7I60 MANUAL

V1.7

This page intentionally not blank

Table of Contents

Table of Contents

GENERAL

DESCRIPTION

The 7I60 is a standalone (no bus) version of the FPGA based Anything I/O card series. It provides 96 I/O bits and 4 serial ports.

There are 2 RS-232 ports and 2 RS-422/RS-485 ports. One of the RS- 232 serial ports is used for downloading initial configurations to the on-card Flash EEPROM for FPGA configuration, the other ports can be used for any purpose.

The 96 I/O bits are available on four 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 status LEDs are provided for debugging puposes as are 8 FPGA driven LEDs and a FPGA driven beeper.Several I/O interface daughter cards are available for the 7I60. 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. 2 daughter cards can plug directly onto the 7I60.

Many IO configuration files are provided with the 7I60 including simple remote I/O, smart 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.

2 FPGA system clocks are provided: a 50MHZ Oscillator and a 32,40,50,52,60,66, 75 and 80 MHz selectable frequency PLL.

The 7I60 uses a 200K gate Xilinx SpartanII FPGA. Free development tools for The SpartanII are available (Xilinx WebPack) from Xilinx's web site.

HARDWARE CONFIGURATION

GENERAL

Hardware setup jumper positions assume that the 7I60 card is oriented in an upright position, that is, with the 9 pin serial connectors towards the person doing the configuration, and the power connectors on top.

HOST BAUD RATE

The host interface serial port baud rate is set via jumper W5,W6, and W7. These jumpers are only read at power-up or PIC processor reset.

W5	W6	W7	BAUD RATE
DOWN	DOWN	DOWN	9600
DOWN	DOWN	UP	19200
DOWN	UP	DOWN	38400
DOWN	UP	UP	56700
UP	DOWN	DOWN	115200
UP	DOWN	UP	230400
UP	UP	DOWN	460800
UP	UP	UP	921600

DEBUG JUMPER

It is possible for a bad FPGA configuration to interfere with host communication, or cause some other system problem. Also the PIC EEPROM jump option can be improperly setup or the alternate code section of the PIC could be faulty. To be able to recover from these possible setup faults, The DEBUG jumper W4 is provided.

Jumper W4 enables or disables FPGA configuration at startup. Jumper W4 also determines whether the power on PIC jump vector in EEPROM is used. When W4 is in the UP position (default), the FPGA is automatically configured from the on card flash memory at power-up or processor reset, and the PIC jump vector in EEPROM is taken When W4 is in then DOWN position, the automatic FPGA configuration process is disabled, and the PIC jump vector is ignored.

.

HARDWARE CONFIGURATION

GCLK2 SOURCE

The CLK2 input to the FPGA can come from the on card 50 MHz oscillator or from I/O bit 48. When W3 is in the left hand position (default), GCLK2 comes from the 50 MHz oscillator. When W3 is oin the right hand position, GCLK2 comes from I/O bit 48.

CONNECTOR POWER

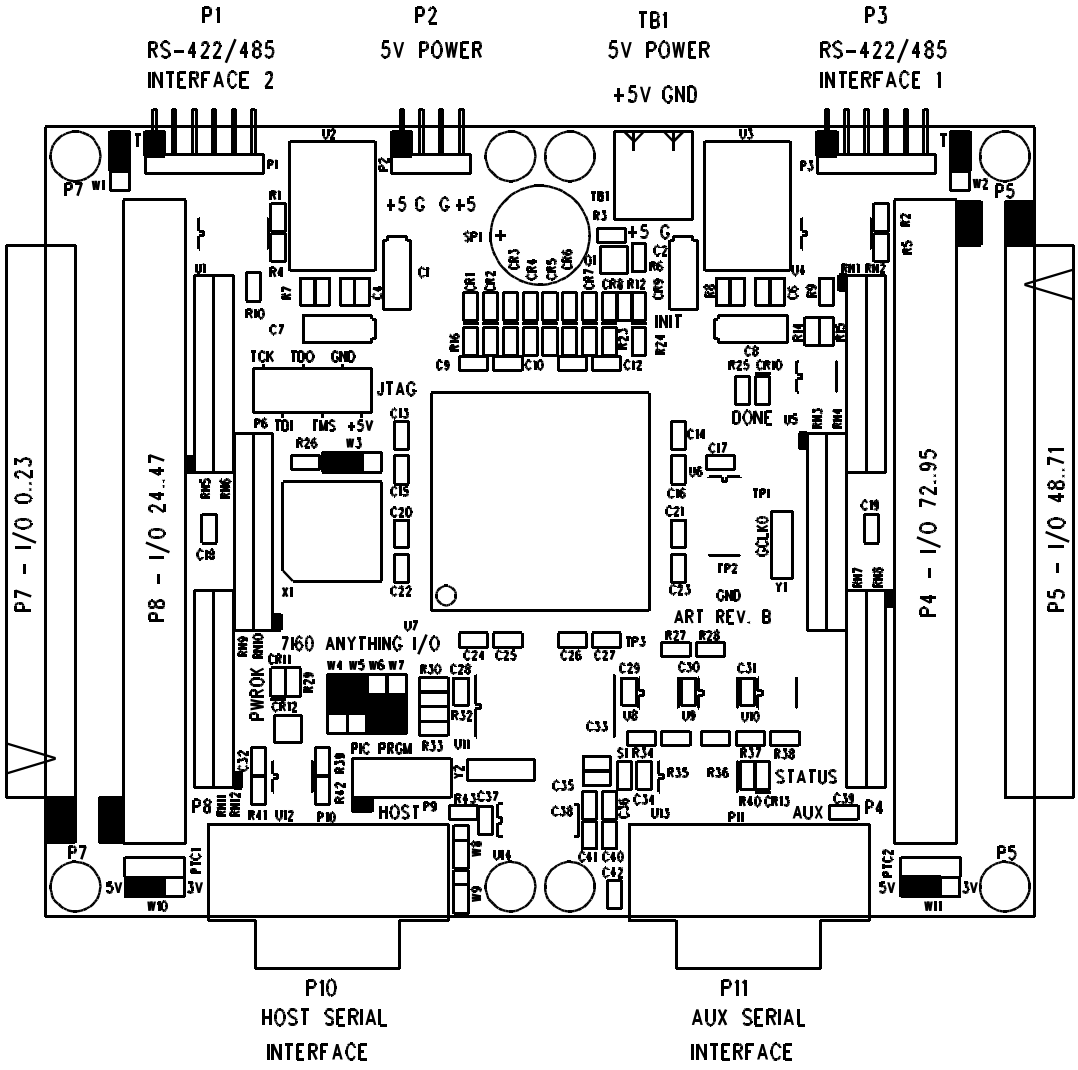
The power connection on the I/O connectors can supply either 3.3V or 5V power. Supplied power should be limited to 400 mA total. W10 selects the power supplied to P7 (I/O 0..23) and P8 (I/O 24..47). W11 selects the power supplied to P5 (I/O 48..71) and P4 (I/O 72..95) . When W10 or W11 are in the left position, 5V power is supplied to the connector and associated pullup resistors. When W10 or W11 are in the right position, 3.3V is supplied. Note that most Mesa I/O adapter cards that connect to Anything I/O cards require 5V.

RS-422/485 TERMINATION

The two RS-422/485 interfaces on the 7I60 can have the receive pair terminated with 130 Ohm resistors to prevent reflections. Termination should be enabled if the 7I60 is at the endpoint of a RS-422 or RS-485 link. Jumpers W1 and W2 control termination. If the jumpers are in the ‘up’ position, termination is enabled. If the jumpers are in the down position, termination is disabled.

CONNECTORS

CONNECTOR LOCATIONS AND DEFAULT JUMPER POSITIONS



CONNECTORS

I/O CONNECTORS

P7, P8, P5, and P4 are the 7I60s 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 7I60IO.PIN file on the 7I60 distribution disk. 7I60 IO connector pinouts are as follows:

P7 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

7I60 I/O CONNECTORS

P8 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

7I60 I/O CONNECTORS

P5 CONNECTOR PINOUT

PIN	FUNC	PIN	FUNC	PIN	FUNC	PIN	FUNC
1	IO48	2	GND	3	IO49	4	GND
5	IO50	6	GND	7	IO51	8	GND
9	IO52	10	GND	11	IO53	12	GND
13	IO54	14	GND	15	IO55	16	GND
17	IO56	18	GND	19	IO57	20	GND
21	IO58	22	GND	23	IO59	24	GND
25	IO60	26	GND	27	IO61	28	GND
29	IO62	30	GND	31	IO63	32	GND
33	IO64	34	GND	35	IO65	36	GND
37	IO66	38	GND	39	IO67	40	GND
41	IO68	42	GND	43	IO69	44	GND
45	IO70	46	GND	47	IO71	48	GND
49	POWER	50	GND				

CONNECTORS

7I60 I/O CONNECTORS

P4 CONNECTOR PINOUT

PIN	FUNC	PIN	FUNC	PIN	FUNC	PIN	FUNC
1	IO72	2	GND	3	IO73	4	GND
5	IO74	6	GND	7	IO75	8	GND
9	IO76	10	GND	11	IO77	12	GND
13	IO78	14	GND	15	IO79	16	GND
17	IO80	18	GND	19	IO81	20	GND
21	IO82	22	GND	23	IO83	24	GND
25	IO84	26	GND	27	IO85	28	GND
29	IO86	30	GND	31	IO87	32	GND
33	IO88	34	GND	35	IO89	36	GND
37	IO90	38	GND	39	IO91	40	GND
41	IO92	42	GND	43	IO93	44	GND
45	IO94	46	GND	47	IO95	48	GND
49	POWER	50	GND				

CONNECTORS

HOST SERIAL CONNECTOR

P10 is the host serial interface, used for communication. P10 is a 9 pin female DE9 connector that will connect directly to PC type serial ports with a male-female extension cable.

P10 CONNECTOR PINOUT

PIN	FUNCTION	DIRECTION
1	RTS1 = HANDSHAKE OUT	FROM 7I60
2	TXD1 = XMIT DATA OUT	FROM 7I60
3	RXD1 RECV DATA IN	TO 7I60
4	N/C	
5	GND	
6	RTS1 = HANDSHAKE OUT	FROM 7I60
7	CTS1 = HANDSHAKE IN	TO 7I60
8	RTS1 = HANDSHAKE OUT	FROM 7I60
9	N/C	

CONNECTORS

AUX SERIAL CONNECTOR

P11 is the AUX serial interface. P11 is a 9 pin male DE9 connector with a pinout that matches 9 pin PC serial ports.

P11 CONNECTOR PINOUT

PIN	FUNCTION	DIRECTION
1	NC	
2	RXD2 RECV DATA IN	TO 7I60
3	TXD2 = XMIT DATA OUT	FROM 7I60
4	RTS2 = HANDSHAKE OUT	FROM 7I60
5	GND	
6	NC	
7	RTS2 = HANDSHAKE OUT	FROM 7I60
8	CTS2 = HANDSHAKE IN	TO 7I60
9	RI2	TO 7I60

CONNECTORS

RS-422/RS-485 CONNNECTORS

P1 and P3 or RS-422/RS-485 interface connectors. The RS-422/RS-485 interfaces are driven by the FPGA. Connector pinout is as follows:.

P1,P3 CONNECTOR PINOUTS

PIN	FUNCTION	DIRECTION
1	GND	
2	RX- RECV- DATA IN	TO 7I60
3	RX+ RECV+ DATA IN	TO 7I60
4	TX+ XMIT+ DATA OUT	FROM 7I60
5	TX- XMIT- DATA OUT	FROM 7I60
6	GND	

CONNECTORS

7I60 JTAG CONNECTOR

P6 is a JTAG programming connector. It is not normally used since the 7I60 can be programmed via the serial interface, but can be useful when debugging.

P6 CONNECTOR PINOUT

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

POWER CONNECTORS

The 7I60 has two 5V power connectors., TB1, a 2 position terminal block, and P2 a four pin male header. TB1 5V power polarity _is_ marked on the 7I60 card but will be described here. When the 7I60 is oriented such that TB1 is at the top of the 7I60 CARD, 5V is on the left side and GND is on the right. P2 pin-out is as follows:

P2 CONNECTOR PINOUT

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

OPERATION

FPGA

The 7I60 uses a Xilinx Spartan-II FPGA in a 208 pin QFP package , PN XC2S200-5PQ208C.

HOST INTERFACE

The 7I60 uses a serial interface for initial communication and writing to the on card flash memory used to configure the FPGA at power-up. The host serial interface can also be used to communicate with the FPGA once the FPGA is configured. The host interface is and RS-232 interface with a 9 pin female connector with a pin-out compatible with PC type serial ports. The communication baud rate is determined by baud rate jumpers described in the hardware setup section. The 7I60 uses RTS to indicate that it is able to receive characters. The host normally can ignore RTS except at the highest baud rates (>230.4 Kbaud)

COMMAND SET

The host serial interface supports several commands. These include commands for initializing the configuration flash memory, reading and writing the FPGA interface port, clock frequency selection and other functions. All commands consist of two letters and may have hexadecimal parameters. All returned data is hexadecimal or ASCII strings. Hexadecimal data can be upper or lower case. Spaces and commas are ignored in the input. A command is terminated with a Carriage Return (CR). Commands echo a Carriage Return when complete. If the command echoes data, the Carriage Return will be echoed after that data. All failed commands (invalid command code, bad data etc) will echo a '?'

The following table lists the host interface commands. In the table, > means Carriage Return, D means a hexadecimal data nibble, A means a hexadecimal Address nibble, and C means a hexadecimal command nibble.

OPERATION

COMMAND SET

READ COMMANDS

COMMAND	FORMAT	FUNCTION
RB	RB > DD	Read intelligent I/O base address
RC	RC > DD	Read Checksum
RD	RD AA > DD	Read FPGA interface byte Direct
RE	RE AA > DD	Read EEPROM byte
RF	RF AA > DDDDDDD....	Read Flash block - 4K bytes (Rev <64)
RF	RF AAAA > DDDDDD....	Read Flash block - 256 bytes (Rev > 63)
RL	RL CCCC > DDDDDDDD	Read Long (intelligent interface)
RP	RP > DD	Read Protect status (not 0 is protected)
RR	RR AAAA > DDDD	Read PIC ROM word at AAAA
RS	RS > DD	Read FPGA interface Status bits (Busy in MSB, ATN in LSB)
RW	RW CCCC > DDDD	Read Word (intelligent interface)
RI	RI > DDDD	Read IFIFO word (FIFOed interface)
RQ	RQ > DDDD	Read QFIFO word (FIFOed interface)
RM	RM CC > DDDDDD...	Read multiple words from IFIFO,CC is data words to read. (FIFOed interface)
Rm	Rm CC > DDDDDD...	Read multiple words from QFIFO,CC is data words to read. (FIFOed interface)
Rd	Rd AA > DDDD	Read word direct (FIFOed interface)
Rs	Rs > DDDD	Read SoftDMC Status register A (FIFOed interface)

COMMAND SET

WRITE COMMANDS

COMMAND	FORMAT	FUNCTION
WB	WB DD >	Write Base address (set intelligent I/O command port base address)
WD	WD AADD >	Write FPGA interface byte Direct
WE	WE AADD >	Write PIC EEPROM location AA with DD
WF	WF AA > DDDDDDD...	Write Flash - bytes data (Rev <64)
WF	WF AAAA > DDDDD...	Write Flash - 256 bytes data (Rev >63)
WL	WL CCCCDDDDDDDD >	Write Long (intelligent interface)
WO	WO DD >	Write Oscillator (change GCLK0 frequency)
WP	WP DD >	Write Protect (DD <> 0 is protected) Protects EEPROM/Flash writes and PIC jumps
WR	WR AAAA DDDD >	Write PIC ROM location AAAA with DDDD
WW	WW CCCCDDDD >	Write Word (intelligent interface)
WI	WI DDDD >	Write word to IFIFO (FIFOed interface)
WQ	WI DDDD >	Write word to QFIFO (FIFOed interface)
Wd	Wd AA DDDD >	Write word direct (FIFOed interface)

OPERATION

COMMAND SET

MISC COMMANDS

COMMAND	FORMAT	FUNCTION
EF	EF AA >	Erase Flash configuration memory block AA, Erase block size is 4K bytes for Rev < 64 and 64K bytes for Rev >63
EC	EC >	Erase (clear) checksum
II	II > ‘7I60'	Inquire Identity
IR	IR > PIC CODE REV.	Inquire Revision
GC	GC >	Configure FPGA from flash memory
GO	GO AAAA >	PIC jump (careful...)
GP	GP >	Load embedded processor ROM from flash

OPERATION

PIC EEPROM USAGE

A few of the PICs on CHIP EEPROM locations are used for default 7I60 setup information. These EEPROM locations and functions are as follows:

LOCATION	FUNCTION	DEFAULT VALUE
00	Startup GCLK0 select	0x04 = 50.121 MHz
01	Intelligent I/O Base address	0xA2 (dont change!)
02	FPGA configuration enable	0xFF
03	Starting FPGA config block	0x00 (Unimplemented)
04	Number of FPGA config blocks	0x28 (Unimplemented)
05	FPGA program D/L enable	0x00
06	FPGA program start block	0x30 (Unimplemented)
07	FPGA program size	0x01 (Unimplemented)
08	Startup Jump low	0x00
09	Startup jump high	0x00

OPERATION

FPGA CONFIGURATION

Before the 7I60/7I60M can do anything useful it must have its FPGA configuration data downloaded from the host CPU to the flash memory used to configure the FPGA at power up. The flash configuration memory on the 7I60 card is written by sending the Xilinx configuration bit file to the 7I60 over the host serial port. The sequence of commands used to download a bit file to the 7I60 is as follows:

For 7I60 with Firmware rev < 64

SERIAL HOST COMMAND	FUNCTION
WP 00	Unprotect flash write operations
EF 00	Erase flash block 00
WF 00 DD DD DD DD	Write flash block 00 (with first 4K bytes of bit file)
EF 01	Erase flash block 01
WF 01 DD DD DD DD	Write flash block 01 (with next 4K bytes of bit file)

The last two commands are repeated for all the blocks in the bit file (40 decimal =0x28h blocks total)

For 7I60 with Firmware rev > 63

SERIAL HOST COMMAND	FUNCTION
WP 00	Unprotect flash write operations
EF 00	Erase flash block 00 (first 65536 bytes)
EF 01	Erase flash block 01 (second 65536 bytes)
EF 02	Erase flash block 02 (third 65536 bytes for 192K bytes total)
WF 0000 DD DD DD DD	Write flash block 0000 (with first 256 bytes of bit file)
WF 0001 DD DD DD DD	Write flash block 0000 (with next 256 bytes of bit file)

The WF command is repeated for the remaining blocks (768 decimal = 0x0300h blocks total)

OPERATION

SC7I60

A utility program SC7I60.EXE is provided to send configuration files to the 7I60. The Pascal and C source for this program is available on the distribution disk, and can be used as an example for writing a custom version of download software. SC7I60 is invoked with the FPGA configuration file, serial port address, and serial port baud rate on the command line:

```
SC7I60 FPGAFILE.BIN COM1 115200
```

Would send the configuration file FPGAFILE.BIN to the 7I60 using the serial port at 3F8 and a baud rate of 115200 baud.

```
SC7I60 FPGAFILE.BIN COM1 115200 -R
```

Does the same thing except a that a block by block readback and compare is done to verify the the flash EEPROM block contents. This doubles the configuration download time.

SC7I60 uses binary FPGA configuration files. These files are standard Xilinx BIT files.

For Windows 2000 and higher the SC7I60W program should be used. Its operation is identical but it will have better performance under windows.

Due to PIC overhead and flash EEPROM programming time, hardware handshaking (RTS-CTS) must be used when downloading FPGA configurations at baud rates higher than 230.4 K baud.

OPERATION

CLOCK SIGNALS

The 4 FPGA clock signals on the 7I60 are routed to 4 separate clock sources. GCLK0 connects to a PLL clock generator with 8 selectable frequencies. GCLK1 connects to the I/O bit 24 signal, GCLK2 connects to a 50 MHz crystal oscillator on the 7I60 card or I/O bit 48 depending on the setting of JPR W3, and GCLK3 connects to the I/O bit 72 signal.

PLL CLOCK FREQUENCIES

GCLK0 is generated by a PLL chip with 8 selectable frequencies. The frequency selected at powerup is determined by the PIC EEPROM data at location 0x00. The frequency can be also be selected dynamically with the serial host Write Oscillator command. The 8 available frequencies are as follows:

SELECT DATA	FREQUENCY
0x00	75.1814 MHz
0x01	31.9452 MHz
0x02	60.1450 MHz
0x03	40.0967 MHz
0x04	50.1209 MHz
0x05	66.4869 MHz
0x06	80.1934 MHz
0x07	51.9109 MHz

For example to set the startup GCLK0 frequency to 40.0967 MHz, the PIC EEPROM location 0x00 would be set to the value 0x03:

WE 00 03

After this has been done, and the 7I60 reset or power cycled,the GCLK0 frequency will be 40.0967 MHz.

To temporarily set the GCLK0 frequency to 80.1934 MHz you could issue the command:

WO 06

OPERATION

PIC-FPGA INTERFACE

The serial host interface gives access to a simple 8 bit bi-directional port on the PIC that connects to the FPGA. There are 7 host commands that use this interface to access the FPGA from the host serial port: RD, RS, RW, RL, WD, WW, and WL.

The interface uses a a simple multiplexed data/address bus arrangement. The PIC's view of the FPGA interface is 256 contiguous eight bit read/write registers. The PIC/FPGA interface pin-out is as follows:

PIC PIN		FPGA PIN	FUNCTION	DIRECTION
21	(RB0)	30	DAB 0	BIDIR
22	(RB1)	27	DAB 1	BIDIR
23	(RB2)	23	DAB 2	BIDIR
24	(RB3)	21	DAB 3	BIDIR
25	(RB4)	18	DAB 4	BIDIR
26	(RB5)	16	DAB 5	BIDIR
27	(RB6)	14	DAB 6	BIDIR
28	(RB7)	10	DAB 7	BIDIR
2	(RA0)	15	ICLK	FROM PIC
3	(RA1)	17	R/W	FROM PIC
4	(RA2)	20	BUSY	TO PIC
5	(RA3)	22	ATT	TO PIC

OPERATION

PIC-FPGA INTERFACE

The built in firmware in the PIC uses this interface to access the FPGA. *The details of this operation are important if you wish to use the serial host interface built into the PIC to access your own FPGA configurations.*

The FPGA /PIC interface functions as follows: At the start of a cycle, ICLK is low, and R/W is high The first thing that is done is that the address is output on the DA bus. Then ICLK is set high to latch the address into the FPGA. On a write cycle, the next step is for the PIC to output the data on the DA bust. Then ICLK is set low to latch the data into the FPGA. On a read cycle, after the address is latched, the pic sets the DA bus for input mode,and then R/W is brought low to enable the FPGA data onto the DA bus. The PIC can then read the FPGAs data.

The RD and WD (Read Direct and Write Direct) host interface commands simple read and write bytes of data to and from selected addresses in the FPGA. For example

WD 04 AA

would write 0xAA to location 0x04 in the FPGA

RD 55

would read back the contents of FPGA location 0x55

INTELLIGENT INTERFACE

The RW, WW, RL and WL commands assume an intelligent interface device in the FPGA. These are used with the motion control FPGA configurations that have a processor inside the FPGA. The intelligent interface assumes that the FPGA has a set of 4 or 6 eight bit registers in in the following arrangement:

BASE	Low byte of command register
BASE+1	High byte of command register
BASE+2	Byte 0 of data register
BASE+3	Byte 1 of data register
BASE+4	Byte 2 of data register
BASE+5	Byte 3 of data register

OPERATION

INTELLIGENT INTERFACE

BASE is the base address of the command register and is stored in PIC EEPROM location 0x01. BASE can also be change dynamically via the WB (Write Base) command. Operation of the intelligent interface commands is as follows:

The RW (Read Word) command: The PIC first polls the BUSY bit (RA2 = FPGA pin 20) and waits for it to go low. Then it writes the 16 bit command word to BASE and BASE+1 and then polls the interface BUSY bit. When the busy bit returns low, the PIC reads bytes 0 and 1 from the data registers at BASE+2 and BASE+3, concatenates them and sends the 16 bit data back to the host.

The RL (Read Long)command is the same as RW except that the PIC reads bytes 0,1,2, and 3 from the data register, concatenates them and sends the 32 bit data back to the host.

The WW (Write Word)command: The PIC first polls the BUSY bit and waits for it to be low. Then it writes the 16 bit data to data registers 0 and 1. The it writes the 16 bit command to command registers 0 and 1.

The WL (Write Long) command is the same as WW except that the PIC writes data registers 0,1,2,and 3 instead of just 0, and 1.

The RS (Read Status) command allows the host to read back the intelligent interface status bits. The byte of data returned by the RS command contains the BUSY bit in the MSB and the ATT bit in the LSB.

OPERATION

FIFOED INTERFACE

In addition to the Intelligent interface the 7I60 PIC supports communication with the FIFOed interface used by SoftDMC version 4.0 and greater. 8 new commands have been added to support the FIFOed interface.

The RI (Read IFIFO) command reads data from the FPGAs IFIFO. If no data is available in the IFIFO the RI command will wait for data to be available for about 1 mS. If no data was available after 1 mS, a NAK will be transmitted, otherwise the 16 bit data read from the IFIFO is transmitted back to the host. Note that a complete 16 bit SoftDMC read transaction requires a SoftDMC command write (with WI) and a following read with RI. A 32 bit read transaction requires a command write (with WI) followed by 2 RI commands to read the data (least significant word first).

The RQ command is identical to the RI command except that it accesses the QFIFO.

The Rd command does a direct 16 bit port read from the FIFOed interface. This can be used for reading Status register B and SoftDMC firmware updates.

The RM command reads multiple words from the IFIFO. The read count is a byte of hex data that follows the command. The RM command will always return the requested amount of data, as it does not check whether there is valid data in the FIFO before reading. This means that you should either read the IRBFIFO count or the half full status bit before issuing a RM command, or it may return invalid data.

The Rm command is identical to the RM command except that it accesses the QFIFO.

The Rs command reads SoftDMC status register A and returns the 16 bit data to the host. This allows reading FIFO and interrupt status bits.

The WI command writes a 16 bit word to the IFIFO. A 16 bit parameter write requires 2 WI commands, the first with the SoftDMC command and the next with the data. A 32 bit parameter write requires 3 WI commands, a SoftDMC command followed by 2 words of data, least significant word first.

The WQ command is identical to the WI command except that it accesses the QFIFO.

The Wd command does a direct 16 bit port write to the FIFOed interface. This can be used for Resetting SoftDMC, clearing interrupts and SoftDMC firmware updates.

OPERATION

USER PIC FIRMWARE

The top 2K of the PICs 4K address space is available for user programs or firmware updates. PIC firmware can be updated from the serial host interface by using the WR (Write ROM) command. If you wish the top ROM code to be executed , you can issue a GO 0800 command to the host interface. The top half firmware can be executed automatically at startup (just after FPGA configuration) by setting the PICs EEPROM jump vector to the second page:

EEPROM 8 = 00, EEPROM 9 = 08

A utility program SP7I60 is provided for writing the top half of the PICs ROM.

SP7I60 is invoked with the binary PIC code file, the COM port and the baud rate on the command line:

```
SP7I60 PIC.BIN 3F8 115200
```

OPERATION

LEDS

The 7I60 has 8 FPGA driven user LEDS. These green LEDS are located in the top cenre of the card below the speaker. 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 7I60MISC.PIN file for FPGA pin locations of the LED signals.

In addition to the user LEDs there are four other LEDS that display board status information.

On the lower left hand side of the card is a green POWER-GOOD LED. This lights when 5V power is within tolerance (>4.75V) and the reset interval has passed.

On the lower right had side of the card is the STATUS LED. This LED is controlled by the PIC and lights when the PIC is busy.

IO LEVELS

The Xilinx FPGAs used on the 7I60 have programmable I/O levels for interfacing with different logic families. The 7I60 does not support use of the I/O standards that require input reference voltages, so only 5 I/O options can be used. The available I/O options are LVTTL (5V tolerant), PCI33_5 (5V tolerant), PCI33_3, PCI66_3, and LVCMOS2. Two of the I/O options allow 5V inputs.

Note that even though the 7I60s FPGA can tolerate 5V signal inputs, its outputs will not swing to 5V. The outputs are push pull CMOS outputs that will drive to the output supply rail of 3.3V. This is sufficient for TTL compatibility but may cause problems with some types of loads. For example when driving an LED that has its anode connected to 5V, the 3.3V high level may not completely turn the LED off. To avoid this problem, either drive loads that are ground referred, Use 3.3V as the VCC for VCC referred loads, or tristate the output signals when no drive is desired (open drain).

SUPPLIED CONFIGURATIONS

I60LOOP

The I60LOOP Is used for I/O bit testing, but can also be used as a 96 bit remote I/O port. The source for I60LOOP is a good example of the PIC interface, and can be used as a starting point for your own designs. The port base address is 0xC0 hex and 24 IO locations are used. The I60LOOP is a byte device, all accesses read or write single bytes. Each data registers has an associated Data Direction Register (DDR) that determines the direction of each bit. The register map is as follows:

PORT	DATA REG	DDR	IO BITS	CONNECTOR
1	C0	C1	0..7	P7
2	C2	C3	8..15	P7
3	C4	C5	16..23	P7
4	C6	C7	24..31	P8
5	C8	C9	32..39	P8
6	CA	CB	40..47	P8
7	CC	CD	48..55	P5
8	CE	CF	56..63	P5
9	D0	D1	64..71	P5
10	D2	D3	72..79	P4
11	D4	D5	80..87	P4
12	D6	D7	88..95	P4

Each I/O bit can be individually programmed to be input or output. All I/O bits will be input on startup. The 7I60 is delivered with the I60LOOP configuration installed for for factory and initial user checking.

SUPPLIED CONFIGURATIONS

I60LOOP

The supplied program 7I60LOOP.EXE can be used to do a loopback test to verify all parallel I/O bits. 7i60LOOP is invoked with the port address and baud rate on the command line:

```
7I60LOOP COM1 115200
```

Would do a loopback test of the 7I60 connected to COM1 at a baud rate of 115200 baud. Two 50 conductor flat cables are needed to do the loopback test, on cable connects from P7 to P8 and the other connects from P4 to P5. I60LOOP also toggles the user LEDS.

For windows > NT an alternate program is provided for loopback testing. This is 7I60LPW.EXE. It functions identically to 7I60LOOP but runs faster on newer versions of windows.

WAVEGEN

Wavegen is a twelve channel sine/arbitrary waveform digital oscillator. Each Oscillator consists of a 48 bit phase accumulator driving a 512 deep by 8 wide lookup table The output of the each of the twelve oscillators is brought out to the IO connectors on the 7I60. The initial data in the tables when first configured is a sine waveform, but this data can be changed from the host for generation of other waveforms. The clock is from the PLL, allowing base frequencies up to 80 MHz. Base address's of the oscillators are at 0x10H intervals (Oscillator 0 base address = 0x00, Oscillator 1 base address is 0x10H etc. Oscillator register map is as follows:

PORT	FUNCTION
X0	FREQUENCY BITS 0..7
X1	FREQUENCY BITS 8..15
X2	FREQUENCY BITS 16..23
X3	FREQUENCY BITS 24..31
X4	FREQUENCY BITS 32..39
X5	FREQUENCY BITS 40..47
X6	NOT USED
X7	NOT USED

SUPPLIED CONFIGURATIONS

WAVEGEN

PORT	FUNCTION
X8	MEMORY POINTER (BITS 0..7)
X9	MEMORY POINTER (BIT 8 (in Lsb))
XA	MEMORY DATA

All ports are write only, but there is a signature port at 0XE0 (=0xA5) to determine WAVEGEN presence.

When writing a new frequency the data bytes should be written from the least significant byte to the most significant byte. This is because the actual phase accumulator frequency latch will not be updated until the most significant byte (at X5) is written.

The initial memory contents is a sine lookup table but this can be changed if desired. To change the memory contents you first write the address in memory you wish to change to the memory pointer registers (X8 and X9), then you write the desired data for that location to the memory data register (XA). This has to be repeated for all 512 locations for a full update.

The 8 bit memory output is available at the I/O pins. Oscillator 0's output is on I/O pins 0..7, Oscillator 1's outputs are on I/O 8..5 etc etc. A simple resistor summing DAC on the sets of 8 outputs can be used to generate analog output signals.

Oscillator frequency will be PLLCLK*FREQUENCY/2^48. Oscillator 0's outputs drive the LEDS.

SOFTDMC

*Soft*DMC is a 4 or 8 axis servo or stepper motion controller using a built in DSP processor. The SoftDMC motion controller is compatible with the 7I25, 7I27 dual Hbridges, the 7I30 quad Hbridge, the 7I32 dual Microstepping driver, the 7I33 Analog servo interface, and the 7I39 dual 3 Phase BLDC driver. Please see the *Soft*DMC manual for more information.

AVAILABLE DAUGHTER CARDS

PART NUMBER	FUNCTION
7I30	Quad 3A 36V HBridge
7I31	Debug LED card
7I32	Dual stepper driver (microstepping)
7I33	Quad analog servo interface
7I34	8 TX + 8 RX pair RS-422 interface
7I34-R	16 RX pair RS-422 interface
7I35	10/100 BaseT Ethernet PHY
7I36	USB interface

REFERENCE INFORMATION

SPECIFICATIONS

	MIN	MAX	NOTES
SUPPLY VOLTAGE	4.75V	5.25V	
OPERATING SUPPLY CURRENT	----	1A	Depends on FPGA configuration
SUPPLY CURRENT FPGA OFF	----	200 mA	
TEMPERATURE -C VERSION	0°C	70°C	
TEMPERATURE -I VERSION	-40°C	85°C	