7I43 MANUAL

V1.6

This page intentionally not blank

Table of Contents

Table of Contents

GENERAL

DESCRIPTION

The 7I43 is a USB/EPP version of the FPGA based Anything I/O card series. It provides 48 I/O bits.

Initial FPGA configurations can be downloaded to the 7I43 the USB or EPP port. The 7I43 has a serial EEPROM for FPGA configuration if ir is not convenient to configure the FPGA by a USB or EPP port at power up.

The 48 I/O bits are available on two 50 pin connectors, 24 bits per connector. The 50 pin connectors have I/O module rack compatible pin-outs. The connector pin-out uses interleaved grounds for lower crosstalk and controlled impedance. Socketed pull-up resistor networks (or optional termination networks) are provided for all I/O bits.

PwrGood, Done, Init and power status LEDs are provided for debugging puposes as are 8 FPGA driven LEDs. Several I/O interface daughter cards are available for the 7I43. These cards include a 4 axis 3A Hbridge, a 2 Axis 3A stepper motor driver, an analog servo amp. interface, an RS-422/485 interface, and a debug LED card. 1 daughter card can plug directly onto the 7I43.

Many IO configuration files are provided with the 7I43 including simple remote I/O, 4 and 8 axis servo motion control, 4 and 8 axis microstepping stepper motor control, multiple channel PWM generator, quadrature counters and more. VHDL source is provided for all configurations.

FPGA system clock is 50MHZ Oscillator. The Spartan3 used can multiply or divide this frequency to suitable values for application use.

The 7I43 uses a 200K or 400K gate Xilinx SpartanIII FPGA. Free development tools for The SpartanIII are available (Xilinx WebPack) from Xilinx’s web site.

HARDWARE CONFIGURATION

GENERAL

Hardware setup jumper positions assume that the 7I43 card is oriented in an upright position, that is, with the USB connector towards the person doing the configuration, and the power connector on top right.

FPGA CONFIGURATION SOURCE

The 7I43's FPGA can be configured via the USB port, The EPP port, or the on card serial EEPROM. Jumpers W4 and W5 select the configuration source.

W4	W5	MODE
DOWN	DOWN	EPP (PARALLEL PORT) CONFIG
DOWN	UP	USB CONFIG
UP	DOWN	EEPROM CONFIG

USB POWER

The 7I43 can be powered by the USB host. The maximum power that can be supplied by a USB host is 450 mA. This will be sufficient for most but not all 7I43 applications. For applications that require more than the 450 mA supplied by the host, the 7I43 has provisions for external power. W6 connects host USB power to the 7I43's power supplies. To use host power, W6 must be set to the "UP" position. If external 5Vpower is used, W6 **must** be set to the "DOWN" position.

WARNING: ***Connecting an external 5V supply to the 7I43 while W6 is in the "UP" position and a USB cable connects the 7I43 to a host computer is likely to damage the computer by feeding external power 'backwards' into the USB port!***

POWER ENABLE

The 7I43 can be set to power-up only after the USB interface is activated. This is the suggested operational mode when the 7I43 is interfaced via USB. For applications where the 7I43 must operate without the USB interface, This function must be disabled. W7 controls the power up enable mode. When W7 is in the "UP" position, the 7I43 power supplies are always enabled. When W7 is in the "DOWN" position, the 7I43 power supplies will only be enabled when the USB interface is active.

HARDWARE CONFIGURATION

CONNECTOR POWER

The power connection on both I/O connectors (Pin 49) can supply either 3.3V or 5V power. Supplied power should be limited to 400 mA total. W1 selects the power supplied to both P3 and P4 . When W1 is in the "UP" position, 5V power is supplied to the connector. When W1 is in the "DOWN" position, 3.3V power is supplied to P3 and P4. Note that most Mesa I/O adapter cards that connect to Anything I/O cards require 5V.

BUS SWITCH MODE

The 7I43 uses bus switch devices in series with all user I/O pins. These devices allow the 7I43 inputs to be 5V tolerant and allow the I/O pins to be pulled up to 5V. The bus switch input protection function works by disconnecting the FPGA from the IO pins when the IO pin voltage rises above a preset threshold. This threshold determines the bus switch operational mode and is selectable on a per connector basis. We refer to the modes as 5V mode and 3.3V mode.

When in 5V mode, the inputs and tri-stated outputs may be pulled up to 5V. This allows driving 5V referred loads such as I/O module racks. The disadvantage of 5V mode is that the output impedance is higher in the high output state (when the FPGA pins are at 3.3V) as the bus switch is off when the FPGA pin is at 3.3V.

When 3.3V mode is selected, the bus switch is always fully on unless input voltages >4V are applied, at which point the bus switch disconnects the FPGA from the I/O pin. 3.3V mode is suggested for general use.

Jumper W2 determines bus switch mode for all user I/O pins. When jumper W2 is in the "UP" position, 5V mode is selected, when ‘down’, 3.3V bus switch mode is selected.

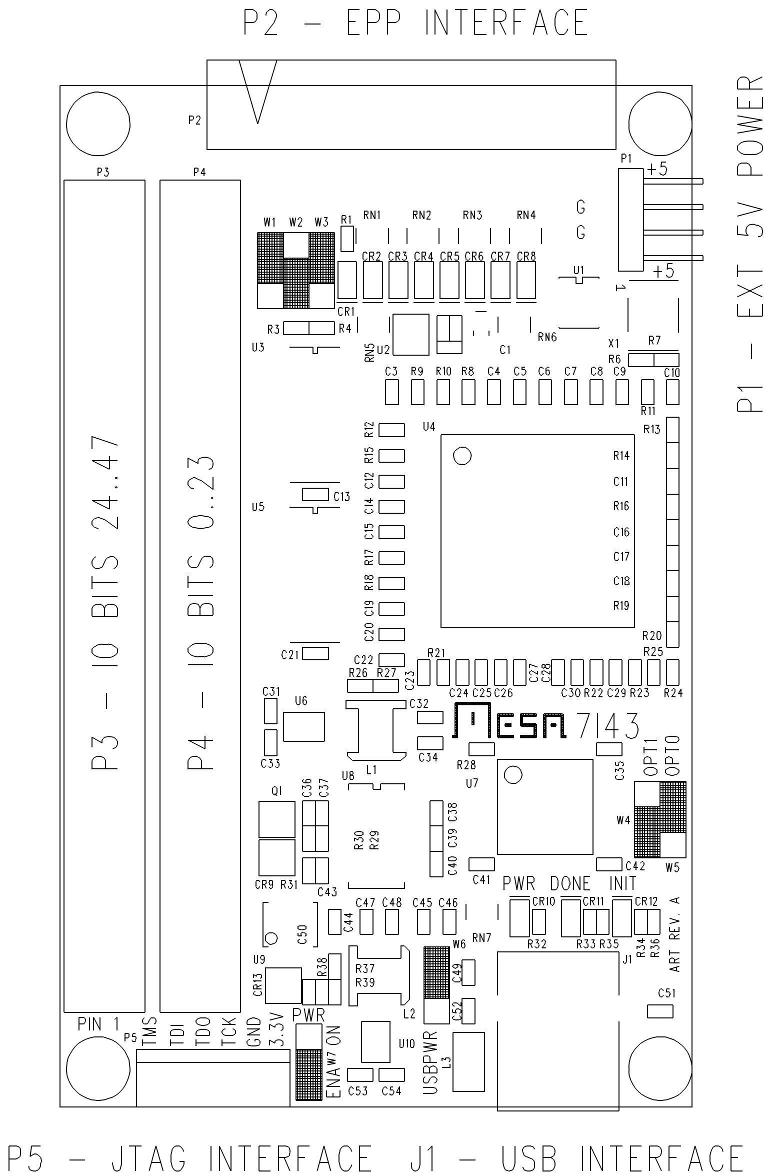
PRE-CONFIGURATION PULL-UPS

The 7I43 has no pull-up resistors on its user I/O pins. This means that before these pins are configured, they will not have a defined state. If this is not desired, internal pull-up resistors on all FPGA pins can be enabled via Jumper W3. When W3 is in the "DOWN" position, user I/O will float until the FPGA is configured. When W3 is in the "UP" position, all FPGA pins including user I/O pins will have a pull-up resistor to 3.3V so the pins will be in a "HIGH" state. It is suggested that the internal pull-ups be enabled unless this causes a problem with connected I/O devices. Note that once the FPGA is configured, each FPGA input pin can have programmable pull-up or pull-down resistors.

CONNECTORS

CONNECTOR LOCATIONS AND DEFAULT JUMPER POSITIONS

(Standard model shown -P version has different defaults)



CONNECTORS

I/O CONNECTORS

P3 and P4 are the 7I43s I/O connectors. These are 50 pin box headers that mate with standard 50 conductor female IDC connectors. For information on which I/O pin connects to which FPGA pin, please see the 7I43IO.PIN file on the 7I43 distribution disk. 7I43 IO connector pinouts are as follows:

P4 CONNECTOR PINOUT

PIN	FUNC	PIN	FUNC	PIN	FUNC	PIN	FUNC
1	IO0	2	GND	3	IO1	4	GND
5	IO2	6	GND	7	IO3	8	GND
9	IO4	10	GND	11	IO5	12	GND
13	IO6	14	GND	15	IO7	16	GND
17	IO8	18	GND	19	IO9	20	GND
21	IO10	22	GND	23	IO11	24	GND
25	IO12	26	GND	27	IO13	28	GND
29	IO14	30	GND	31	IO15	32	GND
33	IO16	34	GND	35	IO17	36	GND
37	IO18	38	GND	39	IO19	40	GND
41	IO20	42	GND	43	IO21	44	GND
45	IO22	46	GND	47	IO23	48	GND
49	POWER	50	GND				

CONNECTORS

7I43 I/O CONNECTORS

P3 CONNECTOR PINOUT

PIN	FUNC	PIN	FUNC	PIN	FUNC	PIN	FUNC
1	IO24	2	GND	3	IO25	4	GND
5	IO26	6	GND	7	IO27	8	GND
9	IO28	10	GND	11	IO29	12	GND
13	IO30	14	GND	15	IO31	16	GND
17	IO32	18	GND	19	IO33	20	GND
21	IO34	22	GND	23	IO35	24	GND
25	IO36	26	GND	27	IO37	28	GND
29	IO38	30	GND	31	IO39	32	GND
33	IO40	34	GND	35	IO41	36	GND
37	IO42	38	GND	39	IO43	40	GND
41	IO44	42	GND	43	IO45	44	GND
45	IO46	46	GND	47	IO47	48	GND
49	POWER	50	GND				

CONNECTORS

7I43 JTAG CONNECTOR

P5 is a JTAG programming connector. It is not normally used since the 7I43 can be programmed via the USB or EPP interface, but can be useful when debugging or reprogramming the CPLD. 3.3V levels are used for JTAG signals. A single JTAG connector is used for both the CPLD and the FPGA, with the CPLD being first in the JTAG chain.

P6 CONNECTOR PINOUT

PIN	FUNCTION	DIRECTION
1	TMS	IN
2	TDI	IN
3	TDO	OUT
4	TCK	IN
5	GND	
6	+3.3v	

POWER CONNECTOR

The 7I43 has an external 5V power connector, P1. This connector supplies power to the 7I43 in EPP, standalone, and USB applications where USB host power is not sufficient to power the 7I43. P1 is a four pin .1" male header. P1 pin-out is as follows:

P1 CONNECTOR PINOUT

PIN	FUNCTION
1	+5V
2	GND
3	GND
4	+5V

CONNECTORS

EPP INTERFACE CONNECTOR

P2 is the EPP printer port interface connector. P2 is a 26 pin header. P2 pin-out matches stands DB25 printer port pin-out, allowing a simple flat cable with a DB25M IDC connector on one end and a 26 pin female header on the other end to interface the hosts printer port to the 7I43.

P2 PIN	**DB25 PIN**	**SIGNAL**	**P2 PIN**	**DB25 PIN**	**SIGNAL**
1	1	/STROBE	2	14	/AUTOFD
3	2	PD0	4	15	/FAULT
5	3	PD1	6	16	/INIT
7	4	PD2	8	17	/SELECTIN
9	5	PD3	10	18	GND
11	6	PD4	12	19	GND
13	7	PD5	14	20	GND
15	8	PD6	16	21	GND
17	9	PD7	18	22	GND
19	10	/ACK	20	23	GND
21	11	BUSY	22	24	GND
23	12	PERROR	24	25	GND
25	13	SELECT	26	VCC	

Note: All handshake signals are available at the CPLD but only /STROBE, /AUTOFD,/SELECTIN and BUSY are forwarded to the FPGA with the standard CPLD configuration.

OPERATION

FPGA

The 7I43 uses a Xilinx Spartan-III FPGA in a 144 pin QFP package, Either PN XC3S200-5PQ144C or XC3S400-5PQ144C depending on 7I43 model.

HOST INTERFACE

The 7I43 uses either a USB or EPP printer port interface to the host. These interfaces can be used for programming the FPGA and accessing the FPGA once programmed.

EPP CONFIGURATION

When the7I43 is jumpered so the configuration source is EPP, *and* the FPGA is not configured (DONE is low), the on card CPLD implements two EPP registers to allow configuring the FPGA via the EPP port.

The two EPP registers are the control register and the data register. The control register is at EPP address 1 and has a single output bit (at D0) that controls FPGA /PROGRAM, and a single input bit (at D0) that reads the FPGA's done status. The data register at EPP address 0, is used for the byte wide configuration data. Reads from the data register will return the FPGA size in D0, 1 = 400K and 0 = 200K.

EPP CONFIGURATION PROCEDURE

EPPWriteAddress(1)	; Select EPP address 0x01 = control register
EPPWriteData(0)	; Set /PROGRAM low
EPPWriteData(1)	; Set /PROGRAM High
(Wait 100 Usec)	; Wait 100 Usec for FPGA to initialize
EPPWriteAddress(0)	; Select EPP address 0x00 = data register
EPPReadData	; Verify FPGA size
EPPWriteData(FPGAByte0)	; Write first byte of FPGA config data
EPPWriteData(FPGAByte1)	; Write second byte of FPGA config data

(write remaining FPGA config bytes)

OPERATION

EPP CONFIGURATION

Once the FPGA is configured, the CPLD EPP registers and EPP handshake logic are disabled and it is the FPGA's responsibility to handle the EPP host interface. The CPLD is still used at this point to forward some of the EPP port handshaking lines through to the FPGA for 5V tolerance.

USB CONFIGURATION

When the7I43 is jumpered so the configuration source is USB, and the FPGA is not configured (DONE is low), the on card CPLD implements a simple data handshake so all data sent to the USB port is written to the FPGAs configuration port.

The CPLD also will echo characters to indicate the FPGA size and DONE status. If a character is sent to the 7I43 and the characters LSb is ‘0' the DONE status will be returned in the echoed characters LSb. If a character with a ‘1' LSb is sent, a character will be echoed indicating the FPGA size. This echoed character will have a ‘0' LSb for 200K 7I43s and a ‘1' LSb for 400K 7I43s. Since it in not desirable to deal with echoed characters for every configuration byte sent to the 7I43, status character echoing is disabled after receiving 4 consecutive characters with a ‘0' LSB..

Once the FPGA is configured the CPLD data handshake is disabled and it is the FPGA's responsibility to handle the interface to the USB chip.

USB CONFIGURATION PROCEDURE

Flush receive buffer	; Optional
Send "0" character	; Optional
Check echoed character for LSb = 0 (done should be low	; Optional
Send "1" character	; Optional
Check echoed character LSb to determine FPGA size	; Optional
Send 4 "0" characters	; Disable echo
Send configuration byte 0	
Send configuration byte 1	
(Send remaining configuration bytes)	

OPERATION

EEPROM CONFIGURATION

For stand-alone applications and when it is not desired to have to preconfigure the FPGA via the host interface at power up, the 7I43 can be configured via its serial EEPROM. Of course the Serial EEPROM must first be programmed with the desired configuration file. The serial EEPROM used is a ST M25P20 SPI flash serial EEPROM.

All access the serial EEPROM is via the FPGA, so programming the serial EEPROM is a "bootstrap" process, where the first step is programming the FPGA with a configuration giving host (EPP or USB) access to the serial EEPROM through the FPGA. Both the EPP and USB GPIO demo configurations allow this EEPROM access via a simple SPI interface built into the configuration.

The SCM7I43P program is an example program for writing the serial EEPROM via the EPP port (DOS only), SCM7I43U is a similar example program for writing the serial EEPROM via the USB port (windows only) The SCM programs rely on EPPIOPR8 (for EPP programming or USBIOPR8 (for USB) configuration file being preloaded into the FPGA before writing the serial EEPROM, as the serial EEPROM can only be accessed through the FPGA. EPPIOPR8 and USBIOPR8 have a simple SPI interface to allow EEPROM access.

EXTRA EEPROM SPACE

The serial configuration EEPROM on the 7I43 has a capacity of 256K bytes, but the configuration bit file for the 400K Spartan 3 chip is only ~208K bytes, leaving 48 K bytes free for FPGA accessible non volatile storage. The 200K gate Spartan 3 chip uses only ~128K bytes of the serial EEPROM leaving 128K bytes free. This storage can be used for non-volatile settings or program storage in stand-alone 7I43 applications.

OPERATION

RECONFIGURATION

Once the 7I43 is configured, the CPLD loader is disabled. In order to reconfigure the FPGA, the FPGA must be reset via /PROGRAM. This can be done by having the FPGA assert its /RECONFIG pin (drive it low). If you wish to have the ability to reconfigure the FPGA without cycling the power, the FPGA configuration must include some way of asserting /RECONFIG. /RECONFIG is FPGA pin 52.

CONFIGURATION FILE STARTUP OPTIONS

Important: *Because the 7I43s CPLD stops configuration when DONE is asserted, the configuration file startup options must be set so that asserting DONE is the last configuration step.* Suggested startup options are as follows:

FPGA STARTUP CLOCK:	**CCLK**
DONE:	**6**
ENABLE OUTPUTS:	**5**
RELEASE WRITE ENABLE:	**4**
RELEASE DLL:	**NO WAIT**

SC7I43P and SC7I43U

Two utility programs, SC7I43P.EXE and SC7I43U are provided to send configuration files to the 7I43. SC7I43P is a DOS only program and SC7I43U is a windows only program.

SC7I43P is invoked with the FPGA configuration file and the Hexadecimal EPP port base address on the command line:

```
SC7I43P FPGAFILE.BIN 378
```

SC7I43U is invoked with the FPGA configuration file and the COM port on the command line:

```
SC7I43U FPGAFILE.BIN COM6
```

SC7I43P and SC7I43U use binary FPGA configuration files. These files can standard Xilinx BIT files or Xilinx PROM format files.

OPERATION

CLOCK SIGNALS

The 7I43 has a 50 MHz crystal controlled clock signal routed to pin 53 (GCLK3) on the FPGA. Four user I/O pins are also GCLK pins:

IO BIT	GCLK	FPGA PIN	IO BIT	GCLK	FPGA PIN
IOBIT0	GCLK6	127	IOBIT24	GCLK7	128
IOBIT17	GCLK4	124	IOBIT40	GCLK5	125

EPP-FPGA INTERFACE

The interface from host EPP printer port to the FPGA uses 12 FPGA pins. These consist of an eight bit bidirectional data bus (D0..D7), and four handshake lines. Note that the handshake lines are fed through the CPLD so depend on the standard CPLD configuration. The D bus connects to the FPGA through 100 Ohm resistors. These resistors provide 5V tolerance and series line termination for driving the cable.

P2 PIN	EPPNAME	SPPNAME	FPGA PIN	DIRECTION
1	/WRITE	/STROBE	84	TO FPGA
2	/DSTROBE	/AUTOFD	79	TO FPGA
8	/ASTROBE	/SELECTIN	80	TO FPGA
21	WAIT	BUSY	82	FROM FPGA
3	D0	D0	68	BIDIR
5	D1	D1	63	BIDIR
7	D2	D2	60	BIDIR
9	D3	D3	59	BIDIR
11	D4	D4	51	BIDIR
13	D5	D5	50	BIDIR
15	D6	D6	47	BIDIR
17	D7	D7	46	BIDIR

OPERATION

EPP-FPGA INTERFACE

The EPP interface implements a simple multiplexed 8 bit data/address bus. The EPPIOPR8 configuration can be used as an example of an EPP interface in the FPGA. This is a simple GPIO interface organized as six eight bit ports.

USB-FPGA INTERFACE

The USB interface uses a FTDI USB interface chip, the FT245R. This is a Full speed interface chip (12 Mbps max). The FT245R appears as a single endpoint communication device, basically a simple bidirectional byte-stream with receive and transmit FIFOs. In order to use the USB configuration and interface you must load the appropriate drivers for you operating system. These drivers are available at FTDICHIP.COM. The utilities supplied with the 7I43 utilize the VCP (Virtual COM Port) series drivers.

The FPGA interface uses a bidirectional 8 bit data bus that is shared with the EPP interface. Because of this sharing you cannot operate the USB and EPP interfaces simultaneously.

SIGNAL NAME	FPGA PIN	DIRECTION	FUNCTION
USBWRITE	56	TO FPGA	XMIT DATA STROBE
/USBRD	55	TO FPGA	RECV DATA STROBE
/USBTXE	83	TO FPGA	XMIT FIFO NOT FULL
/USBRXF	69	FROM FPGA	RECV FIFO HAS DATA
D0	68	BIDIR	DATA BUS
D1	63	BIDIR	DATA BUS
D2	60	BIDIR	DATA BUS
D3	59	BIDIR	DATA BUS
D4	51	BIDIR	DATA BUS
D5	50	BIDIR	DATA BUS
D6	47	BIDIR	DATA BUS
D7	46	BIDIR	DATA BUS

OPERATION

LEDS

The 7I43 has 8 FPGA driven user LEDS. These green LEDS are located in the top center of the card. They can be used for any purpose, and can be helpful as a simple debugging feature. A low output signal from the FPGA lights the LED. See the 7I43MISC.PIN file for FPGA pin locations of the LED signals.

In addition to the user LEDs there are three other LEDS that display board status information. These status LEDS are on the lower right hand side of the card just above the USB connector, The LEDS are a yellow PWR LED, a red DONE LED and a red INIT LED. The DONE and INIT LED can be used to determine FPGA configuration status. The INIT LED will be illuminated when /PROGRAM is asserted or when a CRC error has occurred during the configuration process. The DONE LED will be illuminated when the FPGA is not configured.

BUS SWITCH MODE

The 7I43 uses bus switch devices in series with all I/O pins. These devices allow the 7I43 inputs to be 5V tolerant and allow the I/O pins to be pulled up to 5V. The bus switch input protection function works by disconnecting the FPGA from the IO pins when the IO pin voltage rises above a preset threshold. This threshold determines the bus switch operational mode and is selectable on a per connector basis. We refer to the modes as 5V mode and 3.3V mode.

When in 5V mode, the inputs and tri-stated outputs may be pulled up to 5V. This allows driving 5V referred loads such as I/O module racks. The disadvantage of 5V mode is that the output impedance is higher in the high output state (when the FPGA pins are at 3.3V) as the bus switch is off when the FPGA pin is at 3.3V.

When 3.3V mode is selected, the bus switch is always fully on unless input voltages >4V are applied, at which point the bus switch disconnects the FPGA from the I/O pin. 3.3V mode is suggested for general use.

When the bus switch mode jumper W2 is in the ‘up’ position, 5V mode is selected, when ‘down’, 3.3V bus switch mode is selected.

OPERATION

IO LEVELS

The FPGA used on the 7I43 is a Spartan3. The Spartan3 supports many I/O standards. The 7I43 does not support use of the I/O standards that require input reference voltages, also VCCIO is fixed at 3.3V. The available I/O options for are LVTTL, LVCMOS_33, LVDCI_33, and LVDCI_33_DV2 .

The Spartan3 FPGA chip used on the 7I43 is not 5V tolerant but external bus switch parts are used on the 7I43 to make the I/O pins 5V tolerant. The bus switch parts disconnect the FPGA pins from the I/O pins when the I/O pins are driven to positive voltage levels that would damage the FPGA.

The voltage level that causes disconnect can be selected to be ~4V (3.3V mode) or ~3.3V (5Vmode). For most applications, the 3.3V mode should be used. The 5V mode is useful when driving 5V referred loads.

Note that there is no protection against negative input voltages other than the input clamp diodes in the FPGA and bus switches, so negative input voltages must be limited to -.5V

TERMINATION

The FPGA used on the 7I43 supports series and parallel termination that can be programmed on a pin-for-pin basis. This feature is called DCI. The 7I43 supports DCI on all user I/O pins. The DCI reference resistors are all 100 Ohm 1%.

SUPPLIED CONFIGURATIONS

EPPIOPR8

GENERAL

The EPPIOPR8 configuration is a simple six port GPIO configuration with EPP interface. The GPIO is organized as six eight bit ports, each with an associated Data Direction Register (DDR).The EPPIOPR8 configuration can be used as a starting point for more complicated user configurations.

PORT	DATA REG	DDR	IO BITS	CONNECTOR
0	0x10	0x20	0..7	P4
1	0x11	0x21	8..15	P4
2	0x12	0x22	16..23	P4
3	0x13	0x23	24..31	P3
4	0x12	0x24	32..39	P3
5	0x15	0x25	40..47	P3

In addition to the GPIO bits, the EPPIOPR8 configuration has a simple SPI interface to the configuration EEPROM and a reconfiguration port.. The SPI port allows the utility program SCM7I43P to write configuration data to the serial EEPROM. These registers are mapped as follows:

REGISTER	ADDRESS	FUNCTION
SPICS	0x7D	Single I/O bit to control SPI Chip Select (bit 0)
SPIDATA	0x7E	Eight bit SPI shift register
RECONFIG	0x7F	Reconfig - Writing 0x5A here resets FPGA

The 7I43 is delivered with the EPPIOPR8 configuration installed for factory and initial user checking.

SUPPLIED CONFIGURATIONS

USBIOPR8

GENERAL

The USBIOPR8 configuration is a simple six port GPIO configuration almost identical to the EPPIOPR configuration. It is different because the USB interface is a simple bidirectional byte-stream without separate address and data. Because of this a Little Binary Protocol is used to communicate with standard addressable peripherals in the 7I43 FPGA configuration. The GPIO is organized as six eight bit ports, each with an associated Data Direction Register (DDR).The EPPIOPR8 configuration can be used as a starting point for more complicated user configurations.

PORT	DATA REG	DDR	IO BITS	CONNECTOR
0	0x010	0x020	0..7	P4
1	0x011	0x021	8..15	P4
2	0x012	0x022	16..23	P4
3	0x013	0x023	24..31	P3
4	0x012	0x024	32..39	P3
5	0x015	0x025	40..47	P3

In addition to the GPIO bits, the USBIOPR8 configuration has a simple SPI interface to the configuration EEPROM, a LED port, and a reconfiguration port.. The SPI port allows the utility program SCM7I43P to write configuration data to the serial EEPROM. These registers are mapped as follows:

REGISTER	ADDRESS	FUNCTION
LED	0x07C	8 Status LEDS (‘1' = on)
SPICS	0x07D	Single I/O bit to control SPI Chip Select (bit 0)
SPIDATA	0x07E	Eight bit SPI shift register
RECONFIG	0x07F	Reconfig - Writing 0x5A here resets FPGA

SUPPLIED CONFIGURATIONS

USBIOPR8

LBP

LBP is a master slave protocol where the host sends read, write, or RPC commands to the 7I43, and the 7I43 responds. LBP commands always start with a command header byte. This header specifies whether the command is a read or a write, the number of address bytes(0, or 2), number of data bytes(1, through 8).The 0 address size option indicates that the current address pointer should be used. This address pointer will be post incremented by the data size if the auto increment bit is set . RPC commands allow any of 64 stored commands to be executed in response to the single byte command.

LBP DATA READ/WRITE COMMAND

0	1	WR	RID	AI	AS	DS1	DS0

Bit 7.. 6 **CommandType:** Must be 10b to specify data read/write command

Bit 5 **Write:** 1 to specify write, 0 to specify read

Bit 4 **RPCIncludesData:** 0 specifies that data is from stream, 1, that data is from RPC (RPC only, ignored for non RPC commands)

Bit 3 **AutoInc:** 0 leaves address unchanged, 1 specifies that address is post incremented by data size in bytes.

BIT 2 **AddressSize:** 0 to specify current address, 1 to specify 2 byte address.

Bit 1..0 **DataSize:** Specifies data size, 00b = 1 bytes, 01b = 2 bytes, 10 b= 4 bytes, 011b = 8 bytes.

When multiple bytes are specified in a read or write command, the bytes are always written to or read from successive addresses. That is, a 4 byte read at location 0x21 will read locations 0x21, 0x22, 0x23, 0x24. The address pointer is not modified after the command unless the AutoInc bit is set.

SUPPLIED CONFIGURATIONS

USBIOPR8

EXAMPLE LBP COMMANDS

Write 4 bytes (0xAA, 0xBB,0xCC,0xDD) to addresses 0x010,0x011,0x012,0x013 with AutoInc so that the address pointer will be left at 0x014 when the command is completed:

COMMAND BITS	CT1	CT0	WR	RID	AI	AS	DS1	DS0
LBPWrite: 2 add 4 data	0	1	1	0	1	1	1	0
Write Address LSB	0	0	0	1	0	0	0	0
Write Address MSB	0	0	0	0	0	0	0	0
Write data 0	1	0	1	0	1	0	1	0
Write Data 1	1	0	1	1	1	0	1	1
Write Data 2	1	1	0	0	1	1	0	0
Write Data 3	1	1	0	1	1	1	0	1

Write 2 more bytes (0xEE,0xFF) at 0x014 and 0x015:

COMMAND BITS	CT1	CT0	WR	RID	AI	AS	DS1	DS0
LBPWrite: 0 add 2 data	0	1	1	0	0	0	0	1
Write data 0	1	1	1	0	1	1	1	0
Write data 1	1	1	1	1	1	1	1	1

Read 8 bytes at 0x010,0x011,0x012,0x013,0x014,0x015,0x016,0x017:

COMMAND BITS	CT1	CT0	WR	RID	AI	AS	DS1	DS0
LBPRead: 2 add 8 data	0	1	0	0	0	1	1	1
Read Address LSB	0	0	0	1	0	0	0	0
Read Address MSB	0	0	0	0	0	0	0	0

SUPPLIED CONFIGURATIONS

USBIOPR8

RPC COMMANDS

RPC commands allow previously stored sequences of read/write commands to be executed with a single byte command. Up to 64 RPC's may be stored. RPC write commands may include data if desired, or the data may come from the USB serial data stream. RPCs allow significant command compression which improves communication bandwidth.

LBP RPC COMMAND

1	0	RPC5	RPC4	RPC3	RPC2	RPC1	RPC0

Bit 7..6 **CommandType:** must be 10b to specify RPC

Bit 5..0 **RPCNumber:** Specifies RPC 0 through 63

In the USBIOPR8 configuration, each RPC command has native size of 16 bytes and start 16 byte boundaries in the RPC table area but can cross 16 byte boundaries if larger then 16 byte RPCs are needed. The stored RPC commands consist of LBP headers and addresses, and possibly data if the command header has the RID bit set. RPC command lists are terminated by a 0 byte.

The RPC table starts at address 0xC00 in the USBIOPR8 configuration. This means RPC0 starts at 0xC00, RPC1 starts at 0xC10, RPC2 starts at 0xC20 and so on.

SUPPLIED CONFIGURATIONS

USBIOPR8

EXAMPLE RPC COMMAND LIST

This is an example stored RPC command list. Note RPC command lists must start at a 16 byte boundary in the RPC table but an individual RPC list can extend until the end of the table. This particular RPC example contains 3 LBP commands and uses 11 bytes:

Command1. Writes two data bytes to port 0x10, 0x11 with 2 data bytes supplied by host

Command2. Reads two data bytes from port 0x12,0x13

Command3. Writes a single byte (0xAA) to port 0x14, data contained in RPC table

COMMAND BITS	CT1	CT0	WR	RID	I	AS	DS1	DS0
LBPWrite: 2 add 2 data	0	1	1	0	0	1	0	1
Write Address LSB	0	0	0	1	0	0	0	0
Write Address MSB	0	0	0	0	0	0	0	0
LBPRead: 2 add 2 data	0	1	0	0	0	1	0	1
Read Address LSB	0	0	0	1	0	0	1	0
Read Address MSB	0	0	0	0	0	0	0	0
LBPWrite 2 add 1 data	1	0	1	1	0	1	0	1
Write Address LSB	0	0	0	1	0	1	0	0
Write Address MSB	0	0	0	0	0	0	0	0
Write Data	1	0	1	0	1	0	1	0
Terminator	0	0	0	0	0	0	0	0

The data stream for this RPC would consist of these 3 bytes:

COMMAND BITS	CT1	CT0	R5	R4	R3	R2	R1	R0
RPC 5	1	0	0	0	0	1	0	1
Data 0 for Write 1	0	1	0	1	0	1	0	1
Data 1 for Write 1	1	1	0	0	1	1	0	0

AVAILABLE DAUGHTER CARDS

PART NUMBER	FUNCTION
7I30	Quad 3A 36V HBridge
7I31	Debug LED card
7I32	Dual stepper driver (microstepping)
7I33,7I33T	Quad analog servo interface
7I34	8 TX + 8 RX pair RS-422 interface
7I34-R	16 RX pair RS-422 interface
7I37,7I37T	8 output 16 input isolated I/O
7I39	Dual 3 phase H-Bridge
7I40	Dual H-Bridge
7I42	I/O protector
7I44	8 Channel RS-422 to RJ45 breakout
7I45	Octal16 bit A-D analog servo interface
7I64	24 input, 24 output isolated I/O

REFERENCE INFORMATION

SPECIFICATIONS

	MIN	MAX	NOTES
SUPPLY VOLTAGE	4.5V	5.25V	
3.3V CURRENT SUPPLIED TO P3,P4	----	800 mA	Typically limited to ~200 mA in USB powered case.
5V CURRENT SUPPLIED TO P3,P4	----	2A	Typically limited to ~150 mA in USB powered case.
1.2V CORE POWER CURRENT	----	1A	1A = ~300 mA of 5V draw. Depends on FPGA configuration
TEMPERATURE -C VERSION	0^{o}C	70^{o}C	
TEMPERATURE -I VERSION	-40^{o}C	85^{o}C	