HI-6300 IP Core and Zynq Ultrascale+ MPSoC Demonstration Guide

May 2019

REVISION HISTORY

Revision	Date	Description of Change
AN-6300, Rev. Prelim.	05-09-19	Preliminary Release.
Rev. New	05-31-19	Initial Release. Add updated schematic and BOM.

Purpose

This document serves to aid the developer in integrating the Holt HI-6300 IP Core into a Vivado design. The included software utilizes the Holt API library to demonstrate accessing the Bus Controller (BC), Remote Terminal (RT), and Monitor (MT) features of the IP. The development board used for this demonstration is the Xilinx Zynq UltraScale+ MPSoC ZCU102 Evaluation Kit. An embedded ARM Cortex-A53 interfaces with the HI-6300 IP Core via an AXI4 bus and executes the demonstration software. The demo project is a bare-metal implementation using the Holt API software library. This software is accessible by a terminal console via a USB/UART connector.

Block Diagram

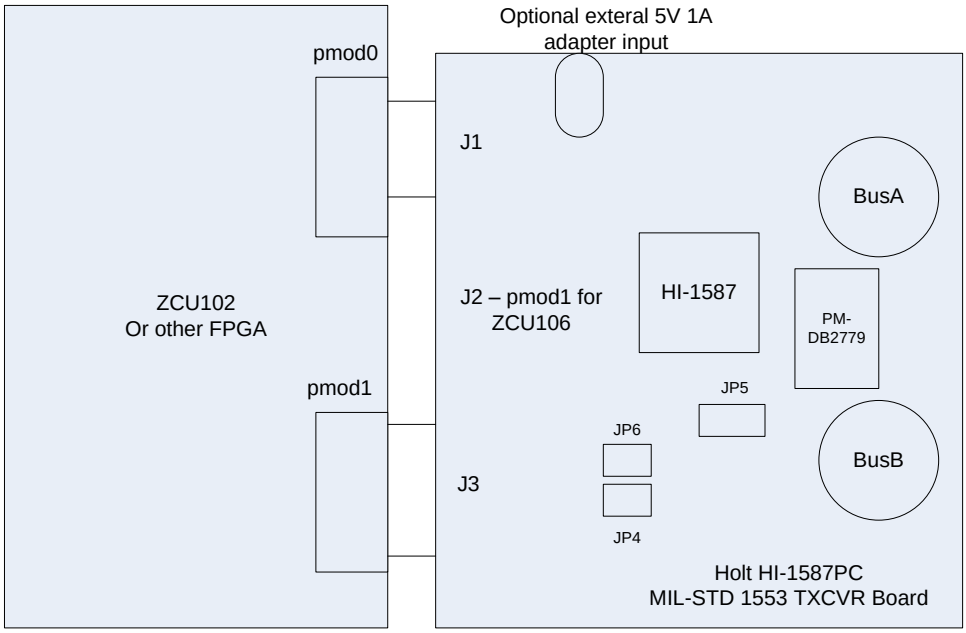


Figure 1. Development Board Block Diagram.

Opening the Vivado Project

This project was built using version 2018.3 of Vivado and can be accessed by unzipping the included archive folder and opening the .xpr file located in the root directory. The top most block diagram can be found by expanding the “Base_Zynq_MPSoc_wrapper” under the design sources folder in the source file explorer window.

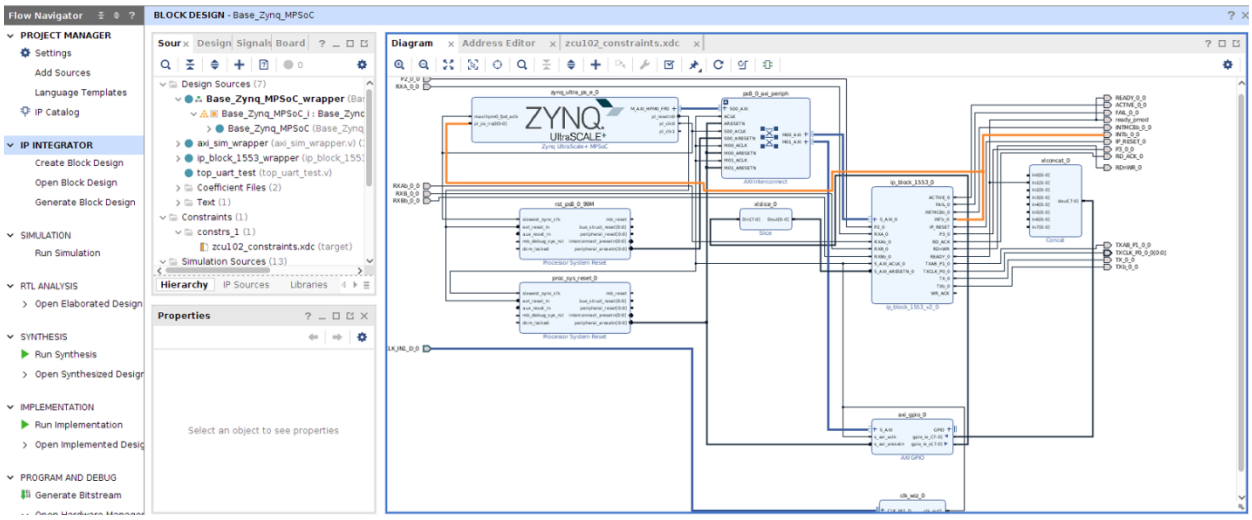


Figure 2 Main block design

Base_Zynq_MPSoC Block Design Overview

This section lists the various IP used in the block design and their purpose.

Zynq UltraScale+ MPSoC	This instantiates the Zynq processing system. The processing system (PS) provides a UART for terminal communication to the demo application. The HI-6300 IP resides in the programmable logic (PL) section and is accessed via an AXI4 bus. A single interrupt line is connected to the HI-6300 IP.
Processor System Reset	Provides asynchronous reset to the AXI interconnect block
AXI Interconnect	Provides access to AXI GPIO IP and 1553 IP Block via the Zynq AXI HPM0-FPD interface.
IP_Block_1553	Contains the HI-6300 IP as well as an AXI4-Lite to IPIF bridge that can drive up to eight instances of the IP. In this example one of the IPIF channels is used.
AXI GPIO	Used to drive the nRESET port on HI-6300 and monitor the status of the nREADY port.
Slice IP	Extracts nRESET port from the AXI GPIO IP and routes it to the HI-6300.
Concat IP	Routes the nREADY port from the HI-6300 into the input of the AXI GPIO IP.
Clock Wizard IP	Utilize onboard 300 MHz clock to create 100 MHz clock to drive HI-6300 and AXI4-lite to IPIF bridge.

IP_Block_1553 Block Design Overview

This section lists the IP used in the 1553 block design. The 1553 IP Block can be accessed by expanding the "ip_block_1553_wrapper" under the design sources in the source file explorer window.

AXI_LITE_IPIF_V1_0	Converts AXI4-Lite bus to IPIF for interfacing with the IP. Supports 8 channels.
Holt IP Glue	Provides glue logic to allow multiple HI-6300 IP to interface with AXI4-Lite to IPIF bridge. This block handles shifting the AXI address to the right by two bits before it is presented to the IP. This is to reverse a left shift by two in the driver software to maintain alignment of the AXI address to a 4-byte alignment.
IP6300_wrapper	Instantiates encrypted HI-6300 IP
Block Memory Generator	Provides block RAM for HI-6300 IP. Default memory contents set to 0x0000.
Utility Vector Logic	Inverts INTb output of HI-6300 IP to allow Zynq MPSoC to detect rising edge of interrupt.

FPGA resource utilization:

The design uses a 100MHz clock source, the example uses the on-board 300MHz clock and divides it down to 100MHz.

Component	LUT	FF	BRAM
HI-6300 IP	9650	5223	0
Glue logic, RAM	188	71	40

Each of the IP in this block design except for the IP6300_Wrapper may be modified by right-clicking on them in the block diagram window and selecting "Edit in IP Packager" Constants are used to drive inputs on the HI-6300 to set RT Address bits, Enables, etc. **The default RT address is 3.**

Programming the Bit File

This section covers programming the FPGA with the bit file generated by Vivado.

In the Flow Navigator, select “Generate Bitstream” under the PROGRAM AND DEBUG menu. This process takes several minutes.

Ensure the USB-JTAG connector is connected to the host PC via a USB to mini-USB cable. Select “Open Target” under the PROGRAM AND DEBUG section of the Flow Navigator in Vivado. Select “Auto Connect” and “Program Device.” The following table lists the user LEDs and their purpose once the bit file has been programmed

USER LED	Name	Description
GPIO_LED DS38	Active	illuminates while BC is transmitting 1553 traffic
GPIO_LED DS37	Ready	illuminates when 1587 Authentication process succeeds
GPIO_LED DS39	Fail	illuminates when 1587 Authentication fails
GPIO_LED DS40	INTb	illuminates while interrupt line is inactive
GPIO_LED DS41	MC8INTb	illuminates while MC8 interrupt line is inactive

Initial Configuration of the Xilinx SDK Workspace

This section covers how to import the example projects necessary to build the demo application for the embedded ARM core. In the included archive project this step is already completed but the instructions are included to recover projects that are accidently removed.

Select “File->Launch SDK” from the Vivado menu to open Xilinx SDK. The tool will open a workspace that is local to the main project directory.

Select “File->Import” from the Xilinx SDK window and select “Existing Projects into Workspace” under the General folder.

Select “Browse” for the root directory and navigate to the “zynq_arm_uart_test_project.sdk” folder and select the “Base Zynq_MPSoC_wrapper_hw_platform_0” folder and click OK.

Perform this operation for “uart_example_project_bsp” and “uart_example_project”. The imported projects contain the hardware definition for the FPGA project. This drives the creation of the board support package project. The board support package allows the developer to utilize the UART, drive GPIOs and access the AXI-4 bus.

Note: When opening the project workspace for the first time, a duplicate Base_Zynq_MPSoC_Wrapper_hw_platform_1 will appear in the project explorer. This can be safely removed.

Downloading the Demo Application

This section covers using Xilinx SDK to download the demo application to the embedded ARM A53 core.

Ensure a mini-USB cable connects the USB-to-JTAG bridge connector on the dev board to the host PC.

Select “File->Launch SDK” from the Vivado menu to open Xilinx SDK in a workspace that resides locally in the Vivado project workspace.

Right-click on the board support package project in the project explorer window and select “Build Project.” Perform the same operation for the UART example project.

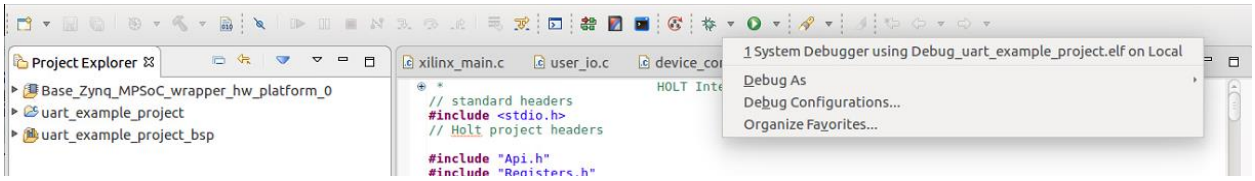
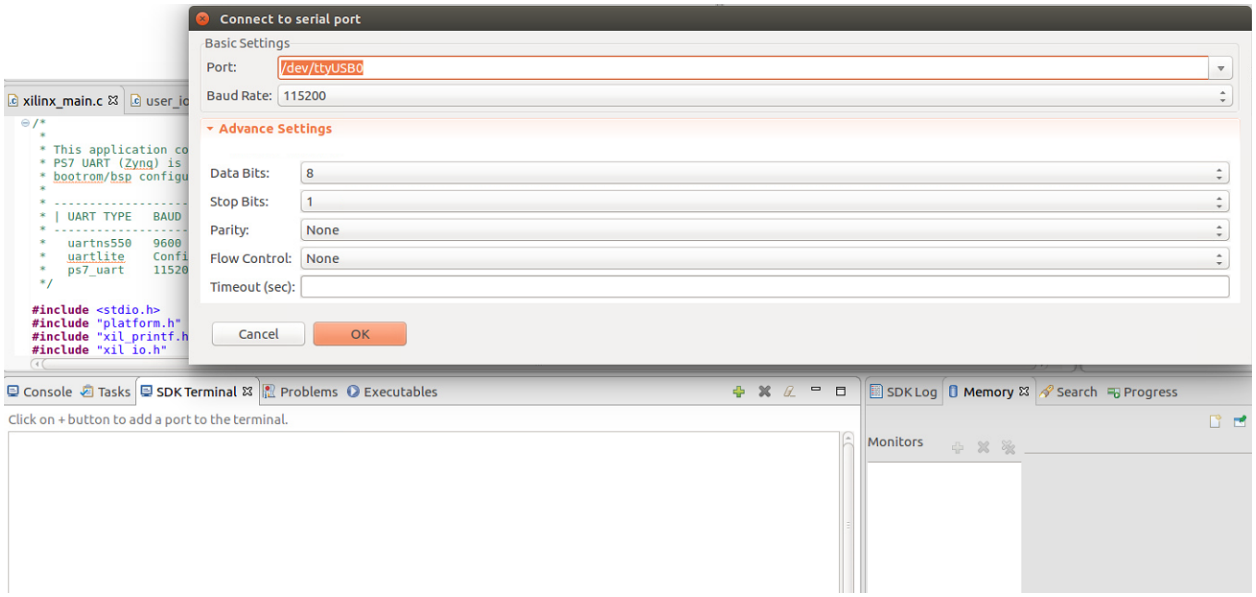


Figure 3 Debugging Demo Application on Target

From the debug menu, select the “System Debugger using Debug_uart_example_project.elf on Local.”



With the “SDK Terminal” window open click the addition symbol to open a serial connection to the embedded processor through UART0.

Note: Linux users must ensure that they have read/write access to the port under the “/dev” directory used to communicate with the board.

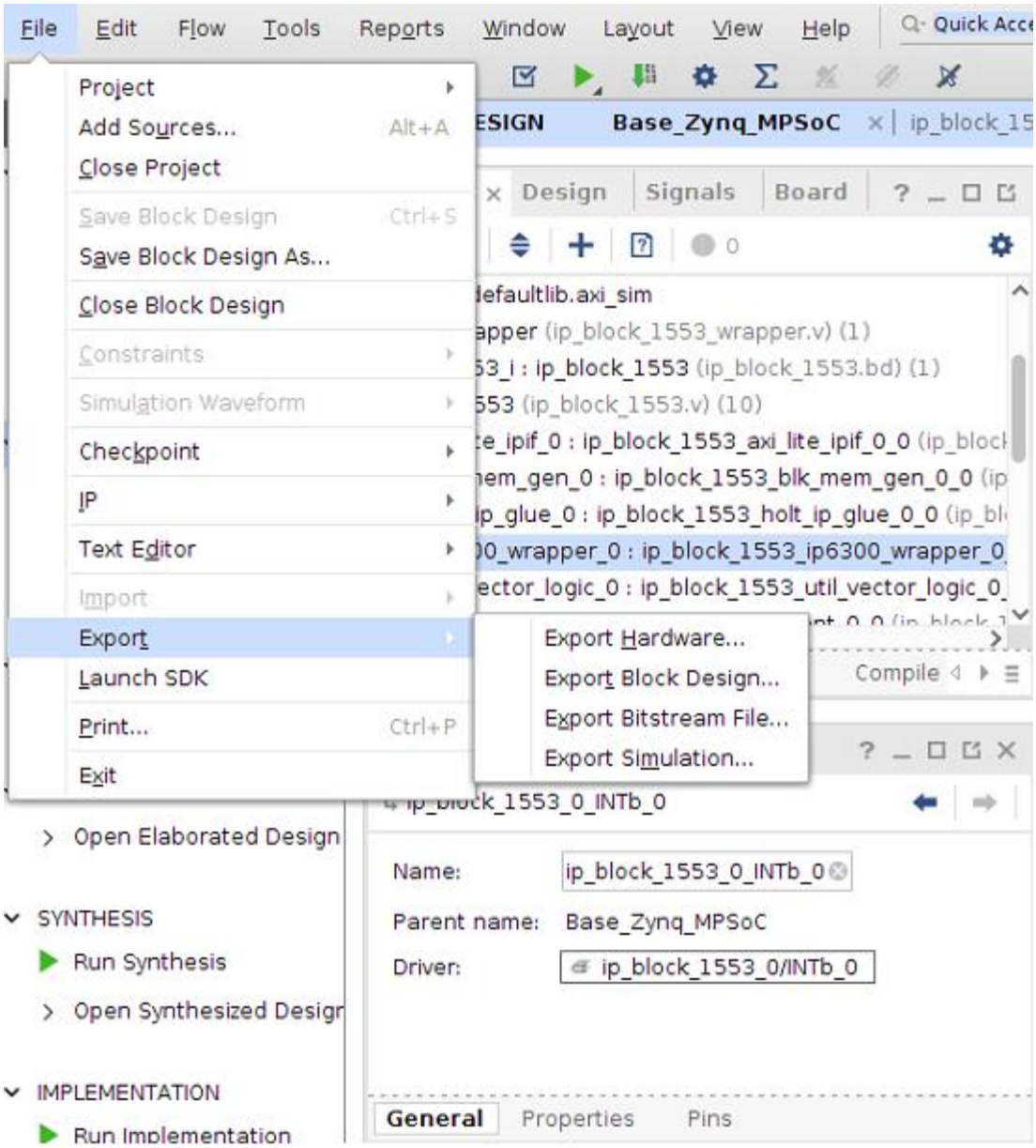
The application can be stepped through and any output will be printed to the SDK Terminal window.

Holt API Demo

For information regarding the demo application available via the UART0 console, see AN-6138API application notes.

Exporting Hardware Definition to Xilinx SDK

This section covers exporting the hardware definition for the FPGA to Xilinx SDK. This must be done whenever there is a change to the block design that would affect the memory space accessible by the Zynq MPSoC. The exported information is used by Xilinx SDK to create the board support package.



From the File Menu, select “Export->Export Hardware” From Vivado, launch the Xilinx SDK from the File Menu. The BSP can be regenerated and the demo application will need to be recompiled.

Constraint File Migration for Xilinx ZCU106 Development Board

This section lists the constraints that require modification to migrate the demo project from the ZCU102 board to the ZCU106 board. Package pin values were obtained from the ZCU106 user guide. The I/O pins listed here include the clock source for the IP, debug LEDs, and the authentication interface with the 1587 daughter card. See Holt HI-1587PC TXCVR schematic for PMOD connections to these signals.

Port Name	ZCU102 Package Pin	ZCU102 I/O Standard	ZCU106 Package Pin	ZCU106 I/O Standard
CLK_IN1_D_0_clk_p	AL8	DIFF_SSTL12	AH12	DIFF_SSTL12
CLK_IN1_D_0_clk_n	AL7	DIFF_SSTL12	AJ12	DIFF_SSTL12
ACTIVE_0_0	AG14	LVCMOS33	AL11	LVCMOS12
READY_0_0	AF13	LVCMOS33	AL13	LVCMOS12
FAIL_0_0	AE13	LVCMOS33	AK13	LVCMOS12
INTb_0_0	AJ14	LVCMOS33	AE15	LVCMOS12
INTMC8b_0_0	AJ15	LVCMOS33	AM8	LVCMOS12
RXA_0_0	A20	LVCMOS33	B23	LVCMOS18
RXAb_0_0	B20	LVCMOS33	A23	LVCMOS18
RXB_0_0	A22	LVCMOS33	F25	LVCMOS18
RXBb_0_0	A21	LVCMOS33	E20	LVCMOS18
TX_0_0	B21	LVCMOS33	K24	LVCMOS18
TXb_0_0	C21	LVCMOS33	L23	LVCMOS18
IP_RESET_0	D20	LVCMOS33	AN8	LVCMOS18
ready_pmod	E20	LVCMOS33	AN9	LVCMOS18
GPIO2	D22	LVCMOS33	AP11	LVCMOS18
int_debug_out	E22	LVCMOS33	AN11	LVCMOS18
TXCLK_P0_0_0	F20	LVCMOS33	AP9	LVCMOS18
P3_0_0	G20	LVCMOS33	AP10	LVCMOS18

P2_0_0	J20	LVCMOS33	AP12	LVCMOS18
TXAB_P1_0_0	J19	LVCMOS33	AN12	LVCMOS18

HI-1587PC TXCVR FPGA Board

An 5V 1-2A power adapter jack is provided for optional use. The transceiver board can be powered by the FPGA 3.3V supply when JP4 is connected. When using the external power adapter JP4 should be opened and a solder connection on JP6 is made.

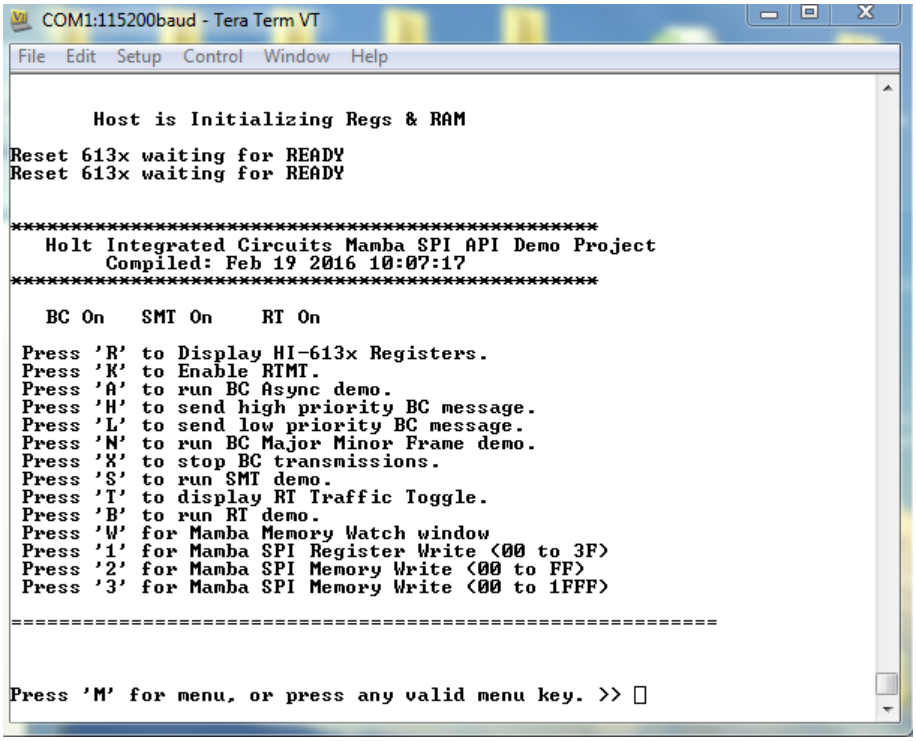
Some DC-DC converter power supplies may not be suitable for the HI-1587 due to poor transient response when the part transitions from idle to full transmit. During transmit the HI-1587 can draw up to 675mA. The VDD voltage should not dip below the minimum VDD = 3.14 listed in the data sheet.

The VDIO voltage pin is used to make it easier to match the voltage from the FPGA. For 3.3V connect VDIO to the 3.3V VDDA and VDDB pins. For 1.8V or 2.5V a suitable voltage source should be connected. The VDIO pin powers the internal digital I/O. See the schematic for the decoupling capacitor recommendations.

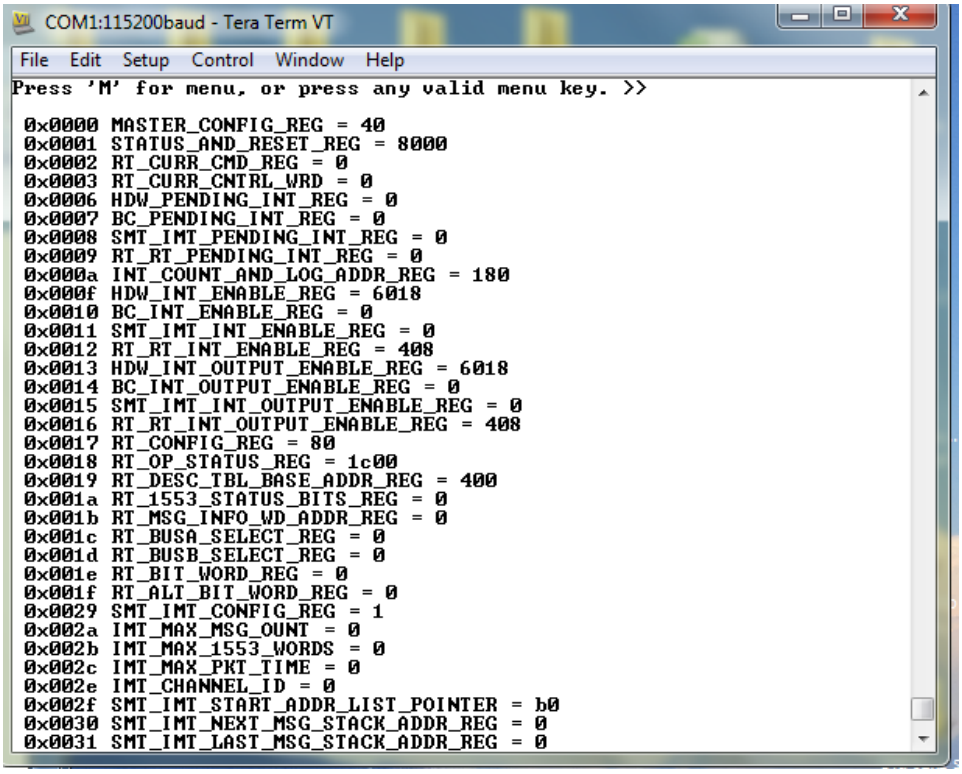
Demonstration Code

An embedded ARM Cortex-A53 interfaces with the HI-6300 IP Core via an AXI4 bus and executes example demonstration software. The demo project is a bare-metal implementation using the Holt API software library. The following sections give examples of accessing the software by a terminal console using a USB/UART connector.

Note: Dates and times shown will differ from the screen captures shown below.



Press 'R' or 'r' to display the HI-6300 registers.



The RT terminal address is set using DIP switches, before applying power. RT addresses 3 and 1 are utilized by the preprogrammed Bus Controller message repertoire. The 6-position DIP switch should already be set with the address values 03, plus odd parity.

If not connected by cable to conventional MIL-STD-1553 buses, a dummy 70Ω load for the buses is provided on the board by connecting solder jumpers JP8 and JP9.

General Structure of Demo Functions

The Holt API demonstration program is contained in module `demos.c`. The Holt API runtime library is contained in the library file `MAMBA API LIB.a` as executable object code. File `demos.c` contains the demo initialization API function calls supporting demonstrations executed from the console menu to initialize the BC, RT and monitor terminals. Key presses are detected in `console.c` which is called from the main loop in `main.c` and executes demo functions in `demos.c`.

Commands 'A' and 'N' transmit BC commands can be viewed on an oscilloscope and optionally display the message traffic data on the console using the 'K' and 'T' command sequence. These demos demonstrate how Holt API's are used to generate BC Asynchronous messages, Major/Minor frames, low priority and high priority messages. View these messages with external MIL-STD-1553 test equipment or view them with an oscilloscope.

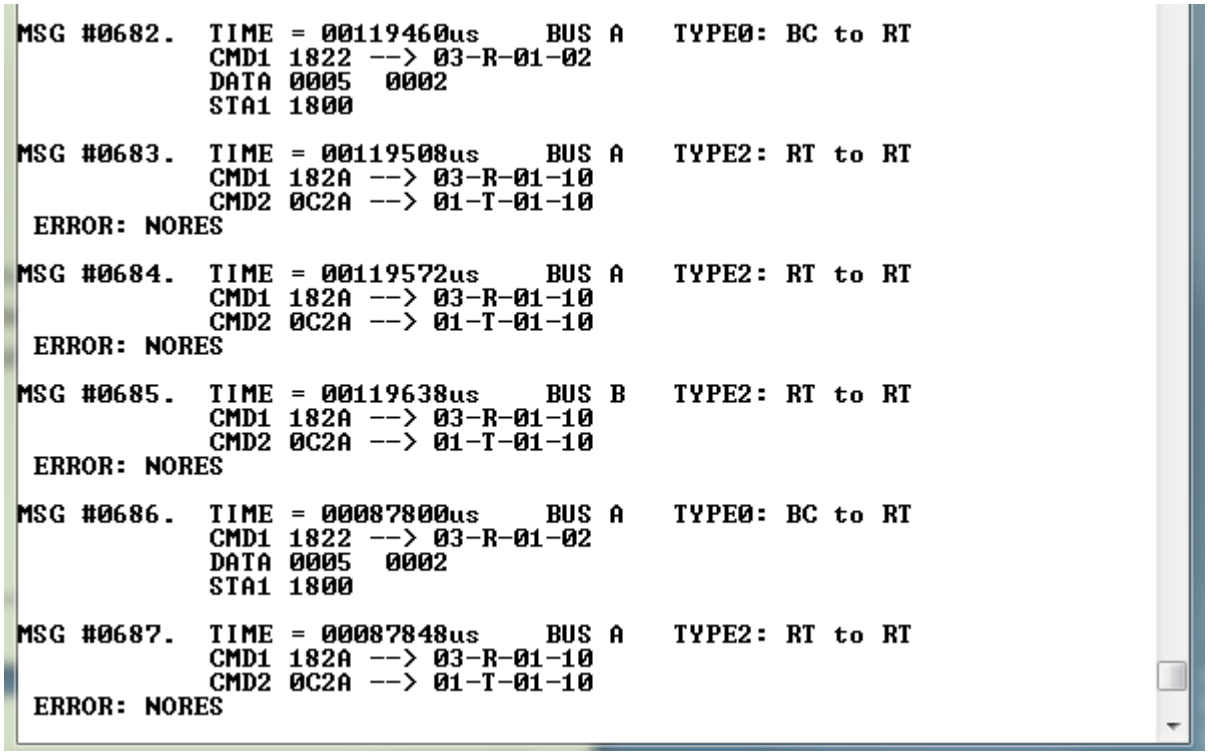
This exercise uses the internal BC to transmit messages, so message traffic data is displayed on the console. Since the internal BC, RTs and SMT share the same bus pins, the RT and SMT monitor terminals

receive the BC messages. If an external BC is already connected to the bus jack though a bus coupler, it is okay to leave it connected, but disable any external BC transmissions that will conflict with the on-chip BC transmissions. Without an external buses connected, close jumpers JP8 and JP9 (as described on page 6) to connect on-board 70Ω dummy bus load resistors across BUS A and BUS B.

BC and RT Mode

1. Press command 'B' to enable the RT.
2. Press command 'K' to enable the RTMT demo.
3. Press command 'T' (or spacebar) to display RT traffic on the console. Command 'T' toggles on and off alternately to enable or disable the RT traffic shown on the console. Using the 'T' command relies on prior execution of command 'K'.
4. Press command 'A' to start the BC transmitting messages.

   Messages will display rapidly on the screen, Press the space bar to stop the console output. The console should freeze and look similar to the screen below. Press space bar again to restart the console output. Using the space bar does not stop BC transmission or prevent RT or MT message reception; it only stops console output.



Some of the BC commands are RT to RT. The BC commands to RT3 only shows no errors but RT to RT commands to RT1 and RT3 will show " ERROR: NORES" since there's no RT at address 1.

The Bus A green LED flashes rapidly (but appears continuously lit) with this demo.

5. BC Low Priority Asynchronous Message Insertion

Command 'L' inserts a low priority message into the scheduled BC message list. Low priority Inserted messages occur upon completion of any BC minor frame in-process when insertion is requested. First, enable the RT by pressing 'B' then Press 'A' to enable the BC transmission. Press 'L' to transmit three extra messages on Bus B. Bus B is used to make it easier to see on the scope and the Bus B LED should flash. If the RT is not enabled, retry messages appear on Bus B; this makes it difficult to see the three inserted messages. This will only work once after a power up or RESET.

The screen shot of these three messages are shown below captured by a Ballard USB UA1133 tester.

Rec #	Time	Message	Bus	Error	Data 4x8	Chan	Swd Bits	Warning
0	T=000:00:0... dT=000:00:0...	Cwd1=0822 (01,R,01,02) <DATA WORDS> Swd1=0800	B		01: DEAD BEEF	1		
1	T=000:00:0... dT=000:00:0...	Cwd1=0C2F (01,T,01,15) <DATA WORDS> Swd1=0800	B		01: BBBB 0202 1414 ... 05: 0505 0606 0707 ... 09: 0909 1010 1111 ... 13: 1313 1414 1515	1		
2	T=000:00:0... dT=000:00:0...	Cwd1=0825 (01,R,01,05) <DATA WORDS> Swd1=0800	B		01: CAFE C0DE 0303 ... 05: 0505	1		

6. BC High Priority Asynchronous Message Insertion.

Follow the same steps as the previous BC low priority message example but this time Press 'H' to insert a single high priority message. This command is repeatable and the Bus B LED will flash with each command. Inserted High Priority messages occur upon completion of any in-process message when insertion is requested.

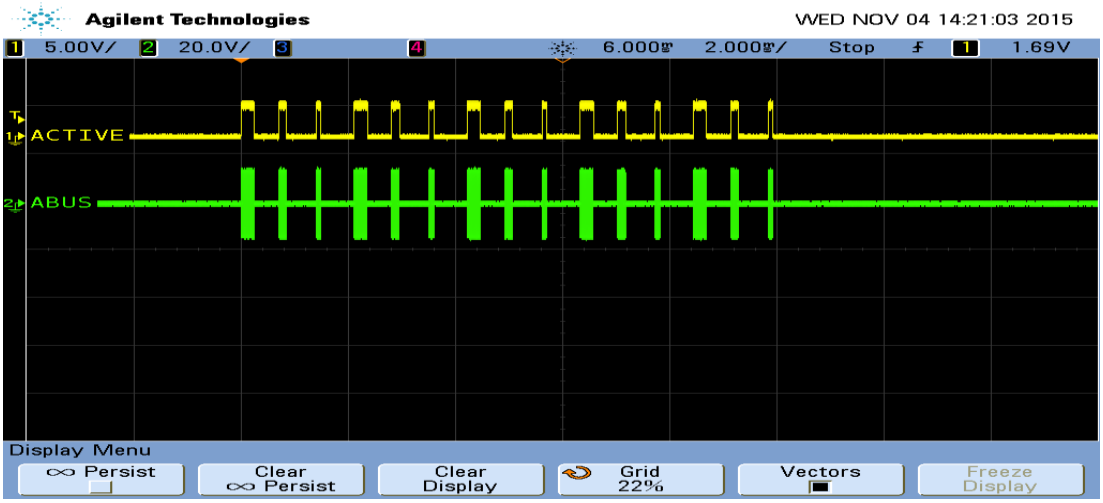
	T=000:00:0... dT=000:00:0...	Cwd1=0822 (01,R,01,02) <DATA WORDS> Swd1=0800	B		01: DEAD BEEF	1		

7. Command 'E' Enumerate Card is reserved for future use.

8. From a RESET, if the BC is started before enabling the 'K' and 'T' sequence to display message traffic, the first message may contain an error. This is normal; this occurs because the RT and MT are enabled midstream of a message in progress.

9. The ‘N’ command transmits fifteen commands to RT address 3. Press ‘B’ to enable the RT.

Press ‘N’ to execute the BC transmissions (15 messages are transmitted) which will appear on the bus as shown below. To optionally see the message traffic on the console, enable the RT message traffic by pressing ‘T’ if it hasn’t already been enabled.



Command ‘N’ (15 message) Traffic capture using a Ballard USB 1553 monitor.

Rec #	Time	Message	Bus	Error	Data 4x8	Chan	Swd Bits	Warning
0	T=000:00:00:00.1583950 dT=000:00:00:00.000000	Cwd1=1C2A (03,T,01,10) <DATA WORDS> Swd1=1800	A		01: 1000 1001 1002 1003 05: 1004 1005 1006 1007 09: 1008 1009	0		
1	T=000:00:00:00.15831208 dT=000:00:00:00.000258	Cwd1=1825 (03,R,01,05) <DATA WORDS> Swd1=1800	A		01: AAAA 0202 0303 0404 05: 0505	0		
2	T=000:00:00:00.15831466 dT=000:00:00:00.000257	Cwd1=1822 (03,R,01,02) <DATA WORDS> Swd1=1800	A		01: BBBB 0202	0		
3	T=000:00:00:00.1584128 dT=-000:00:00:00.000337	Cwd1=1C2A (03,T,01,10) <DATA WORDS> Swd1=1800	A		01: 1000 1001 1002 1003 05: 1004 1005 1006 1007 09: 1008 1009	0		
4	T=000:00:00:00.1584446 dT=000:00:00:00.000318	Cwd1=1825 (03,R,01,05) <DATA WORDS> Swd1=1800	A		01: AAAA 0202 0303 0404 05: 0505	0		
5	T=000:00:00:00.1584704 dT=000:00:00:00.000258	Cwd1=1822 (03,R,01,02) <DATA WORDS> Swd1=1800	A		01: BBBB 0202	0		
6	T=000:00:00:00.1584967 dT=000:00:00:00.000262	Cwd1=1C2A (03,T,01,10) <DATA WORDS> Swd1=1800	A		01: 1000 1001 1002 1003 05: 1004 1005 1006 1007 09: 1008 1009	0		
7	T=000:00:00:00.15841224 dT=000:00:00:00.000256	Cwd1=1825 (03,R,01,05) <DATA WORDS> Swd1=1800	A		01: AAAA 0202 0303 0404 05: 0505	0		
8	T=000:00:00:00.15841542 dT=000:00:00:00.000318	Cwd1=1822 (03,R,01,02) <DATA WORDS> Swd1=1800	A		01: BBBB 0202	0		
9	T=000:00:00:00.1585205 dT=-000:00:00:00.000337	Cwd1=1C2A (03,T,01,10) <DATA WORDS> Swd1=1800	A		01: 1000 1001 1002 1003 05: 1004 1005 1006 1007 09: 1008 1009	0		
10	T=000:00:00:00.1585463 dT=000:00:00:00.000258	Cwd1=1825 (03,R,01,05) <DATA WORDS> Swd1=1800	A		01: AAAA 0202 0303 0404 05: 0505	0		
11	T=000:00:00:00.1585721 dT=000:00:00:00.000257	Cwd1=1822 (03,R,01,02) <DATA WORDS> Swd1=1800	A		01: BBBB 0202	0		
12	T=000:00:00:00.15851043 dT=000:00:00:00.000321	Cwd1=1C2A (03,T,01,10) <DATA WORDS> Swd1=1800	A		01: 1000 1001 1002 1003 05: 1004 1005 1006 1007 09: 1008 1009	0		
13	T=000:00:00:00.15851301 dT=000:00:00:00.000258	Cwd1=1825 (03,R,01,05) <DATA WORDS> Swd1=1800	A		01: AAAA 0202 0303 0404 05: 0505	0		
14	T=000:00:00:00.15851565 dT=000:00:00:00.000264	Cwd1=1822 (03,R,01,02) <DATA WORDS> Swd1=1800	A		01: BBBB 0202	0		

When a BC message is transmitted to a RT that is not enabled, “RT no response” (NORES) error is indicated.

```
MSG #0170.  TIME = 00086918us    BUS A   TYPE2: RT to RT
            CMD1 182A --> 03-R-01-10
            CMD2 0C2A --> 01-T-01-10
 ERROR: NORES
```

RT Mode using external BC

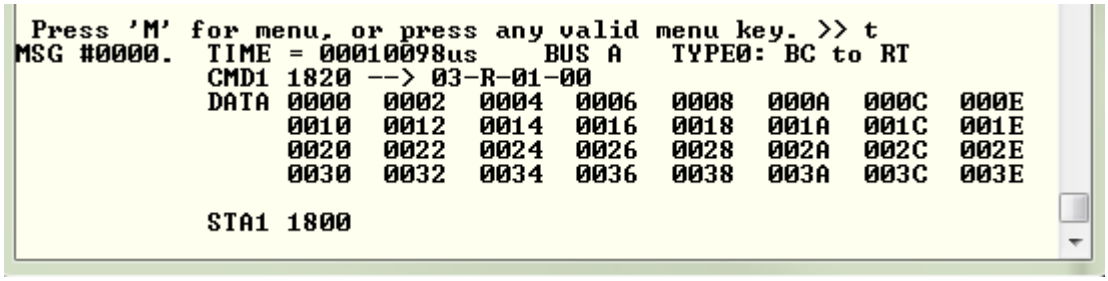
Using an external BC tester (such as Ballard USB 1553) to transmit messages to the demo board.

10. When an external BC is connected using conventional 1553 buses, use cables to connect the demo board circular tri-axial bus jacks to bus coupler ports on the A and B bus networks. In this case, the on-board dummy bus load 70 Ω resistors should be disconnected. See JP8 and JP9 described on page 6.

    If bus couplers are not readily available, bench testing can be done by enabling the on-board dummy bus load 70 Ω resistors (see page 6) and connecting BC tester cables directly to the demo board tri-axial jacks for buses A and B.

11. Press the RESET button and then Press 'B' to enable the RT then Press 'K' and 'T' to activate the RT traffic on the console. Compose a BC to RT message with SA=1 and 32 data words similar to the message shown below.

12. The console should show the message transmitted by the BC, after the transaction.

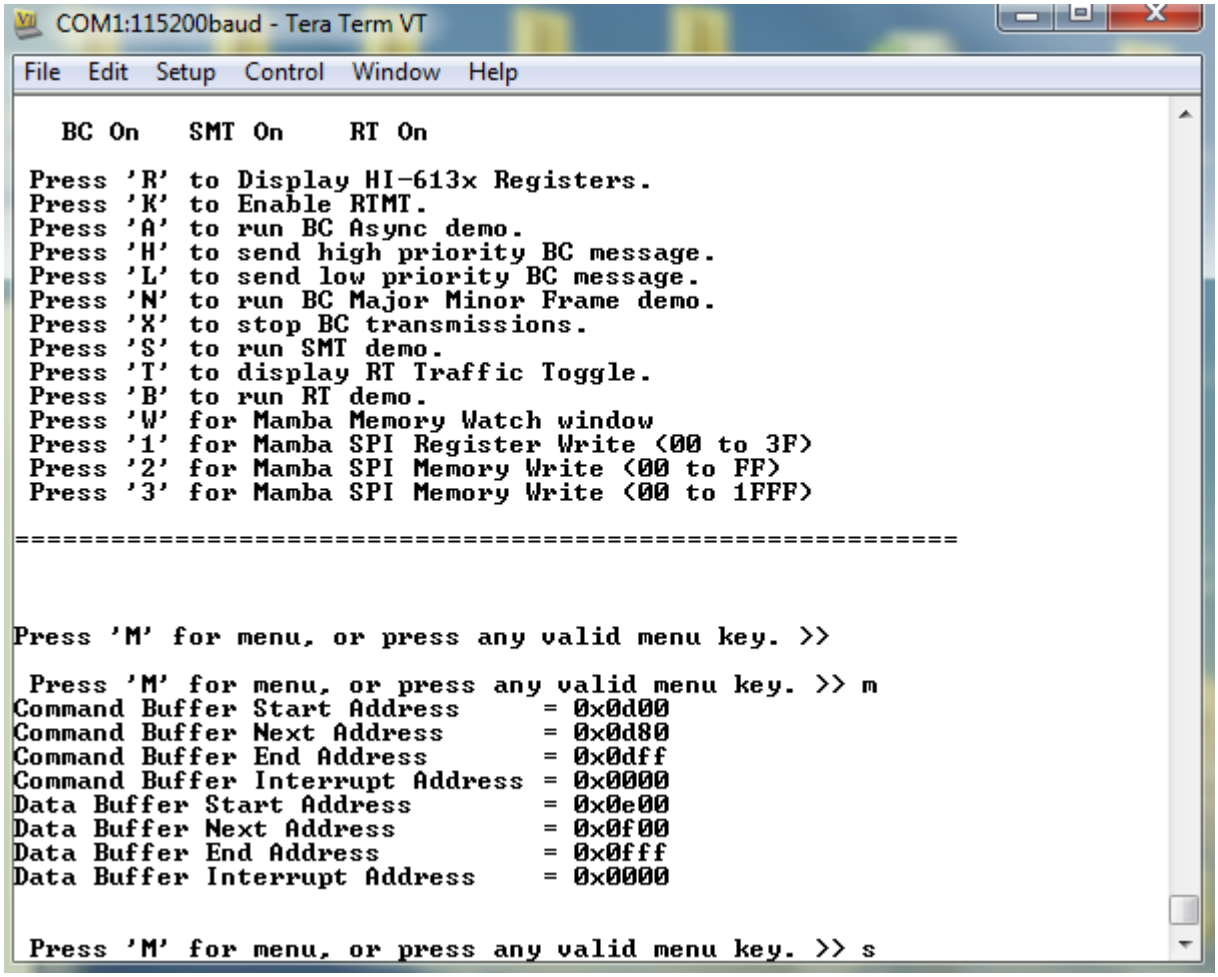


13. When transmitting repeating messages at a high rate typical of MIL-STD 1553, the RT Traffic shown on the console may not keep pace due to limitations of the console 115,200 baud rate and the prolific use of slow-to-execute `printf` function calls in the C program's console user interface. Depending on the message content and repetition rate, some messages may not show on the console. All messages are transacted properly and captured by enabled RT and MT, some messages simply will not be shown on the console.

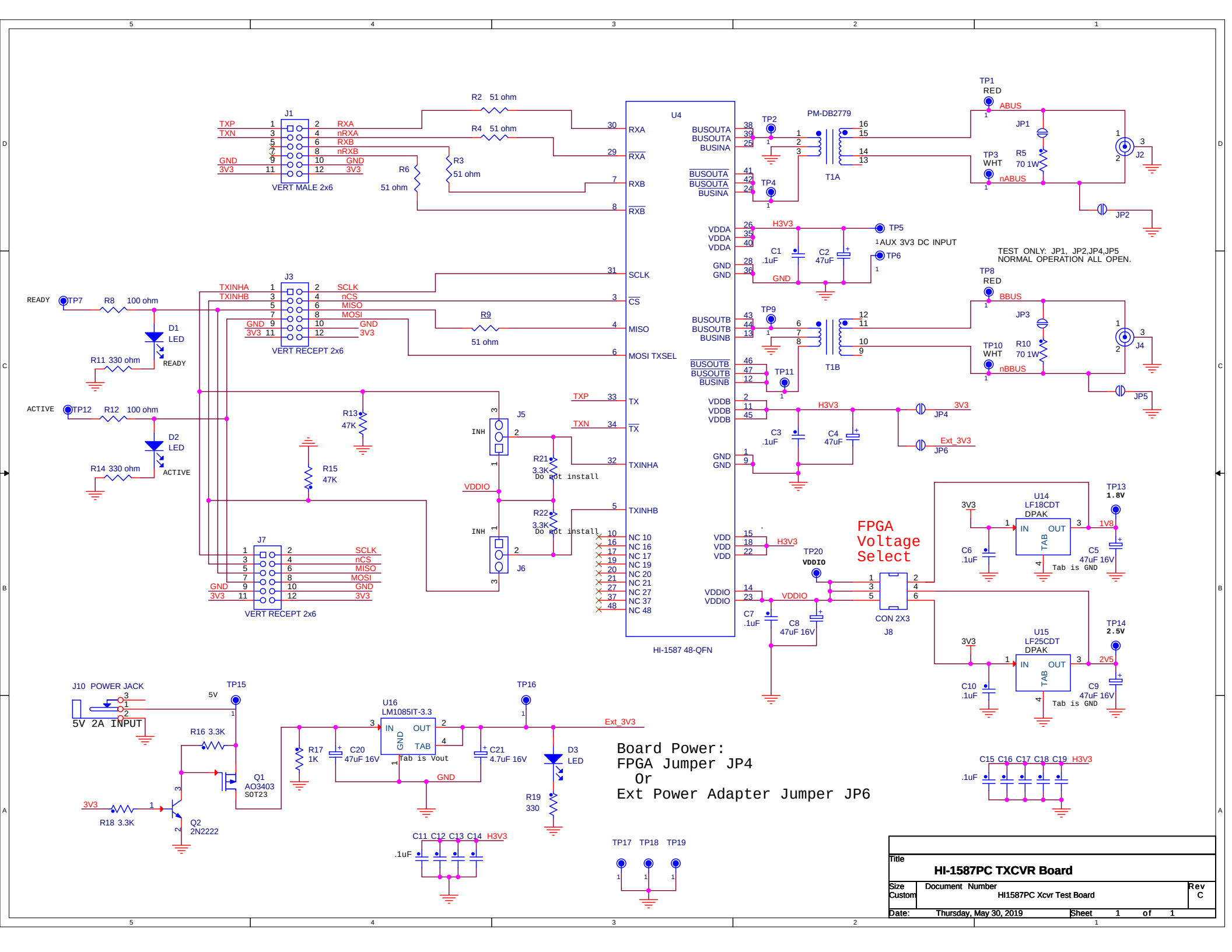
MT Mode

Press ‘S’ to enable SMT simple monitor. No other terminal is required; ‘S’ can be used after board reset.

A list of addresses shows the Command Stack and Data Stack buffer start address and end address. After sending some messages to the monitor use this command to display the addresses and use the Memory Watch window to view the Command and Data in memory. The SMT is also initialized with the ‘K’ command that provides message details. The ‘T’ RT Traffic feature toggles display of formatted RT message data on the console.



Item	Qty	Description	Reference	Digikey P/N	Mfg P/N
1	1	PCB, Bare, Evaluation Board	N/A	N/A	NewTek 13930
2	6	Capacitor, 47uF 20% 16V Tant SMD 6032	C2,C4,C5;C8;C9;C20	399-9739-1-ND	Kemet T491C476M016AT
3	1	Capacitor, 4.7uF 10% 16V Tant SMD 3216	C21	399-8439-1-ND	Kemet T494A475K016AT
4	14	Capacitor, Cer 0.1uF 20% 50V Y5V 0805	C1;C3;C6;C7;C10;C11;C12;C13;C14;C15;C16;C17;C18;C19	399-9157-1-ND	Kemet C0805C104M5VACTU
5	2	Connector 3-Lug Triax Jack, TRB BJ77	J2,J4	1097-1030-ND	Cinch BJ77
6	4	Connector Rcpt 12-pin IDC Gold	for J2 and J3	A113153-ND	TE 2-1658527-0
7	2	Header, Male 2x6, .1" Pitch	J3,J7	S2012EC-06-ND	Sullins PREC006DAAN-RC
8	2	Flat Ribbon Cable Gray 12 Conductors	6" long per/board	732-11802-ND	Wurth 63911215521CAB
9	1	Connector PWR Jack 2.1x5.5mm R/A Solder	J10	CP-102A-ND	Cui PJ-102A
10	1	Header, Male 2x6, .1" Pitch, R/A, 0.405" Tail	J1	3M156383-12-ND	3M 929745-02-06-EU
11	1	Header, Male 2x3, .1" Pitch	J8	S2011E-03-ND	Sullins PBC03DAAN
12	2	Conn Header .100 Sgl 3 Pos Male	J5,J6	S1011E-03-ND	Sullins PBC03SAAN
13	3	Shunt, 2 pin, 0.1"	for J5,J6 and J8	S9000-ND	Sullins STC02SYAN
14	6	Conn solder jumper	JP1;JP2;JP3;JP4;JP5;JP6	N/A	
15	3	Led Green SMD 0805	D1,D2,D3	160-1179-1-ND	LiteOn LTST-C170GKT
16	2	Res 69.8 Ohm 1W 1% 2512 SMD	R5;R10	RHM69.8BBCT-ND	Rohm MCR100JZHF69R8
17	5	Res 51, 1/8W 5% 0805 SMD	R2;R3;R4;R6;R9	P51ACT-ND	Panasonic ERJ-6GEYJ510V
18	2	Res 100, 1/8W 5% 0805 SMD	R8;R12	P100ACT-ND	Panasonic ERJ-6GEYJ101V
19	3	Res 330, 1/8W 5% 0805 SMD	R11;R14,R19	P330ACT-ND	Panasonic ERJ-6GEYJ330V
20	1	Res 1K, 1/8W 5% 0805 SMD	R17	P1.0KACT-ND	Panasonic ERJ-6GEYJ102V
21	4	Res 3.3K, 1/8W 5% 0805 SMD	R16;R18;R21;R22	P3.3KACT-ND	Panasonic ERJ-6GEYJ332V
22	2	Res 47K, 1/8W 5% 0805 SMD	R13;R15	P47KACT-ND	Panasonic ERJ-6GEYJ473V
23	4	Test Point, 0.040"	TP2;TP4;TP9;TP11 (DNI)		
24	5	Test Point, Orange Insulator, 0.062"	TP5,TP13,TP14,TP16,TP20	36-5013-ND	Keystone 5013
25	3	Test Point, Red Insulator, 0.062"	TP1,TP8,TP15	36-5010-ND	Keystone 5010
26	6	Test Point, Black Insulator, 0.062"	TP3,TP6,TP10,TP17,TP18,TP19	36-5011-ND	Keystone 5011
27	1	Test Point, Yellow Insulator, 0.062"	TP7	36-5014-ND	Keystone 5014
28	1	Test Point, White Insulator, 0.062"	TP12	36-5012-ND	Keystone 5012
29	1	HI-1587PC 48-QFN 6X6mm	U1	N/A	Holt HI-1587PC
30	1	Isolation Transformer PM-DB2779	T1	N/A	Holt-Premiers Magnetics DB2779
31	1	IC Reg Linear 3.3V 3A TO-263-3	U16	L085ISX-3.3/NOPBCT	TI LM1085ISX-3.3/NOPB
32	1	IC Reg Linear 2.5V 500mA DPAK	U15	497-6449-1-ND	STM LF25CDT-TR
33	1	IC Reg Linear 1.8V 500mA DPAK	U14	497-5222-1-ND	STM LF18CDT-TR
34	1	Mosfet P-CH 30V 2.6A SOT-23	Q1	785-1003-1-ND	Alpha AO3403
35	1	Trans NPN 30V 0.6A SOT-23	Q2	MBT2222LT1GOSCT-	ON MMBT2222LT1G
36	1	AC/DC Wall Mount Adapter 5V 2A 10W	n/a	102-4136-ND	Cui SWI10-5-N-P5
37	4	Hookup Solid wire - 20AWG - Black - 1" Long	Triax wiring	C2028B-XX-ND	General Cable C2028A.12.01
38	4	Stand-off, Threaded #4-40F, 3/4" Long Round	n/a	36-3481-ND	Keystone 3481
39	4	Machine Screw, #4-40 x 5/16"	n/a	H343-ND	B&F Supply PMS 440 0025 PH
40	4	Lock Washer, Int.Tooth #4-40	n/a	H236-ND	B&F Supply INTLWZ 004
		** Remark: Mounting J2 and J3 with its GND tab is on Top.			


J1
VERT MALE 2x6
TXP
TXN
GND
3V3
RXA
nRXA
RXB
nRXB
R2 51 ohm
R4 51 ohm
R3 51 ohm
R6 51 ohm
U4
HI-1587 48-QFN
PM-DB2779
T1A
T1B
TP1 RED
ABUS
JP1
R5 70 1W
J2
TP3 WHT
nABUS
JP2
TP8 RED
BBUS
JP3
R10 70 1W
J4
TP10 WHT
nBBUS
JP5
TEST ONLY: JP1, JP2,JP4,JP5
NORMAL OPERATION ALL OPEN.
TP5
AUX 3V3 DC INPUT
H3V3
C1 .1uF
C2 47uF
C3 .1uF
C4 47uF
JP4
3V3
Ext_3V3
JP6
J3
VERT RECEPT 2x6
TXINHA
TXINHB
SCLK
nCS
MISO
MOSI
R9 51 ohm
MOSI TXSEL
READY
TP7
R8 100 ohm
D1 LED
R11 330 ohm
ACTIVE
TP12
R12 100 ohm
D2 LED
R14 330 ohm
R13 47K
R15 47K
J5
J6
VDDIO
R21 3.3K Do not install
R22 3.3K Do not install
J7
VERT RECEPT 2x6
TXP
TXN
TXINHA
TXINHB
NC 10
NC 16
NC 17
NC 19
NC 20
NC 21
NC 27
NC 37
NC 48
TP9
TP11
TP20 VDDIO
FPGA Voltage Select
CON 2X3
J8
C7 .1uF
C8 47uF 16V
U14 LF18CDT DPAK
TP13 1.8V
C6 .1uF
C5 47uF 16V
Tab is GND
U15 LF25CDT DPAK
TP14 2.5V
C10 .1uF
C9 47uF 16V
J10 POWER JACK
5V 2A INPUT
TP15
5V
R16 3.3K
Q1 AO3403 SOT23
R18 3.3K
Q2 2N2222
U16 LM1085IT-3.3
IN
OUT
GND
TAB
Tab is Vout
R17 1K
C20 47uF 16V
C21 4.7uF 16V
TP16
Ext_3V3
D3 LED
R19 330
C11 C12 C13 C14 H3V3
.1uF
C15 C16 C17 C18 C19 H3V3
TP17 TP18 TP19
Board Power:
FPGA Jumper JP4
Or
Ext Power Adapter Jumper JP6
Title
HI-1587PC TXCVR Board
Size Custom
Document Number
HI1587PC Xcvr Test Board
Rev C
Date: Thursday, May 30, 2019
Sheet 1 of 1