bits are all 0, manual mode is selected and all the accesses are performed by bit-banging the appropriate bits in the SPI_MCTL register.
RATE encodings are shown in the table below.

RATE[1:0]	Selected Clock Rate
00	Manual Mode
01	(Input Clock) / 100
10	(Input Clock) / 50
11	User Defined

bits 5:3 reserved

bit 2 LAB – Low Address Bits
When this bit is set to 0, all 8 bits of the MS address byte are used.
When this bit is set to 1, only bits 3:0 of the MS address byte are used.

bits 1:0 IB[1:0] – Idle Bytes
Selects the number of idle (don't care) bytes output after the address phase of a Read Command.

IB[1:0]	Number of Idle Bytes
00	0 Idle Bytes
01	1 Idle Bytes
10	2 Idle Bytes
11	3 Idle Bytes

SPI_DATA_0 / RM_DATA / B_CNT – Data Byte 0 or RAM Data or Byte Count
This register holds data bits 7:0 read from, or to be written to the SPI slave internal registers.
Acts as the data window to access the TX_RAM or RX_RAM buffers when the
"RAM Access" code is written into bits 7:4 of the SPI_CMD register.
Holds the number of bytes to be transferred by a SPI Stream Write or Stream Read command.
A Byte Count of 0 causes a 256-byte transfer.
Upon reset, the content of this register is undefined.

SPI_DATA_1 – Data Byte 1
Holds data bits 15:8 read from, or to be written into the SPI slave internal registers
when the data size is 16 bits.

SPI_ADDR_0 / RM_ADDR – SPI Address 0 or RAM Address
This register holds the SPI address/op-code bits 7:0.
It also holds the address of the RAM location accessed by a RAM Write or RAM Read.
Upon reset, the content of this register is undefined.

7	6	5	4	3	2	1	0
A[7]	A[6]	A[5]	A[4]	A[3]	A[2]	A[1]	A[0]

SPI_ADDR_1 / RM_SEL – SPI Address 1 or RAM Select
This register holds SPI address/op-code bits 15:8 when a 16-bit SPI address string is selected.
Bit 0 also selects the RAM to be accessed by a RAM Write or RAM Read command.
0: TX_RAM; 1: RX_RAM
Upon reset, the content of this register is undefined.

7	6	5	4	3	2	1	0
A[15]	A[14]	A[13]	A[12]	A[11]	A[10]	A[9]	A[8]

SPI_ADDR_2 – SPI Address 2
This register holds SPI address/op-code bits 23:16 when a 24-bit SPI address string is selected.
Upon reset, the content of this register is undefined.

7	6	5	4	3	2	1	0
A[23]	A[22]	A[21]	A[20]	A[19]	A[18]	A[17]	A[16]

SPI_ADDR_3 / SPI_MCTL – SPI Address 3 or Manual Control
This register holds SPI address/op-code bits 31:24 when a 32-bit SPI address string is selected.
When Manual Mode is selected (RATE[1:0] bits set to 0), this register is used to control
the SPI interface.
The register formats, are shown below.
Upon reset, the content of the SPI_MCTL register is 0x08.

SPI_ADDR_3 Format

7	6	5	4	3	2	1	0
A[31]	A[30]	A[29]	A[28]	A[27]	A[26]	A[25]	A[24]

SPI_MCTL Format

7	6	5	4	3	2	1	0
res	res	EBD	OBEN	SCS	SCLK	MOSI	MISO

bits 7:6 Reserved

bit 5 EBD – External Buffer Direction Control
This bit is for testing only and it is currently unused

bit 4 OBEN _ Output Buffer Enable
When set to 1, the serial data output buffer is enabled.
This bit is used in Manual Mode only

bit 3 SCS – SPI Interface Select
This bit drives the chip select signal.
This bit is used in Manual Mode only

bit 2 SCLK – SPI Interface Clock
This bit drives the clock signal.
This bit is used in Manual Mode only

bit 1 MOSI – Master Output / Slave Input
This bit drives the serial data output signal
This bit is used in Manual Mode only

bit 0 MISO – Master Input / Slave Output
This bit returns the status of the serial data input signal

MDIO / MDC Interface

This module provides the necessary logic to allow the main CPU to access the internal registers of the Marvel 98X1422 device.
Data serialization and de-serialization can be controlled either by the hardware or by the software through bit-banging.
The following modes of operation are provided:

Manual Mode
This mode is selected when the SCLKS[1:0] field in the MI_MCTL register is set to 0.
In this mode the host software generates the output bit stream by bit-banging the MDIO_O and MDC bits, and it reads the input bit stream through the MDIO_I bit.

Direct Access Mode
This mode is selected when the SCLKS[1:0] field in the MI_MCTL register is not set to 0.
The host software writes the devices addresses into the MI_DEVAD and MI_PRTAD registers.
For a write cycle it also write the data to be written into the MI_AD_0 and MI_AD_1 registers.
It then sets the OP field and the ST bit in the MI_CSR register to start the MDIO operation.
The CPLD generates the serial bit stream (and reads the response, if the access is a read) as required by the MDIO/MDC protocol. Upon completion it clears the ST bit.
For read accesses, the data is returned through the MI_AD_0 and MI_AD_1 registers.

MI_CSR – Management Interface Control and Status (0x20)
Used to start Management Interface accesses.

7	6	5	4	3	2	1	0
res	res	OP[1]	OP[0]	res	res	res	ST

Bits 7:6 reserved

bits 5:4 **OP[1:0]** - Operation Code
This field specifies the operation to be performed.
Encodings are shown in the table below.
The software should not change this field while an operation is in progress (ST = 1).
Upon reset, all bits are set to 0.

OP[1:0]	CPLD Operation
00	Management Interface Address Execute ADDRESS instruction on the (MDIO, MDC) management interface.
01	Management Interface Write Execute WRITE instruction on the management interface.
10	Management Interface Read & Increment Address (1) Execute READ INC instruction on the management interface.
11	Management Interface Read Execute READ instruction on the management interface.

Note 1: The MI_ADDR_0 and MI_ADDR_1 registers are not affected. Only the peripheral device internal address register is incremented.

bits 3:1 reserved

bit 0 ST – Operation Start
Writing a 1 into this bit position starts an CPLD operation as specified by the OP field.
Cleared automatically when the operation is completed.
Writing a zero into this bit position has no effect.

MI_MCTL – Management Interface Manual Control (0x21)
This register is used to access the Management interface.
Upon reset, all implemented bits are set to 0.

7	6	5 4	3	2	1	0
802.3u	res	SCLKSL[1:0]	MDIO_OBE	MDC	MDIO_O	MDIO_I

bit 7 802.3u – Protocol Select

When this bit is set to 1, the protocol is compliant with IEEE 802.3u Clause 22

IEEE Clause 45 Compliant Sequence

	31 30	29 28	27 23	22 18	17 16	15 0
preamble	00	OP[1:0]	PRTAD	DEVAD	10/Z0	DATA/ADDR (16 bits)

IEEE Clause 22 Compliant Sequence

	31 30	29 28	27 23	22 18	17 16	15 0

preamble	01	OP[1:0]	PHYAD	REGAD	10/Z0	DATA (16 bits)

bit 6 Reserved

bits 5:4 SCLKSL[1:0] – Clock Frequency Select
These bits select the clock frequency for the Management Interface.
When these bits are all 0, manual mode for the management interface is selected and all the accesses are handled by the host software thru bits 0 to 3.
SCLKSL encodings are shown in the table below.

SCLKSK[1:0]	Selected Frequency (50 MHz input clock)
00	Manual Mode
01	(Input Clock) / 32
10	(Input Clock) / 16
11	(Input Clock) / 8

bit 3 MDIO_OBE – MDIO Output Buffer Enable
When this bit is set to 1 the MDIO signal output buffer is enabled.
When it is 0, the MDIO output buffer id Tri-Stated.

bit 2 MDC – MDC Output.
When Manual Mode is selected, this bit drives the MDC signal.

bit 1 MDIO_O – MDIO Output.
When Manual Mode is selected, this bit drives the MDIO signal.

bit 0 MDIO_I – MDIO Input, read-only.
Returns the state of the MDIO signal.

MI_AD_0 – Address/Data Register 0 (0x22)
This register holds bits 7:0 of the data read from, or to be written to the device internal register. It contains bits 7:0 of the address during an 'ADDRESS' instruction.
Upon reset, the content of this register is undefined.

MI_AD_1 – Address/Data Register 1 (0x23)
This register holds bits 15:8 of the data read from, or to be written to the device internal register. It contains bits 15:8 of the address during an 'ADDRESS' instruction.
Upon reset, the content of this register is undefined.

MI_DEVAD – Device Address (0x24)
This register holds bits 4:0 of the Device Address
Upon reset, the content of this register is undefined.

MI_PRTAD – Port Address (0x25)
This register holds bits 4:0 of the Port Address
Upon reset, the content of this register is undefined.

I2C Interface

This module contains a set of registers to allow the host software to access any device connected to the I2C bus.

I2C_CSR – I2C Interface Control and Status (0x28)
Used to start I2C Interface accesses.

7	6	5	4	3	2	1	0
res	res	OP[1]	OP[0]	res	res	ERR	ST

bit 7 Reserved

bits 5:4 OP[1:0] - Operation Code
This field specifies the operation to be performed.
Encodings are given in the table below.
The software should not change this field while an operation is in progress (ST = 1).
Upon reset, all bits are set to 0.

OP[1:0]	CPLD Operation
00	**I2C Byte Read** – Reads a data byte from the peripheral device The returned data byte is stored into the I2C_DATA register
01	**I2C Byte Write** - Writes a data byte to the peripheral device The data byte to be written is held in the I2C_DATA register
10	**I2C Acknowledge Polling** Generates a START condition followed by the device address, the read/write bit set to 0 and a STOP condition. The Acknowledge status from the peripheral device is returned thru the NO_ACK bit in the I2C_STAT register.
11	Reserved

bits 3:2 Reserved

bit 1 ERR – Error
This bit returns a 1 when any of the error bits in the I2C_STAT register is set.
Writing a 1 into this bit position stops any I2C interface operation in progress.
Writing a zero into it has no effect.

bit 0 ST – Operation Start
Writing a 1 into this bit position starts an CPLD operation as specified by the OP field.
Cleared automatically when the operation is completed.
Writing a zero into this bit position has no effect.

Note: When an operation is started, the host software must wait until the operation is completed (ST bit cleared) before reading any result or before starting another operation.

I2C_DA – I2C Device Address Register (0x29)

7 1	0
DEV_AD[6:0]	0

bits 7:4 DEV_AD[6:1] – Device address bits

bit 0 reserved

I2C_DATA – Data (0x2A)

Holds the data to be written to or to be read from the I2C peripheral device during a byte write or read operation.

I2C_STAT – I2C Access Status (0x2B)

7 3	2	1	0
res	I2C_FLT	TOUT	NO_ACK

bits 7:3 Reserved

bit 2 I2C_FLT – I2C Fault, read-only
Set to 1 when any one of the I2C bus signals (SCL, SDA) is stuck at 0.
Cleared by the hardware when a new operation is started.

bit 1 TOUT – Time Out, read-only
Set to 1 if the peripheral device forces the SCL line low (to slow down the access) by more than 16 times the bit period.
Cleared by the hardware when a new operation is started.

bit 0 NO_ACK – No Acknowledge, read-only
Set to 1 when no Acknowledge Signaling from the peripheral device is detected.
Cleared by the hardware when a new operation is started.

I2C_ADDR_0 – Address 0 (0x2C)
This register holds the lower 8 bits of the I2C address

I2C_ADDR_1 – Address 1 (0x2D)
This register holds the upper bits of the I2C address.
It is used when a 2-byte address for the I2C bus is specified.

I2C_CTL – I2C Interface Control (0x2E)
This register is used to access the I2C interface.
Upon reset, bits 6, 5, 4 are cleared and bits 3, 2 are set to 1.

7	6	5	4	3	2	1	0
res	A_WIDTH	RATE[1]	RATE[0]	SDA_O	SCL_O	SDA_I	SCL_I

bit 7 Reserved

bit 6 A_WIDTH - Address Width
A value of 0 selects a one-byte address.
A value of 1 selects a 2-byte address.

bits 5:4 RATE[1:0] - I2C Bit Rate
These bits select the bit rate of the I2C interface.
The encodings are shown in the table below.

These encodings will provide bit rates of 50 kbps, 100 kbps and 400 kbps when a 50 MHz input clock is used.

RATE[1:0]	**Selected Bit Rate**
00	(input clock) / 1000
01	(input clock) / 500 (100 kbps at 50 MHz)
10	(input clock) / 100

11	(input clock) / 50

bit 3 SDA_O – SDA Output.
Setting this bit to 0 forces the SDA signal low.
Setting it to 1 allows the SDA signal to be pulled high by the external pull-up resistor.
Upon reset, this bit is set to 1.

bit 2 SCL_O – SCL Output
Setting this bit to 0 forces the SCL signal low.
Setting it to 1 allows the SCL signal to be pulled high by the external pull-up resistor.
Upon reset, this bit is set to 1.

bit 1 SDA_I – SDA Input, read-only.
Returns the level of the SDA signal.

bit 0 SCL_I – SCL Input, read-only
Returns the level of the SCL signal.

User Flash Memory Interface

The CPLD has an internal User Flash Memory (UFM) that can be used to store any information that needs to be retained after being reset or powered off. The UFM consists of 2 sectors, and each sector has 4096 bits of memory, organized in a 256 x 16 bit fashion. Each data word field can be written a word at a time, but it can only be written to once before any erase operation. The erase functionality of the UFM can only erase the entire sector, not a specific word.

UFM_CTL – User Flash Memory Control

7	6	5	4	3	2	1	0
res	res	OP[1]	OP[0]	SEC_SEL	DATA_VLD	OP_STOP	OP_START

bits 7:6 reserved

bits 5:4 **OP[1:0]** - Operation Code
This field specifies the operation to be performed.
Encodings are shown in the table below.
The software should not change this field while an operation is in progress (ST = 1).
Upon reset, all bits are set to 0.

OP[1:0]	**CPLD Operation**
00	UFM Read
01	UFM Write
10	UFM Sector Erase All words in the selected sector will be set to the value of 0xFFFF
11	reserved

.
bit 3 SEC_SEL – Sector Select.
This bit selects either sector 0 or sector 1 of the UFM.

bit 2 DATA_VLD – UFM Data Valid, read-only
This bit is set to 1 when the data to be read is valid after a read operation completes.
This bit is provided for testing and can be ignored.

bit 1 OP-STOP
Writing a 1 into this bit position stops any operation in progress.
This bit is provided to reset the UFM access logic in case the operation does not terminate.
It is not used in normal operation.
This bit is self clearing..

bit 0 OP-START
Writing a 1 into this bit position starts an operation as specified by the OP field.
Cleared automatically when the operation is completed.

Note: When an operation is started, the host software must wait until the operation is completed (OP_START bit cleared) before reading any result or before starting another operation.

UFM_ADDR – User Flash Memory Address
Upon reset, all bits are set to 0.

7	6	5	4	3	2	1	0
UFM_ADDR[7:0]							

bits 7:0 UFM_ADDR – UFM Address.
This register holds the UFM address of the memory location to be accessed in a read or write operation.

UFM_DATAIN_L – User Flash Memory Data-in, Low-Byte
Upon reset, all bits are set to 0.

7	6	5	4	3	2	1	0
UFM_DATAIN_L[7:0]							

bits 7:0 UFM_DATAIN_L – UFM Data-in Low-Byte.
This register holds the lower byte of the data word to be written to the UFM in a write operation.

UFM_DATAIN_H – User Flash Memory Data-in, High-Byte
Upon reset, all bits are set to 0.

7	6	5	4	3	2	1	0
UFM_DATAIN_H[7:0]							

bits 7:0 UFM_DATAIN_H – UFM Data-in, High-Byte.
This register holds the upper byte of the data word to be written to the UFM in a write operation.

UFM_DATAOUT_L – User Flash Memory Data-out, Low-Byte
Upon reset, all bits are set to 0.

7	6	5	4	3	2	1	0
UFM_DATAOUT_L[7:0]							

bits 7:0 UFM_DATAOUT_L – UFM Data-out, Low-Byte.
This register holds the lower byte of the data word returned after a read operation.

UFM_DATAOUT_H – User Flash Memory Data-out, High-Byte
Upon reset, all bits are set to 0.

7	6	5	4	3	2	1	0
UFM_DATAOUT_H[7:0]							

bits 7:0 UFM_DATAOUT_H – UFM Data-out, High-Byte.
This register holds the upper byte of the data word returned after a read operation.

3.0 Timing Diagrams

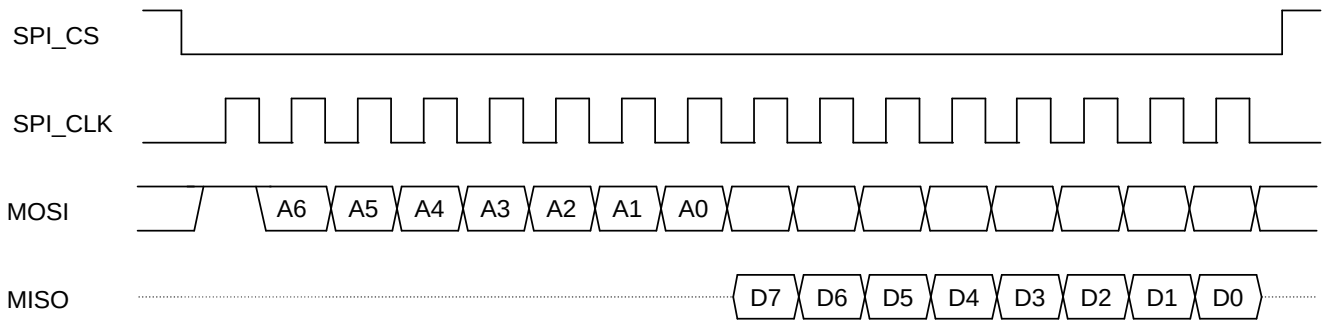


Figure 3-1. Debug SPI Port Read Access, 7-bit Address, 8-bit Data

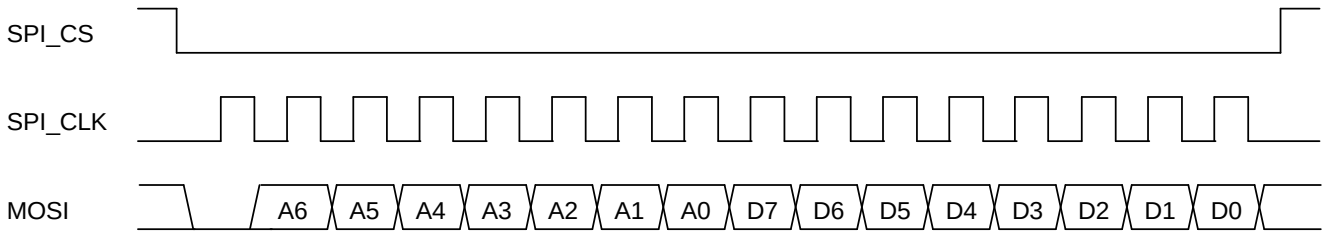


Figure 3-2. Debug SPI Port Write Access, 7-bit Address, 8-bit Data

4.0 History

Document Rev	Changes
0.2	