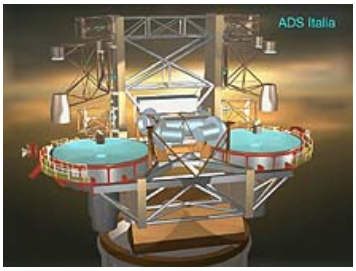

LBT PROJECT
2x8.4m TELESCOPE

Doc.No. : 675s002
Issue : a
Date : 9-March-2007

LBT PROJECT

2 X 8.4m OPTICAL TELESCOPE

General Purpose SERDES Communications PC Board Description and Operations Manual

	Signature	Date
Prepared	Mike Gusick	9 March 2007
Reviewed	David Ashby	19 March 2007
Approved		

1. Revision History

Issue	Date	Changes	Responsible
a	9-Mar-07	First draft	Mike Gusick

2. Table Of Contents

1. Revision History 2
2. Table Of Contents 3
3. About this document 4
   3.1. Reference Documents 4
   3.2. Abbreviations 4
4. Introduction 5
   4.1. GPSCOM Board Hardware 5
   4.2. GPSCOM Board General Design 7
5. FPGA Hardware and HDL code 8
6. SERDES 11
7. Fiber Transceiver 12
8. Voltage Supervisors and Watchdog Timer 13
9. Application Board Interface 14

3. About this document

This document describes the hardware and firmware design of the general purpose communications board. The goal is to describe the function and operation of the board when configured to operate in the instrument rotator system.

3.1. Reference Documents

[RD1] CAN670s007a - Instrument Rotator and Cable Chain Detailed Design Description
[RD2] CAN675s001a - General Purpose SERDES Communications Protocol Standard
[RD3] CAN675s003a - Instrument Rotator and Cable Chain Reef SERDES Interface Module Description and Operations Manual
[RD4] CAN675s004a - Instrument Rotator and Cable Chain Interface Module Description and Operations Manual
[RD5] CAN675r001a.pdf General Purpose SERDES Communications PC Board Schematics
[RD6] CVS archive TBD GPSCOM board VHDL source code
[RD7] CAN675x001a Xilinx Virtex II FPGA
[RD8] CAN675x002a Cypress Hotlink II Single Channel Hotlink II Transceiver
[RD9] CAN675x003a Agilent HFBR-5710L Small Form Factor Pluggable Optical Transceiver

3.2. Abbreviations

BGA	Ball Grid Array
Bd	Buad (i.e. signal rate/data rate)
CRC	Cyclic Redundancy Check
FPGA	Field Programmable Gate Array
GPSCOM	General Purpose SERDES Communications
GBd	Giga-Baud
IC	Integrated Circuit
I/O	Input-Output
MCS	Mount Control System
PROM	Programmable Read Only Memory
SERDES	Serializer-Deserializer
SFP	Small Form-Factor Pluggable
SRAM	Static Random Access Memory
VHDL	Very High Speed Integrated Circuit Hardware Description Language
WDT	Watch Dog Timer

4. Introduction

The purpose of the LBT designed general purpose SERDES communications (GPSCOM) board in the instrument rotator system is to provide a high speed fiber communications link between the instrument rotator control system and the instrument rotator application board and associated hardware.

4.1. GPSCOM Board Hardware

The general purpose SERDES communications (GPSCOM) board was designed to provide a high speed communications link between the instrument rotator controller in the telescope auxiliary control room[RD1][RD3] and the instrument rotator application board [RD4] and associated hardware on the telescope elevation axis. It was also designed as a generic communications board with a reconfigurable FPGA to operate in other applications.

The heart of the communications board is the Serializer/De-serializer (SERDES) integrated circuit and the field programmable gate array (FPGA) integrated circuit. The SERDES IC's transmitter accepts 8 bit parallel data as input, serializes the data and then transmits the data in a serial high speed format. The SERDES IC's receiver accepts serial high speed data and de-serializes the data and outputs the data in 8 bit parallel format. The FPGA provides an interface to the SERDES IC and is programmed to meet the input/output signal requirements and function of the instrument rotator system. The ability to configure the FPGA I/O allows this board to be configured for applications other than the instrument rotators. Figure 1 and 2 are photos of the proto type of the general purpose communication board.

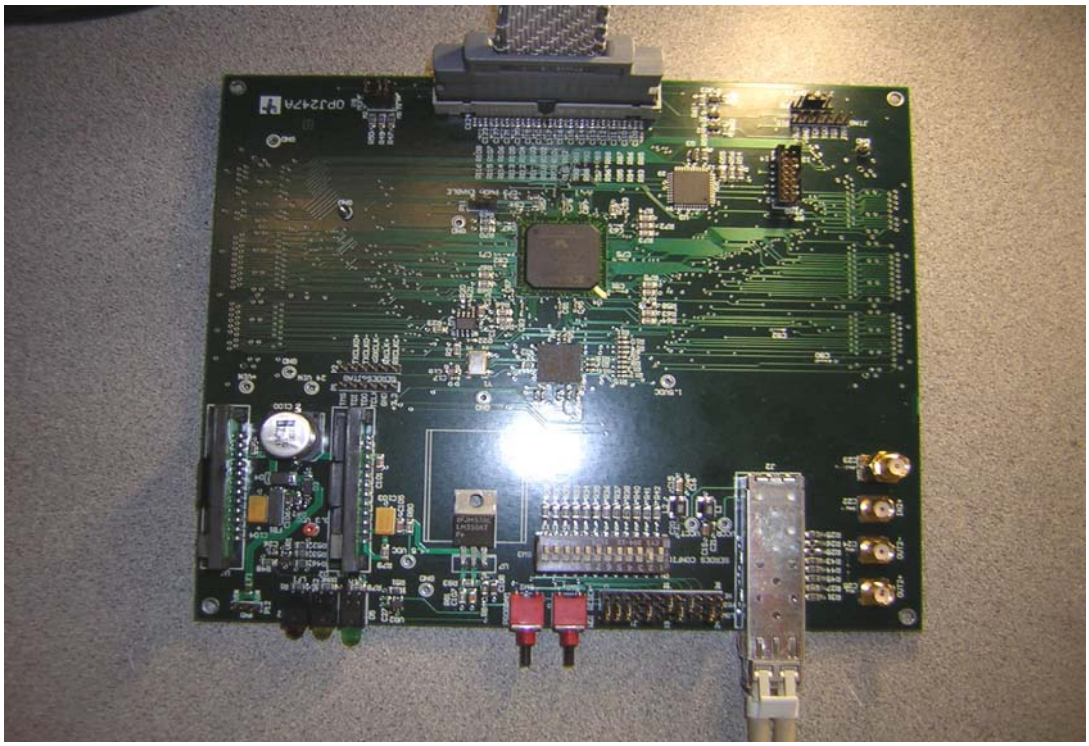

Figure 1

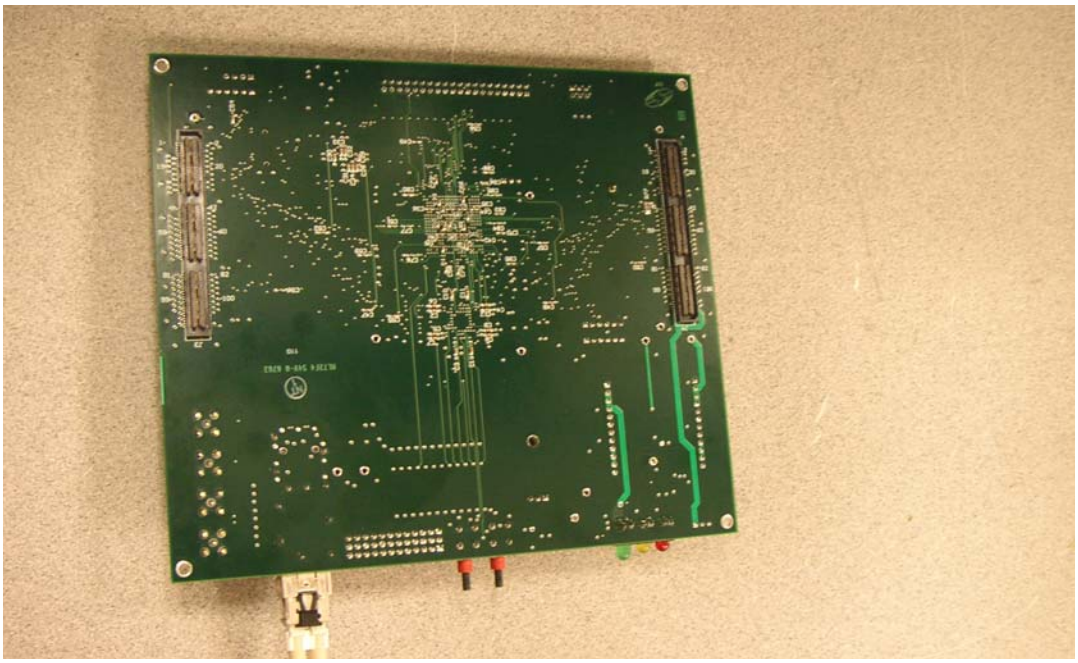

Figure 2

4.2. GPSCOM Board General Design

The GPSCOM board is an 8 layer PCB that was designed as a 50 ohm controlled impedance board [RD5]. Figure 3 is a block diagram of the GPSCOM board interface. The parallel electrical interface is through two high speed 120 pin 50 ohm connectors on the back of the board. All critical high speed signals are single ended and use 50 ohm source terminations. However, the Virtex II FPGA can also be configured for other I/O standards and terminations including differential signals. Two serial interfaces are available on the board: 1) High speed serial 62.5/125umTX/RX fiber link, 2) High speed serial 50 ohm TX/RX coaxial cable link. The maximum Cypress Hotlink II SERDES serial communications data rate is 1500MBaud and is determined by the crystal clock oscillator and SERDES settings on the board. The prototype board for the rotators operates with a 50 MHz crystal clock oscillator and 500Mb data rate. The board contains onboard regulated supplies for the FPGA/SERDES and other components. The input voltage is either supplied through one of the 120 pin application board connectors or a 3 pin header for stand alone operation/testing. The input voltage range is 12VDC to 25VDC. A 40 pin logic analyzer header with on board isolation network is available to monitor FPGA signals.

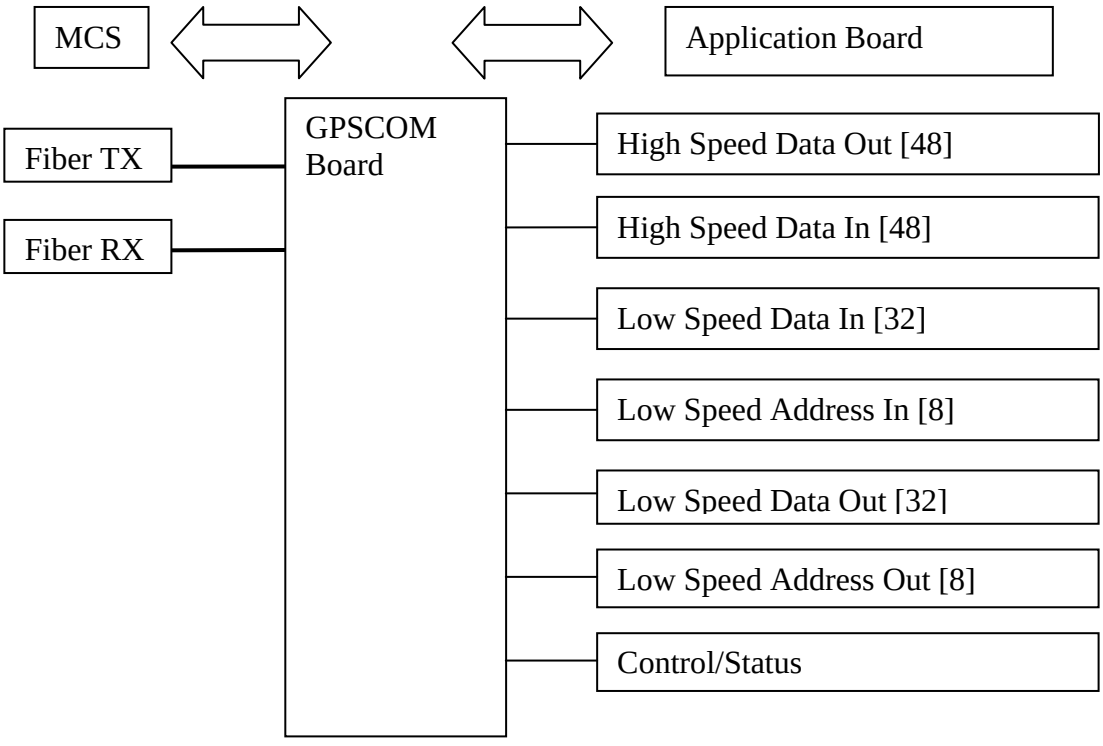

Figure 3

5. FPGA Hardware and HDL code

The FPGA is a SRAM based Xilinx Virtex II 500,000 logic gate device in a 456 pin ball grid array (BGA) package [RD7]. Configuration of the FPGA is done via a JTAG interface. The JTAG chain can be configured to support the local FPGA and Xilinx XC18V04 configuration PROM or can also be looped through the application board to also support any JTAG devices residing on the application board. The FPGA synthesizable code for the rotator is a synchronous design written in VHDL. The final configuration code is stored in a Xilinx XC18V04 configuration PROM on the board. The board is automatically configured on power-up or by depressing the program switch located on the board.

The FPGA VHDL [RD6] design includes 7 major code blocks: transmitter, receiver, CRC generator, transmit sequencer, receive sequencer, synchronizer, and control/status/loop-back registers.

The transmitter block builds transmit data frames and sends them to the SERDES through an 8 bit parallel interface to be transmitted in a serial bit stream by the SERDES. See [RD8] for a more detailed explanation of the communication protocol used on the GPSCOM board for the instrument rotators. The transmit block also appends a CRC to the end of the transmit data frame so the integrity of the transmitted data can be checked at the receive end.

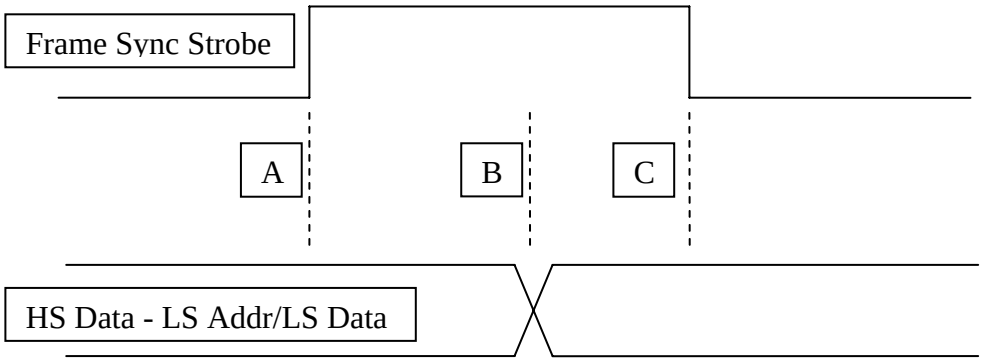

Figure 4

Figure 4 is a timing diagram for the transfer of data to the transmitter. The frame sync strobe edges are synchronous with the system clock. Data to be transmitted is requested from the transmitter on the rising edge of the frame sync pulse (A) at which time the low speed address and low speed data ports can be loaded (B). High speed asynchronous data from the application board is send through a 2 stage data synchronizer and then presented

to the transmitter high speed port. All data is latched into the transmitter on the falling edge of the frame sync pulse (C). The frame sync pulse width is adjustable in the transmitter and has a default setting of 5 clock cycles. Synchronous data should available at least 1 clock cycle prior to the falling edge of frame sync for the data to be transmitted during that transmit frame. With a system clock of 50 MHz (20 ns) this allows 4 clock cycles (80 ns) for new data to be presented to the transmitter after the rising edge of frame sync.

The transmit sequencer block requests low speed bus data from the application board and sends the address and data to the transmit block using hand shake signaling. The amount of low speed data requested from the application board and transmitted is determined by the contents of the control register and can be remotely configured by the local control side of the communications link. Figure 5 is a timing diagram for the low speed read cycle. The address is placed on the bus and then signaled ready by the falling edge of the read enable strobe. The application board decodes the address and places data for that address on the bus. The data is then latched in the transmit sequencer on the rising edge of the strobe. The read strobe width is adjustable with a default width of two clock cycles.

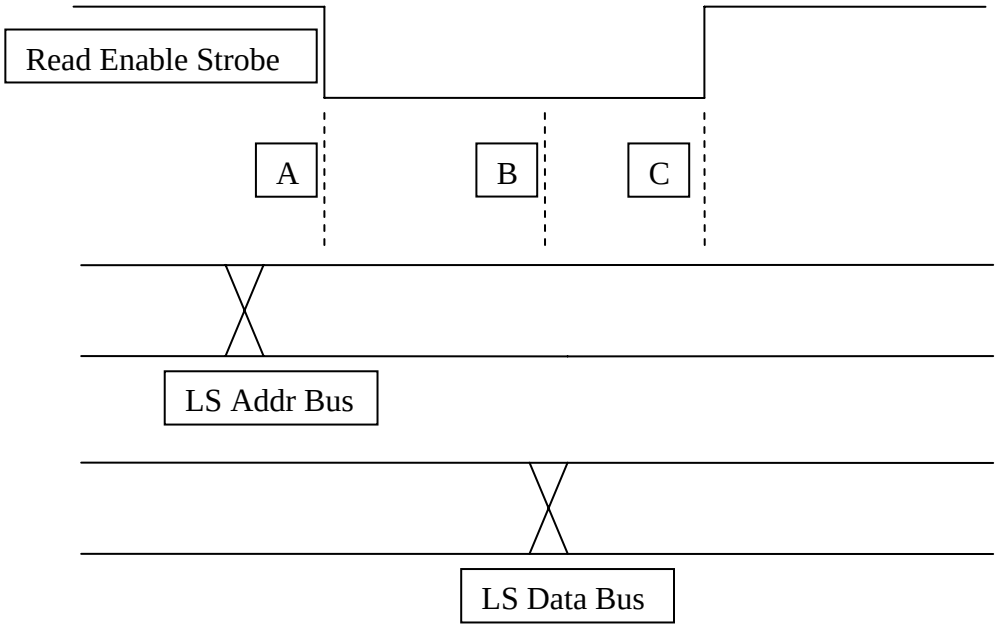

Figure 5

The receiver block receives parallel 8 bit data from the SERDES and decodes the received data frames and sends them to the receive sequencer after successfully passing a CRC check of the received data. Any data or format errors detected by the receiver are passed back to the originated side through a status register that is transmitted to the originating side during each transmit data frame. Error signals are also passed along to the receive sequencer block. Figure 6 is a timing diagram for the transfer of data from the receiver to the receiver sequence. Received data that successfully passes a CRC check is place on the low speed bus along with its address and then signaled ready on the rising edge of the frame ready strobe. The sequencer should latch the address and data on the falling edge of the strobe.

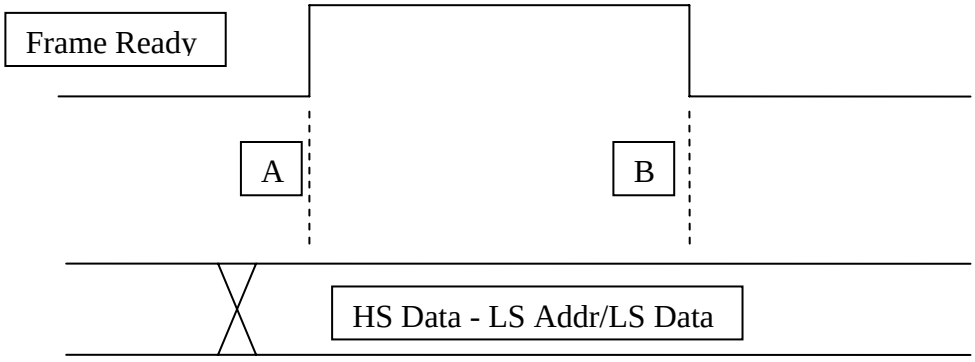

Figure 6

The receive sequencer block accepts a data structure from the receiver block and sends the high and low speed data to the application board. The receive sequencer also sends any low speed data addressed to the control or loop back register to the register block. Figure 7 is a timing diagram of the low speed bus write cycle. The address and data are placed on the bus and signaled ready on the falling edge of the write enable strobe. The address and data should be latched in the application board on the rising edge of the strobe. The write strobe width is adjustable with a default width of two clock cycles.

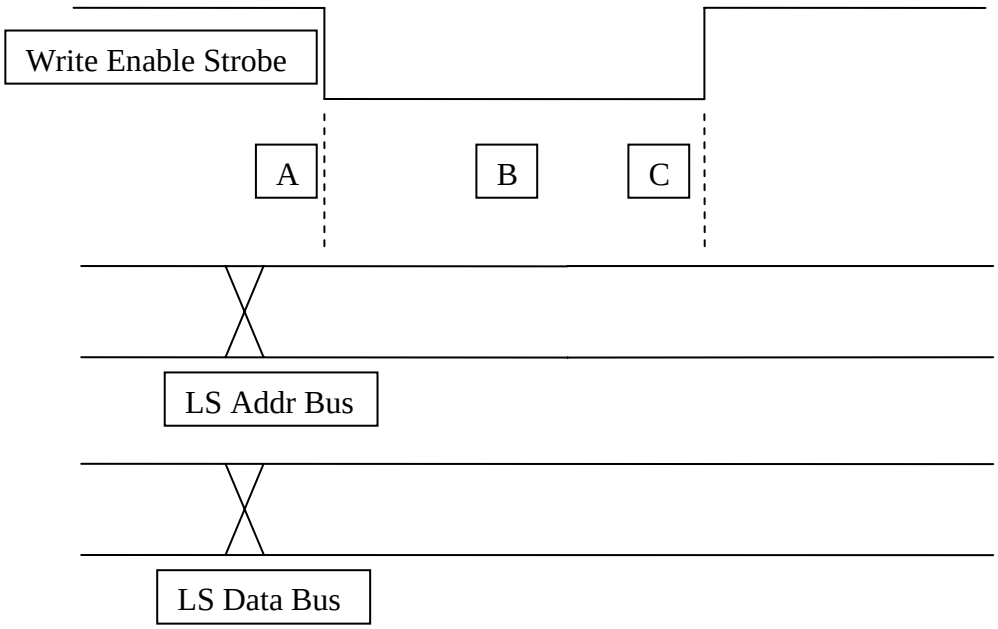

Figure 7

A CRC block allows an 8 bit parallel CRC to be generated for use by the transmitter and receiver blocks to ensure the validity of any data sent across the high speed serial communication link.

The synchronizer block synchronizes the asynchronous high speed input signals to the system clock in the FPGA.

6. SERDES

The SERDES is a single channel Hotlink II device from Cypress Semiconductor in a 100 pin BGA package [RD8]. The device allows parallel 8 bit data to be sent and received over a high speed serial bit stream transceiver. The 8 bit parallel data is encoded/decoded into/from 10 bit data characters using 8b/10b encoding and sent/received on a high speed serial data link. The 8b/10b encoding ensures an adequate number of data transition exists for the receiver PLL to recover a clock from the received data and also allows the serial data to be send down a balanced transmission line such as fiber optic cabling using a relatively small protocol overhead (2 bits for every 8 data bits). The SERDES is configured using J1 and SW1. Table 1 and table 2 list the default settings for J1 and SW1.

J1 Header Settings - Low = Gnd, High = 3.3V, Mid = Open

Signal	Jumper Setting
SDASEL	Low
SPDSEL	Low
RXCKSEL	Low
DECMODE	Mid
TXCKSEL	Low
TXMODE(1)	Mid
TXMODE(0)	High
RXMODE	Low
FRAMCHAR	High
RFMODE	Mid
PARCTL	Low

Table 1

SW1 Switch Settings - On = 3.3V, Off = Gnd

Switch	Signal	Switch Setting
1	LPEN	Off
2	RFEN	On
3	RXRATE	Off
4	TXCLK	Off
5	TXRATE	Off
6	NC	Off
7	INSEL	Off
8	BISTLE	Off
9	OELE	Off
10	RXLE	Off
11	BOE(0)	Off
12	BOE(1)	Off

Table 2

7. Fiber Transceiver

The fiber transceiver is a small form-factor pluggable (SFP) 1.25 GBd fiber transceiver from Agilent [RD9]. The device conforms to the SFP standard that allows the use of multiple protocol fiber transceivers from multiple vendors. The Agilent transceiver targets the 1.25 GBd Ethernet and 1.0625 GBd Fibre channel applications, but per their application engineers also functions across a wide range of input data rates. The fiber transceiver on the prototype GPCOM board runs at 500 Mb using a LBT designed data

protocol. See [RD2] for a more detailed description of the data protocol and typical application data rates.

8. Voltage Supervisors and Watchdog Timer

The GPSCOM board contains two voltage supervisor circuits that monitor the 1.5V and 3.3V on board power supplies. The supervisors hold the FPGA in reset until the two supplies reach stable voltage levels by holding the prog_b pin low on the FPGA. Once the supplies are stable they release the prog_b program pin and allow the FPGA to continue the configuration process. If either supply drops below a predefined level the prog_b pin will be brought low until the supply again reaches the proper level and becomes stable. This will force the FPGA to be reconfigured from the on board PROM.

The GPSCOM board also contains a watchdog timer IC. The watchdog input is driven from an FPGA output and expects to see a rising or falling edge on this input at least once every 0.8 seconds to retrigger the watchdog. If not, the watchdog reset/ output will be asserted for approximately 0.180 seconds and again expects the input to be toggle or the cycle is again repeated. The watchdog output does not reset any logic on the GPSCOM board, but is passed to the application board. The output is also available on a FPGA input should the GPSCOM board want to directly monitor this signal. Generally the watchdog output should be looped through all critical components on the application board then be brought back into the FPGA so this signal can be transmitted back on the serial communication link to form a round trip heart beat signal. The heart beat signal origination is the DSP at the local end and is sent down the serial communications link to drive the watchdog input on the GPSCOM board (see figure 8). This ensures that all critical components attached to the serial communications link are “alive” and responding.

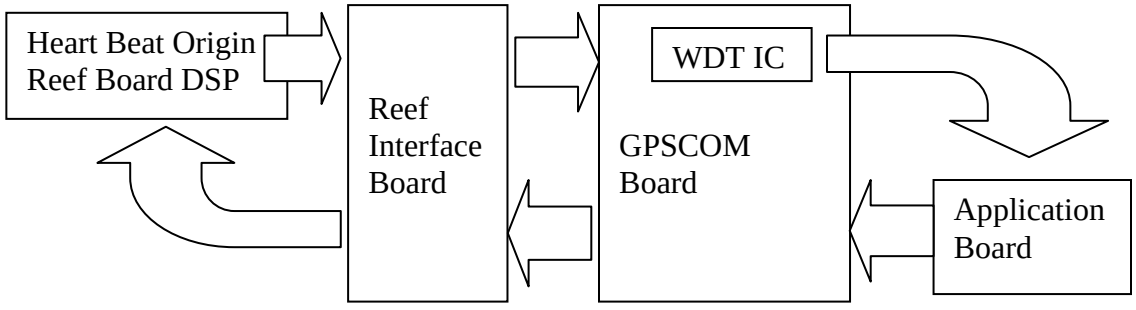

Figure 8

9. Application Board Interface

Table 3 and table 4 below lists the signal and pin assignments on the two 120 pin high speed connectors which form the electrical interface to the application board. It is worth noting that since the majority of signals connect to the programmable FPGA I/O on the GPSCOM board and this interface can be easily modified if the need arises. Care should be taken when assigning the FPGA I/O to ensure the recommended simultaneous switched outputs are not exceeded for each FPGA I/O bank.

J3 Connector

Pin	Pin Name	Description
1	NC	No Connect
2	HS_IN(8)	High Speed Bus Input
3	NC	No Connect
4	HS_OUT(32)	High Speed Bus Output
5	ASYNC_RESET_OUT	Async Reset Output
6	WDI/	Watchdog Timer Output
7	NC	No Connect
8	WR_EN	Low Speed Bus Write Strobe
9	NC	No Connect
10	HS_OUT(36)	High Speed Bus Output
11	SPARE_IN(0)	Spare I/O (to analyzer header also)
12	SPARE_IN(1)	Spare I/O (to analyzer header also)
13	SPARE_IN(2)	Spare I/O (to analyzer header also)
14	SPARE_IN(3)	Spare I/O (to analyzer header also)
15	SPARE_IN(4)	Spare I/O (to analyzer header also)
16	SPARE_IN(5)	Spare I/O (to analyzer header also)
17	SPARE_IN(6)	Spare I/O (to analyzer header also)
18	SPARE_IN(7)	Spare I/O (to analyzer header also)
19	SPARE_IN(8)	Spare I/O (to analyzer header also)
20	SPARE_IN(9)	Spare I/O (to analyzer header also)
21	SPARE_IN(10)	Spare I/O (to analyzer header also)
22	SPARE_IN(11)	Spare I/O (to analyzer header also)
23	SPARE_IN(12)	Spare I/O (to analyzer header also)
24	SPARE_IN(13)	Spare I/O (to analyzer header also)
25	SPARE_IN(14)	Spare I/O (to analyzer header also)
26	SPARE_IN(15)	Spare I/O (to analyzer header also)
27	NC	No Connect
28	NC	No Connect
29	HS_IN(37)	High Speed Bus Input

30	HS_IN(47)	High Speed Bus Input
31	HS_IN(45)	High Speed Bus Input
32	HS_IN(46)	High Speed Bus Input
33	TMS	JTAG TMS
34	TCK	JTAG TCK
35	TDI_APP	JTAG TDI
36	TDO_APP	JTAG TDO
37	HS_IN(43)	High Speed Bus Input
38	HS_IN(44)	High Speed Bus Input
39	HS_IN(41)	High Speed Bus Input
40	NC	No Connect
41	HS_IN(39)	High Speed Bus Input
42	HS_IN(40)	High Speed Bus Input
43	NC	No Connect
44	HS_IN(38)	High Speed Bus Input
45	NC	No Connect
46	HS_IN(36)	High Speed Bus Input
47	IN_ADDR(0)	Low Speed Input Bus Address
48	IN_ADDR(1)	Low Speed Input Bus Address
49	IN_ADDR(2)	Low Speed Input Bus Address
50	IN_ADDR(3)	Low Speed Input Bus Address
51	IN_ADDR(4)	Low Speed Input Bus Address
52	IN_ADDR(5)	Low Speed Input Bus Address
53	IN_ADDR(6)	Low Speed Input Bus Address
54	IN_ADDR(7)	Low Speed Input Bus Address
55	IN_DATA(0)	Low Speed Input Bus Data
56	IN_DATA(1)	Low Speed Input Bus Data
57	IN_DATA(2)	Low Speed Input Bus Data
58	IN_DATA(3)	Low Speed Input Bus Data
59	IN_DATA(4)	Low Speed Input Bus Data
60	IN_DATA(5)	Low Speed Input Bus Data
61	IN_DATA(6)	Low Speed Input Bus Data
62	IN_DATA(7)	Low Speed Input Bus Data
63	OUT_DATA(8)	Low Speed Output Bus Data
64	OUT_DATA(9)	Low Speed Output Bus Data
65	OUT_DATA(10)	Low Speed Output Bus Data
66	OUT_DATA(11)	Low Speed Output Bus Data
67	OUT_DATA(12)	Low Speed Output Bus Data
68	OUT_DATA(13)	Low Speed Output Bus Data
69	OUT_DATA(14)	Low Speed Output Bus Data
70	OUT_DATA(15)	Low Speed Output Bus Data
71	OUT_DATA(16)	Low Speed Output Bus Data
72	OUT_DATA(17)	Low Speed Output Bus Data

73	OUT_DATA(18)	Low Speed Output Bus Data
74	OUT_DATA(19)	Low Speed Output Bus Data
75	OUT_DATA(20)	Low Speed Output Bus Data
76	OUT_DATA(21)	Low Speed Output Bus Data
77	OUT_DATA(22)	Low Speed Output Bus Data
78	OUT_DATA(23)	Low Speed Output Bus Data
79	OUT_DATA(24)	Low Speed Output Bus Data
80	OUT_DATA(25)	Low Speed Output Bus Data
81	OUT_DATA(26)	Low Speed Output Bus Data
82	OUT_DATA(27)	Low Speed Output Bus Data
83	OUT_DATA(28)	Low Speed Output Bus Data
84	OUT_DATA(29)	Low Speed Output Bus Data
85	OUT_DATA(30)	Low Speed Output Bus Data
86	OUT_DATA(31)	Low Speed Output Bus Data
87	HS_IN(33)	High Speed Bus Input
88	HS_IN34)	High Speed Bus Input
89	HS_OUT(0)	High Speed Bus Output
90	HS_OUT(1)	High Speed Bus Output
91	HS_OUT(2)	High Speed Bus Output
92	HS_OUT(3)	High Speed Bus Output
93	HS_OUT(4)	High Speed Bus Output
94	HS_OUT(5)	High Speed Bus Output
95	HS_OUT(6)	High Speed Bus Output
96	HS_OUT(7)	High Speed Bus Output
97	CLK2_OUT	Clock 2 Output
98	HS_IN(9)	High Speed Bus Input
99	HS_IN(10)	High Speed Bus Input
100	HS_IN(11)	High Speed Bus Input
101	HS_IN(12)	High Speed Bus Input
102	HS_IN(13)	High Speed Bus Input
103	HS_IN(14)	High Speed Bus Input
104	HS_IN(15)	High Speed Bus Input
105	HS_IN(16)	High Speed Bus Input
106	HS_IN(17)	High Speed Bus Input
107	HS_IN(18)	High Speed Bus Input
108	HS_IN(19)	High Speed Bus Input
109	HS_IN(20)	High Speed Bus Input
110	HS_IN(21)	High Speed Bus Input
111	NC	No Connect
112	NC	No Connect
113	HS_IN(22)	High Speed Bus Input
114	NC	No Connect
115	NC	No Connect

116	NC	No Connect
117	HS_OUT28)	High Speed Bus Output
118	HS_IN(32)	High Speed Bus Input
119	HS_OUT(30)	High Speed Bus Output
120	HS_OUT(31)	High Speed Bus Output

Table 3

J4 Connector

Pin	Pin Name	Description
1	OUT_DATA(3)	Low Speed Output Bus Data
2	HS_OUT(46)	High Speed Bus Output
3	HS_OUT(43)	High Speed Bus Output
4	HS_OUT(45)	High Speed Bus Output
5	HS_OUT(41)	High Speed Bus Output
6	HS_OUT(42)	High Speed Bus Output
7	NC	No Connect
8	HS_OUT(40)	High Speed Bus Output
9	HS_OUT(38)	High Speed Bus Output
10	HS_OUT(39)	High Speed Bus Output
11	IN_DATA(24)	Low Speed Input Bus Data
12	IN_DATA(25)	Low Speed Input Bus Data
13	NC	No Connect
14	NC	No Connect
15	HS_IN(28)	High Speed Bus Input
16	NC	No Connect
17	NC	No Connect
18	HS_OUT(26)	High Speed Bus Output
19	HS_IN(42)	High Speed Bus Input
20	HS_OUT(25)	High Speed Bus Output
21	HS_OUT(27)	High Speed Bus Output
22	HS_IN(35)	High Speed Bus Input
23	HS_OUT(24)	High Speed Bus Output
24	NC	No Connect
25	HS_OUT(44)	High Speed Bus Output
26	NC	No Connect
27	NC	No Connect
28	HS_OUT(37)	High Speed Bus Output
29	NC	No Connect

30	NC	No Connect
31	NC	No Connect
32	NC	No Connect
33	CLK1_OUT	Clock 1 Output
34	HS_OUT(23)	High Speed Bus Output
35	HS_IN(27)	High Speed Bus Input
36	HS_OUT(22)	High Speed Bus Output
37	IN_DATA(26)	Low Speed Input Bus Data
38	IN_DATA(27)	Low Speed Input Bus Data
39	IN_DATA(28)	Low Speed Input Bus Data
40	IN_DATA(29)	Low Speed Input Bus Data
41	FAULT	APP Board Fault Input
42	RD_EN	Low Speed Bus Read Strobe Output
43	ASYNC_RESET_IN	Async reset input
44	COM_GOOD	Communications good output
45	IN_DATA(30)	Low Speed Input Bus Data
46	IN_DATA(31)	Low Speed Input Bus Data
47	HS_IN(0)	High Speed Bus Input
48	HS_IN(1)	High Speed Bus Input
49	HS_IN(2)	High Speed Bus Input
50	HS_IN(3)	High Speed Bus Input
51	HS_IN(4)	High Speed Bus Input
52	HS_IN(5)	High Speed Bus Input
53	HS_IN(6)	High Speed Bus Input
54	HS_IN(7)	High Speed Bus Input
55	HS_OUT(8)	High Speed Bus Output
56	HS_OUT(9)	High Speed Bus Output
57	HS_OUT(10)	High Speed Bus Output
58	HS_OUT(11)	High Speed Bus Output
59	HS_OUT(12)	High Speed Bus Output
60	HS_OUT(13)	High Speed Bus Output
61	HS_OUT(14)	High Speed Bus Output
62	HS_OUT(15)	High Speed Bus Output
63	HS_OUT(16)	High Speed Bus Output
64	HS_OUT(17)	High Speed Bus Output
65	HS_OUT(18)	High Speed Bus Output
66	HS_OUT(19)	High Speed Bus Output
67	IN_DATA(18)	Low Speed Input Bus Data
68	IN_DATA(19)	Low Speed Input Bus Data
69	HS_OUT(20)	High Speed Bus Output
70	HS_OUT(21)	High Speed Bus Output
71	HS_OUT(33)	High Speed Bus Output
72	HS_OUT(29)	High Speed Bus Output

73	OUT_ADDR(0)	Low Speed Output Bus Address
74	OUT_ADDR(1)	Low Speed Output Bus Address
75	OUT_ADDR(2)	Low Speed Output Bus Address
76	OUT_ADDR(3)	Low Speed Output Bus Address
77	OUT_ADDR(4)	Low Speed Output Bus Address
78	OUT_ADDR(5)	Low Speed Output Bus Address
79	HS_IN(24)	High Speed Bus Input
80	HS_IN(25)	High Speed Bus Input
81	OUT_ADDR(6)	Low Speed Output Bus Address
82	OUT_ADDR(7)	Low Speed Output Bus Address
83	OUT_DATA(0)	Low Speed Output Bus Data
84	OUT_DATA(1)	Low Speed Output Bus Data
85	OUT_DATA(2)	Low Speed Output Bus Data
86	HS_OUT(47)	High Speed Bus Output
87	OUT_DATA(4)	Low Speed Output Bus Data
88	OUT_DATA(5)	Low Speed Output Bus Data
89	OUT_DATA(6)	Low Speed Output Bus Data
90	OUT_DATA(7)	Low Speed Output Bus Data
91	IN_DATA(8)	Low Speed Input Bus Data
92	IN_DATA(9)	Low Speed Input Bus Data
93	IN_DATA(10)	Low Speed Input Bus Data
94	IN_DATA(11)	Low Speed Input Bus Data
95	IN_DATA(12)	Low Speed Input Bus Data
96	IN_DATA(13)	Low Speed Input Bus Data
97	IN_DATA(14)	Low Speed Input Bus Data
98	IN_DATA(15)	Low Speed Input Bus Data
99	IN_DATA(16)	Low Speed Input Bus Data
100	IN_DATA(17)	Low Speed Input Bus Data
101	IN_DATA(20)	Low Speed Input Bus Data
102	IN_DATA(21)	Low Speed Input Bus Data
103	HS_OUT(35)	High Speed Bus Output
104	HS_OUT(34)	High Speed Bus Output
105	HS_IN(26)	High Speed Bus Input
106	NC	No Connect
107	NC	No Connect
108	HS_IN(29)	High Speed Bus Input
109	HS_IN(30)	High Speed Bus Input
110	HS_IN(31)	High Speed Bus Input
111	IN_DATA(22)	Low Speed Input Bus Data
112	IN_DATA(23)	Low Speed Input Bus Data
113	NC	No Connect
114	HS_IN(23)	High Speed Bus Input
115	+24VDC	+24VDC Board Power Input

116	+24VDC	+24VDC Board Power Input
117	+24VDC	+24VDC Board Power Input
118	+24VDC	+24VDC Board Power Input
119	+24VDC	+24VDC Board Power Input
120	+24VDC	+24VDC Board Power Input

Table 4

Doc_info_start
Title: General Purpose SERDES Communications PC Board Description and Operations Manual
Document Type: *Specification / Technical Report / Technical Manual*
Source: Steward Observatory
Issued by: *Michael Gusick*
Date_of_Issue: *9 March 2007*
Revised by:
Date_of_Revision:
Checked by:
Date_of_Check:
Accepted by:
Date_of_Acceptance:
Released by:
Date_of_Release:
File Type: MS Word
Local Name:
Category: 675
Sub-Category: *002*
Assembly: *Instrument Rotator and Cable Chain*
Sub-Assembly:
Part Name:
CAN 675s002
Revision: *a*
Doc_info_end