HOLT INTEGRATED CIRCUITS INC.

ADK-2130mPCIe Technical Manual

Jan. 16, 2020

This page intentionally blank

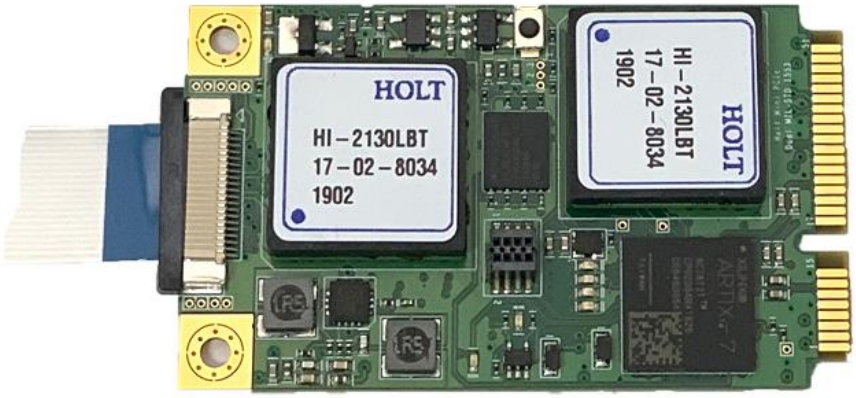
REVISION HISTORY

Revision	Date	Description of Change
AN-2130mPCIe Rev. New	1/16/20	Initial Release

Introduction

The Holt full size F2 Mini PCIe card reference design features one or two Holt HI-2130 MIL-STD 1553 multi-channel terminals with integrated transformers on a single Mini PCIe card. The card is designed to operate in a PC or single board computer with a Linux OS (note: Windows support will be available in the near future). The Demo software uses the Holt API Library functions, providing an abstraction layer that greatly simplifies host programming. This Technical Manual covers the hardware, software and instructions for how to transfer the Holt Flash Drive (FD) files to the user's PC. See the Quick Start Guide (QSG) for an introduction to the card, Application Development Kit (ADK) contents and how to run the demonstration software using the Holt bootable Flash Drive. Use the instructions in this guide to install the Holt software and Eclipse IDE tool used for software development on to a user's PC. Use the instructions in the QSG when booting from the Holt USB FD. See technical guide AN-PCIeFPGA for the FPGA design.

Figure 1 – Mini PCIe card EV-2130mPCIe-2F



Features

- Mini PCIe F2 full size, PCIe gen2 single lane
- Single Channel ADK-2130mPCIe-1F or Dual channel ADK-2130mPCIe-2F
- Mini Card Electromechanical Specification Rev. 1.2
- Independent two dual-redundant MIL-STD 1553 channels using proven HI-2130's
- Each Channel with BC, Dual RT and MT
- Transformer coupled MIL-STD 1553 interface
- Holt API Library support
- Linux OS
- Operating temperature -40° to +85°
- Customizable FPGA
- Demo software for BC, RT, RT2 and MT

PC requirements

Linux OS: Ubuntu: 16.04.6 LTS or 18.04 LTS (Holt FD installed).

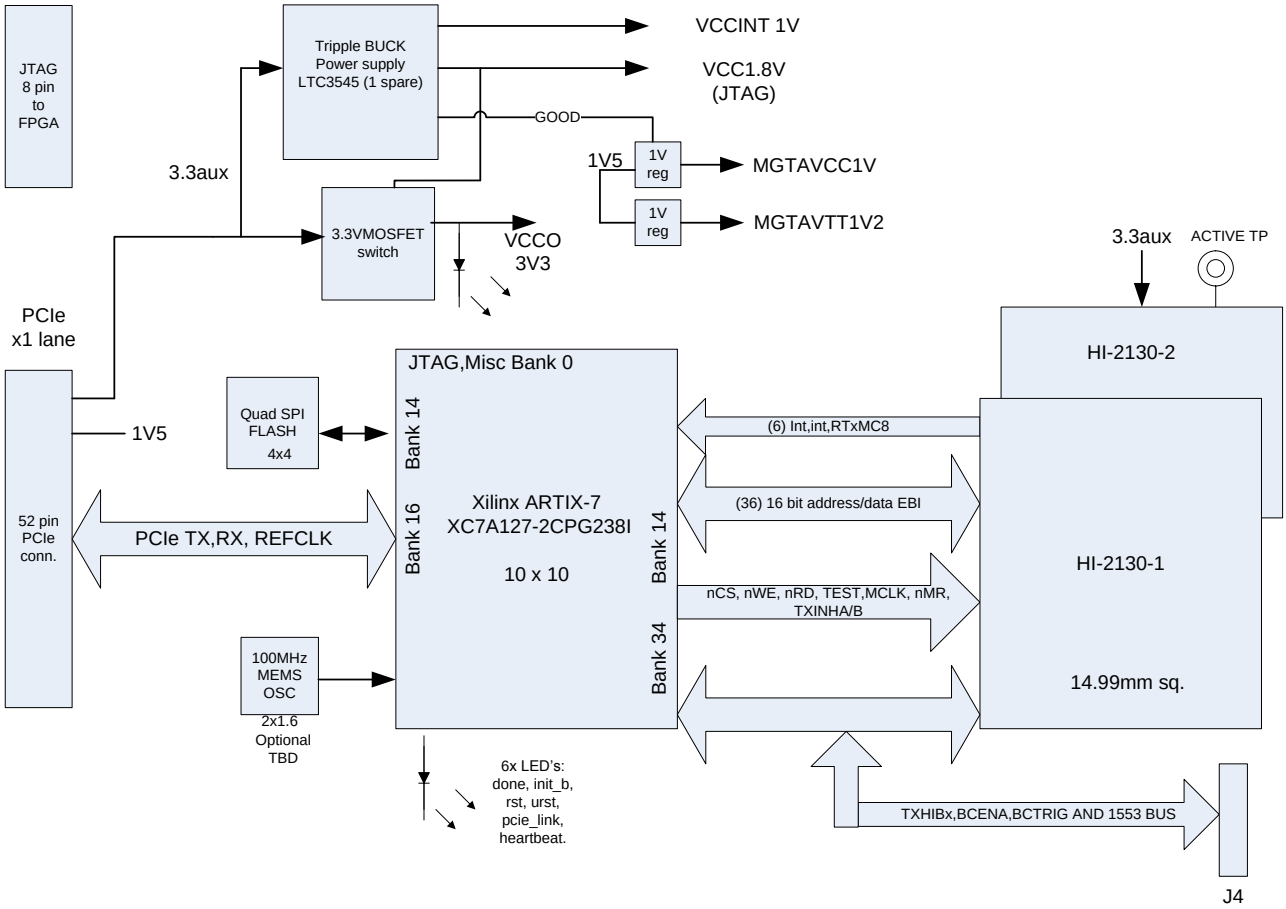
System RAM: 8G is recommended for Eclipse, 16G for Vivado.

Hard Drive Capacity: 10G minimum for the Eclipse project.

Mini PCIe Slot Standard Full size F2

USB 2.0 or 3.0 Port: For transferring Holt Flash Drive files.

Mini PCIe board block diagram



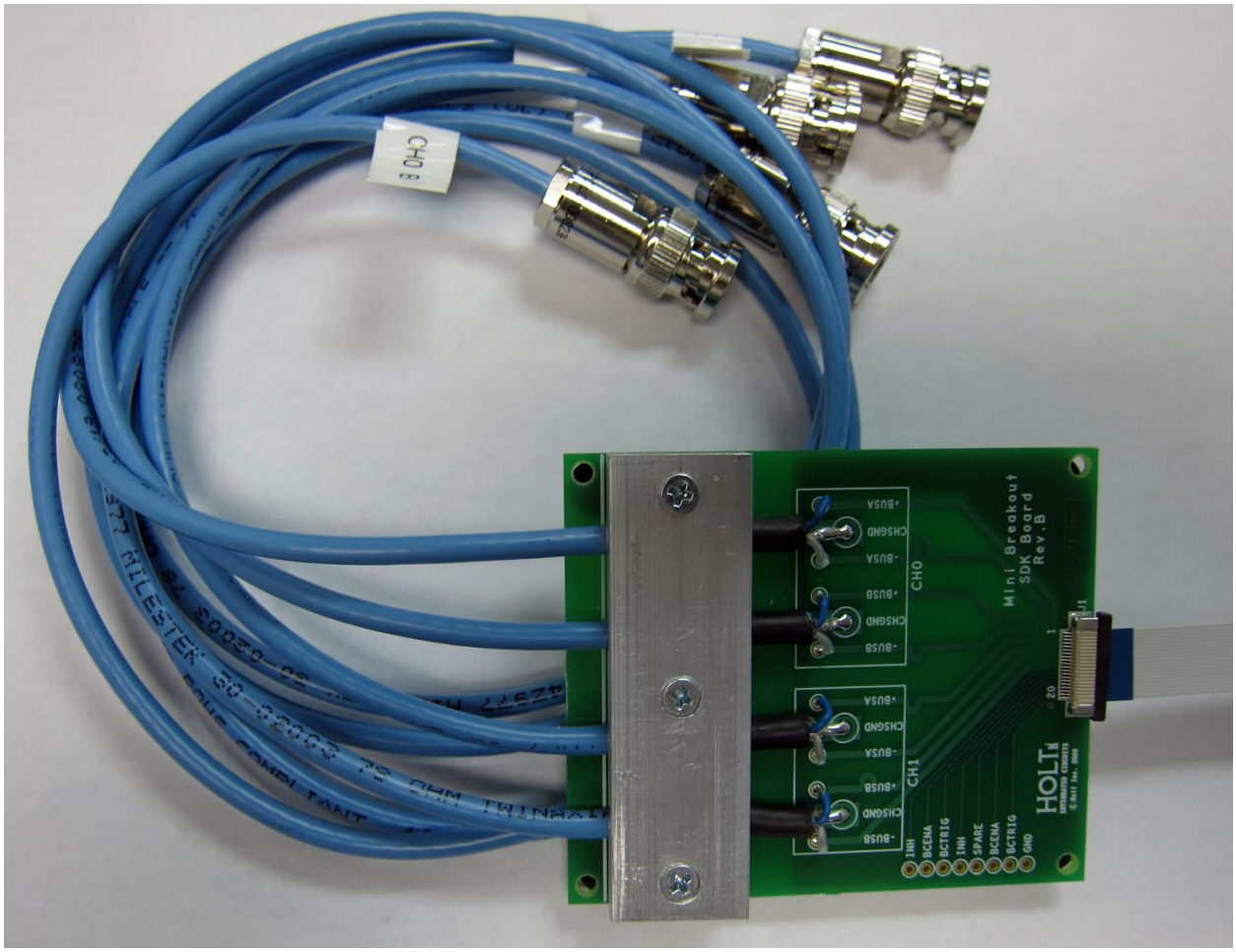
Break Out Board

The Break Out board connects to the PCIe card through a ribbon cable and is used to break out both BusA and BusB connections from both channels (Dev0 and Dev1). The Transmit Inhibits inputs are pulled high which present a low to the HI-2130 devices which enables 1553 bus transmissions by default. Table 8 is a list of the signals and descriptions on the inter-connect ribbon J4 connector. For board component locations, connectors, IC's and test points, see Figure 3 – Board References at the end of the document. Two break-out boards are available.

Single channel: mPCIe_breakout-1F

Dual channel: mPCIe_breakout-2F

Figure 2 – mPCIe_breakout-2F break-out board



First Things First

In order to successfully execute the Holt demo program the PC must be able to detect the card's PCIe interface at power up.

1. Reviewing the Holt QSG-2130mPCIe.pdf prior to reading this document is highly recommended.
2. Linux experience and the terminal window with the command line are highly recommended.
3. Power down the PC and install the Holt Mini PCIe card. Connecting the break-out board is not critical for this step so is optional.
4. The Ubuntu OS on the Holt FD cannot be used to install Ubuntu on a user's PC. If the user has an older version of Ubuntu than 18.04 LTS it may be possible to upgrade it.
   We recommend downloading Ubuntu 18.04 LTS from the Ubuntu website and using the provided installation instructions. Holt has not tried any other Linux version as of this date. Windows support is planned in the future.
   https://ubuntu.com/#download

5. Power up the PC. Shortly after powering up the computer the green PCIe Link LED 8 should be on and the Heart Beat LED 9 should flash on and off.
6. When LED 8 illuminates, the computer boot up sequence has detected the PCIe link on the card. Another way to verify this is to open a terminal window and execute the following command.

   ```
   holt@holt-desktop:~/holt$ lspci | grep Xilinx
   "Memory controller: Xilinx Corporation Device 7011"
   ```

   If this is displayed the PC successfully detected the PCIe link.

   If the 'Xilinx Corporation Device 7011" was not displayed try turning off the PC and reinstalling the card making sure the card is fully plugged in. If the LED's never light up it might be due to missing 1.5V on the Mini PCIe slot. Refer to the PC motherboard manual to see if 1.5V needs to be enabled by a configuration or jumper setting. Another PC may need to be used.

PC preparation for installing Eclipse and the Holt software

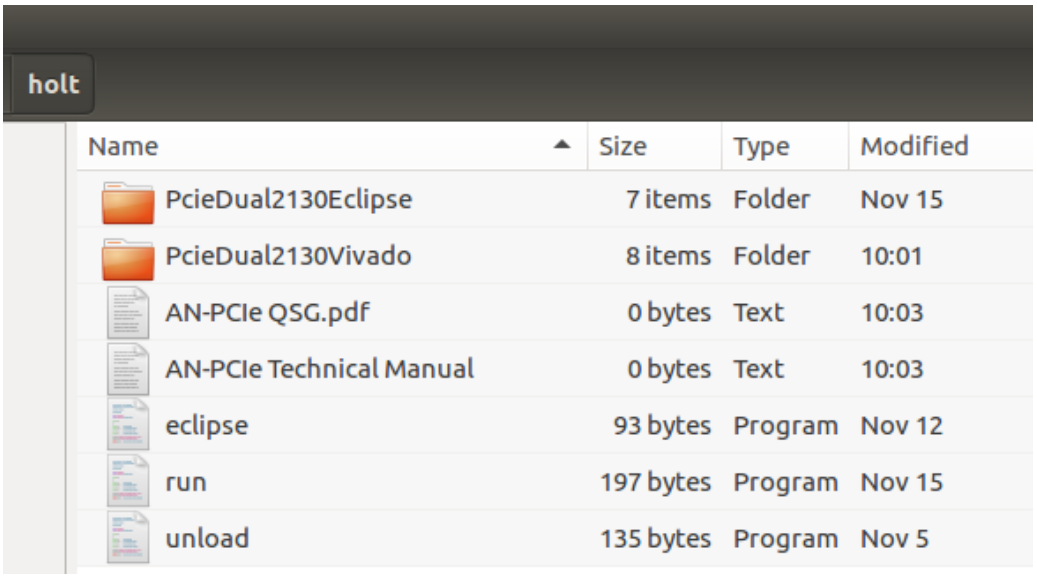
The Eclipse C/C++ IDE tool is free from www.eclipse.org and is used for this project. If Eclipse is already installed this step can be bypassed.

1. Java JRE must be installed. Open a terminal window and type “sudo apt install default-jre” and press return. Confirm the installation and press (Y). After Java is installed enter “java –version” to see a message with a version to confirm the installation.

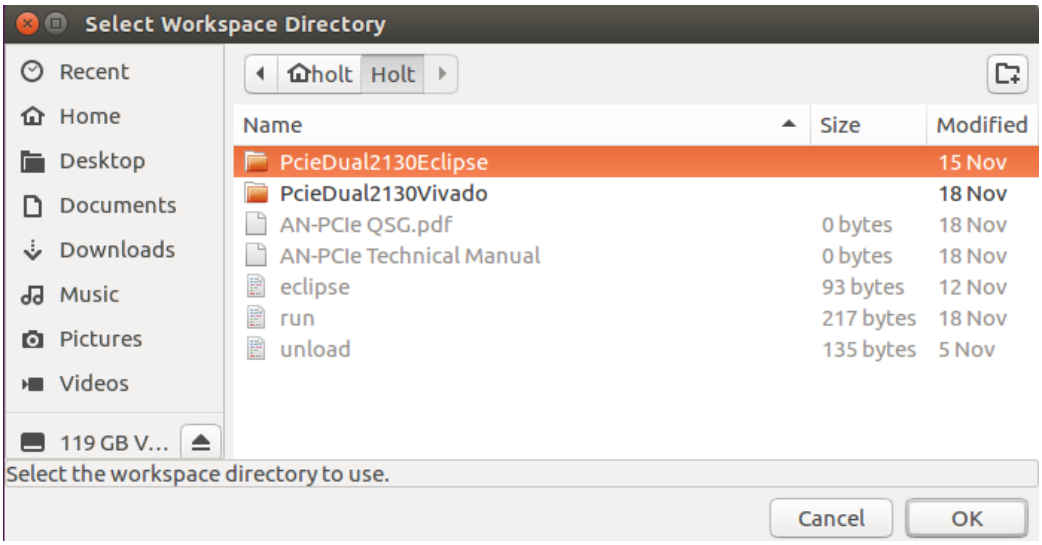
2. Go to the eclipse download page and download software. If using the eclipse installer select the C/C++ package. After the installation is complete optionally launch eclipse to see it working then exit the program. It’s possible to create a program short-cut from a terminal window where eclipse is installed then drag the link to the desktop. Navigate to where Eclipse is installed and right click on the eclipse program icon and select “Make Link” to create a link. Drag the link to the desktop.

The Holt demo software is comprised of three project sub-folders residing in a single Eclipse workspace folder.

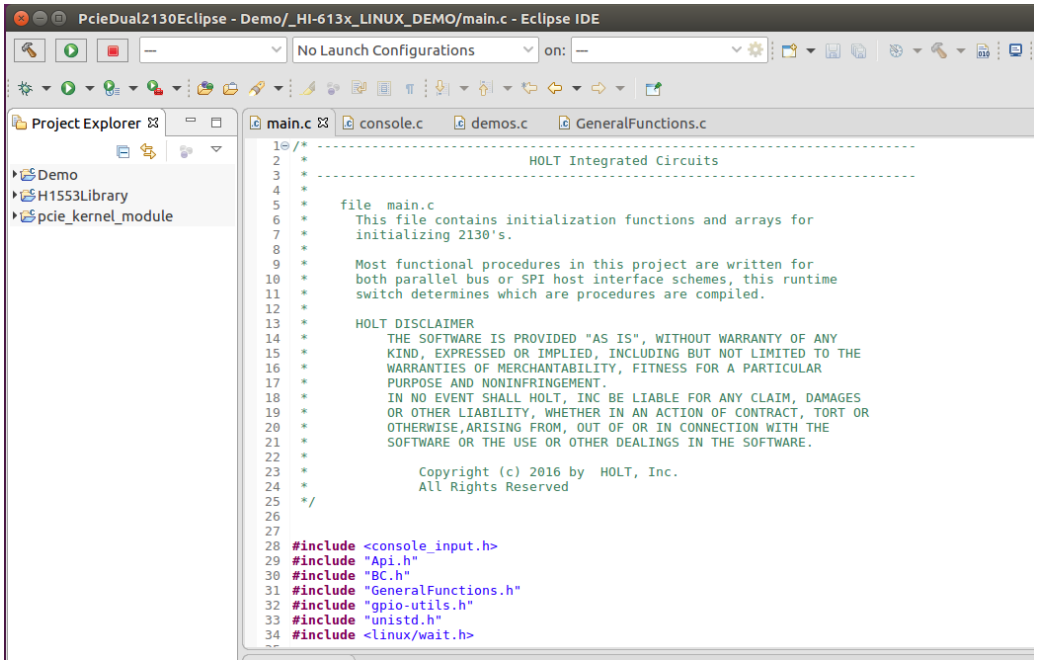
3 On the Ubuntu OS home folder create a Holt folder.
Copy all the folders, script files and other miscellaneous documents from the Flash Drive to the newly created Holt folder on the PC. The folder and files should appear similar to this.



3. Launch Eclipse and use the Browse button to select the project folder shown below and click the OK button.



4. When Eclipse opens the project the first time the projects must be imported. If the Eclipse Project Explorer (PE) window isn't shown type in "Project Explorer" into the Quick Access window located on the top right corner of the screen and select it from the list of items shown. The three project folder should be shown in the PE.



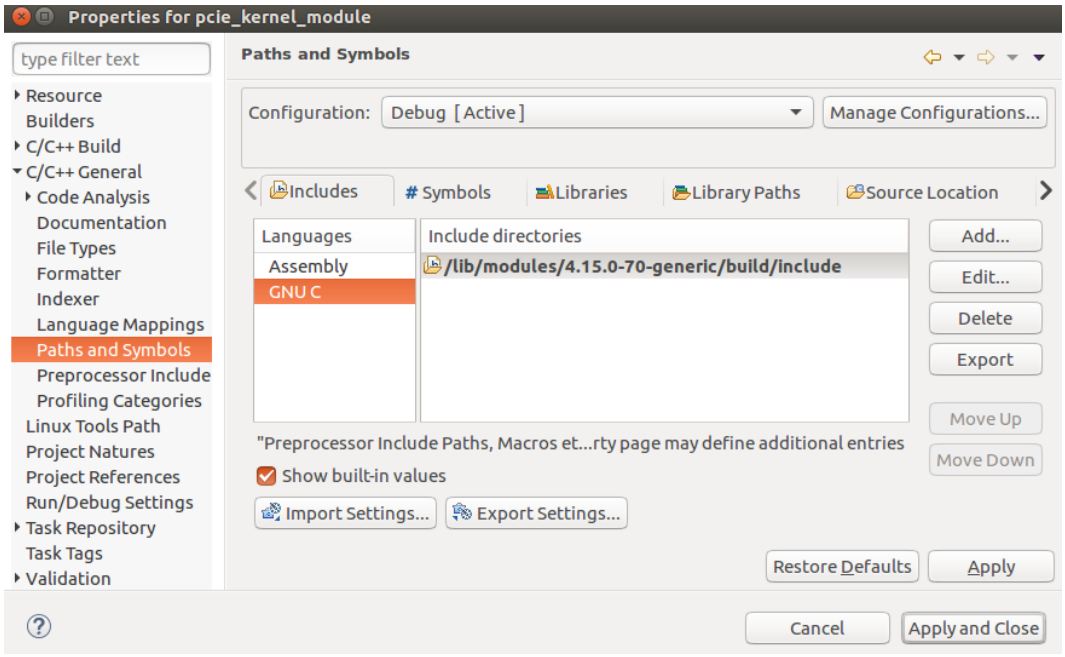
5. Project descriptions for the three project are contained in sub-folders:

Demo – The Demo software resets both devices and initializes both devices using the Holt API library functions. There are two build configurations: Debug and Debug_precompiled_library. The Debug_precompiled_library requires a binary library file produced by the H1553Libary project. See below. Rebuilding the 'Debug' configuration compiles all the demo files including the Holt API library files and outputs the Demo executable.

H1553Libary – This project builds the API library output file libH1553Library.a that is linked with the Debug_precompiled_libary target demo project above.

pcie_kernel_module – This project builds the Linux loadable kernel module 'pcie_lkm.ko'. This module must be loaded into the Linux OS before executing the Demo project or an error will be produced. This module provides support for writing and reading HI-2130 device registers and memory space using the Linux default PCIe drivers.

6. Steps to rebuild the projects before executing the Demo executable.
   DO THIS FIRST!
   The kernel module must be rebuilt with the version of OS header files located in the /lib/modules on the PC. Typically there's a folder named like 4.15.0-70-generic or similar. The path of this folder needs to be set in the kernel module project settings as shown below. This is accomplished by right clicking the kernel module project folder and selecting "properties" then selecting the "paths and Symbols items as shown. If the entry does not match the folder name on the PC edit the entry to make it match.



7. Rebuild the kernel module project. Before rebuilding the kernel module project it's a good idea to perform a project 'clean' first. Perform a project clean by right-clicking on the 'pcie_kernel_module' folder in the PE area and select 'clean' from the context menu. After the clean is performed rebuild the project also from the context menu. The project should build without any errors. There are typically some warnings which can be ignored. The build produces a new version of pcie_lkm.ko kernel module. To allow this module to be loaded into the OS kernel perform this command in a terminal window.

```
sudo chmod 777 pcie_lkm.ko  or  sudo chmod +x pcie_lkm.ko
```

8. Open a terminal window in the pcie_kernel_module subfolder and execute this bash script to load the newly built kernel module.

```
sudo  sh holt_pcie_load
```

Loading the kernel module is only required once after powering up the PC. If the kernel module project is modified and rebuilt the exiting kernel module must be unloaded before the new one is reloaded.

```
sudo sh unload_pcie_load
sudo sh load_pcie_load
```

These script commands were embedded in the 'run' script used on the QSG instructions.

9. Clean and Rebuild the H1553Library project using the same steps previously provided. Navigate in the PE and find the file: libH1553Library.a. Copy this file and paste it in the Demo project folder in PE.

10. Clean and Rebuild the Demo project. There are two Demo project targets: "debug" rebuilds the Holt API source files and Demo files to produce the Demo executable. This is the easiest and fastest way to rebuild the Demo project. Without Holt API source files the Debug_precompiled_library target must be used. This project uses the libH1553Library.a file which should already be in the Demo folder that was pre-built and provided by Holt.

TIP: An alternate method to rebuild any of the projects is to first select the corresponding project folder in the PE window area then click on the build "hammer" Icon that appears on the top of the Eclipse screen. Select the build target using the pull-down menu.

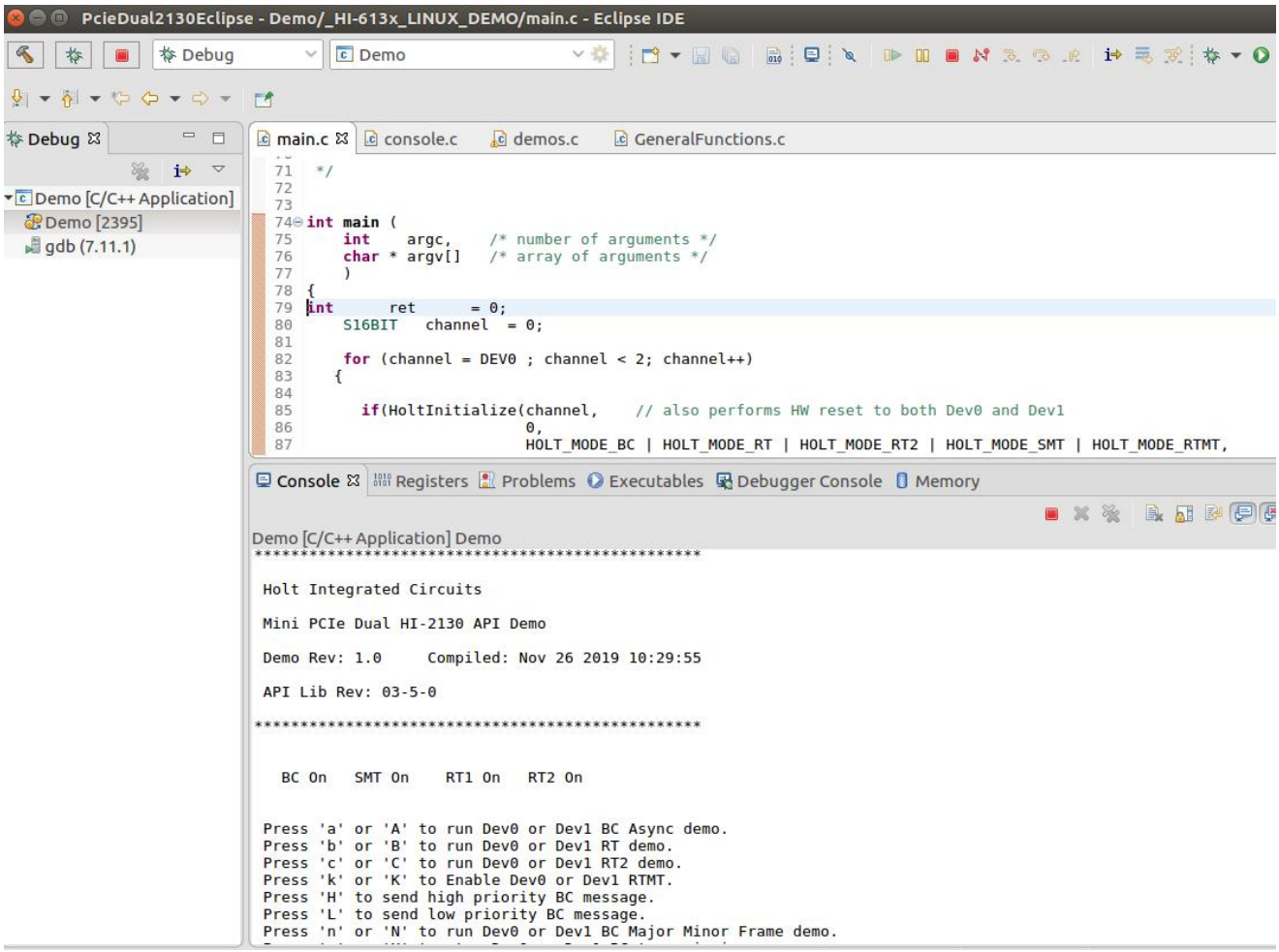
11. There are two ways to run the Demo (after loading the kernel module and rebuilding the Demo project). The first method is to simply open a terminal window in the Holt directory and execute the 'run' bash script. This will load the kernel module and execute the Demo program automatically. This method was used in the QSG document.

12. Run the Demo with the integrated debugger in Eclipse.

    a. First, ensure the kernel module was previously loaded.
    b. In Eclipse, click on the green (bug) debug icon near the top left side or top right side of the screen. A simple window appears asking to select either C/C++ Controller Application or Local C/C++ Application. Select Local C/C++ Application.
    c. The debugger should present main.c in a window with the first line of code highlighted in green. The user can now press the Resume icon to run the program or press F8.

    d. The debugger allows breakpoints, single-stepping and other debugger capabilities typical of a high end integrated source level debugger. See the Eclipse help to gain a better understanding how to use Eclipse debugger.

    e. The Demo console menu appears in a Console window. Expand the window size by double clicking on console tab. The full menu should be shown. When making code changes use the Eclipse debugger. Executing the Demo from the terminal window (not using Eclipse) presents a little better view and experience.


PcieDual2130Eclipse - Demo/_HI-613x_LINUX_DEMO/main.c - Eclipse IDE
Debug
Demo
Debug
Demo [C/C++ Application]
Demo [2395]
gdb (7.11.1)
main.c
console.c
demos.c
GeneralFunctions.c
71 */
72
73
74 int main (
75 int argc, /* number of arguments */
76 char * argv[] /* array of arguments */
77)
78 {
79 int ret = 0;
80 S16BIT channel = 0;
81
82 for (channel = DEV0 ; channel < 2; channel++)
83 {
84
85 if(HoltInitialize(channel, // also performs HW reset to both Dev0 and Dev1
86 0,
87 HOLT_MODE_BC | HOLT_MODE_RT | HOLT_MODE_RT2 | HOLT_MODE_SMT | HOLT_MODE_RTMT,
Console
Registers
Problems
Executables
Debugger Console
Memory
Demo [C/C++ Application] Demo
Holt Integrated Circuits
Mini PCIe Dual HI-2130 API Demo
Demo Rev: 1.0 Compiled: Nov 26 2019 10:29:55
API Lib Rev: 03-5-0
BC On SMT On RT1 On RT2 On
Press 'a' or 'A' to run Dev0 or Dev1 BC Async demo.
Press 'b' or 'B' to run Dev0 or Dev1 RT demo.
Press 'c' or 'C' to run Dev0 or Dev1 RT2 demo.
Press 'k' or 'K' to Enable Dev0 or Dev1 RTMT.
Press 'H' to send high priority BC message.
Press 'L' to send low priority BC message.
Press 'n' or 'N' to run Dev0 or Dev1 BC Major Minor Frame demo.

Demo project description

The Demo software initializes the Linux drivers which allow the host to communicate with the HI-2130 devices through the PCIe interface. This is performed in HoltInitialize() in main.c. The Xilinx FPGA serves as the hardware interface between the PCIe bus and both HI-2130's.

A menu is presented on the console and the program waits for the user to press a console command. Commands are accepted by the chk_key_input() function. Some of the code for utilities is contained in console.c.

All 1553 BC, RT and SMT demos contained in module Demo.c use Holt API library functions to initialize and control the terminals. The BC is initialized to transmit messages and the RT and SMT are initialized to be able to read and write 1553 data words. The Holt HI-613x API manual should be studied along with the Demo code to gain a better overall understand how the code works before making modifications.

Demo Preparations

Before running the Demo, connect the ribbon cable between the PCIe card and the Break Out board.

> Carefully insert the small ribbon cable to the Mini PCIe Card J4 connector and the other end to the Break Out board. The cable is fine pitched and the connector plastic fasteners are delicate so care must be exercised to avoid damaging the connectors. Using two hands carefully pull out the plastic fastener, insert the cable and push in the fastener on both sides.

It's possible to exercise the demonstrations without the Break Out board connected but this is not recommended and may not always work due to lack of 1553 bus termination.

For a review of the Demos refer to the QSG where some examples and console screens were presented. Some of the console commands already covered in the QSG will not be covered here.

As mentioned in the QSG the two RT addresses for both devices are the same. RT1=3 and RT2=1. This simplifies the demos since the BC demos transmit message commands to only these two RT addresses and the RT's are programmed with the same RT addresses. The user can use command '9' to change the RT addresses for other purposes.

Demos

Execute the Demo program by executing the 'run' script in the Holt folder. The QSG provides instructions how to execute the 'run' script and run most of the demos.

The run script contains eight terminal commands.

```
cd PcieDual2130Eclipse/pcie_kernel_module
sudo sh holt_pcie_unload
sudo sh holt_pcie_load
cd ..
cd Demo
cd Debug
ls
sudo ./Demo
```

```
holt@holt-desktop:~/holt$ sh run
06:00.0 Memory controller: Xilinx Corporation Device 7011
[sudo] password for holt:
Holt Linux driver unloaded
Holt Linux driver loaded
1553LibrarySrc_linkedfolder  Demo  _HI-613x_LINUX_DEMO makefile
objects.mk  sources.mk
Setting nMR chan 0 LOW Setting nMR chan 0 HIGH READY asserted
Setting nMR chan 1 LOW Setting nMR chan 1 HIGH READY asserted
Number of Devices found: 2


 Initial default RT addresses:
 DEV0:RT1=3 DEV0:RT2=1  DEV1:RT1=3  DEV1:RT1
 Optionally use console command '9' to change these RT addresses
BEFORE RUNNING RT

***********************************************
 Holt Integrated Circuits
 Mini PCIe Dual HI-2130 API Demo
 Demo Rev: 1.0     Compiled: Dec 11 2019 08:41:16
 API Lib Rev: 03-5-0
***********************************************

   BC On   SMT On    RT1 On   RT2 On

 Press 'a' or 'A' to run Dev0 or Dev1 BC Async demo.
 Press 'b' or 'B' to run Dev0 or Dev1 RT demo.
 Press 'c' or 'C' to run Dev0 or Dev1 RT2 demo.
 Press 'k' or 'K' to Enable Dev0 or Dev1 RTMT.
 Press 'l' or 'L' to send high priority BC message.
 Press 'h' or 'H' to send low priority BC message.
 Press 'n' or 'N' to run Dev0 or Dev1 BC Major Minor Frame demo.
 Press 'x' or 'X' to stop Dev0 or Dev1 BC transmissions.
 Press 'S' to run SMT demo.
 Press 't' to display RT Traffic Toggle.
```

```
------------ Utilities -------------
Press 'r' or 'R' to Display Dev0 or Dev1 HI-2130 Registers.
Press 'w' for Memory Watch window
Press 'f' Reads J4 connector and FPGA control signals
Press '1' for Register Write
Press '2' for Memory Write
Press '3' RT Mode Code data word reads
Press '4' Master Reset and reinitializes terminals
Press '5' Toggle Dev0 BCENA on/off
Press '6' Toggle Dev1 BCENA on/off
Press '9' Set RT addresses
Press '0' Toggle between User and Demo(default) modes

Press 'M' for menu, or press any valid menu key. >>
```

Note: For a -1F card (one HI-2130) the upper case commands will not be presented on the console menu.

In the following descriptions some commands such as ‘r’ will have an upper case ‘R’ equivalent command for the second device. Some of the examples are shown for Dev0.

Command ‘r’ displays the devices system registers by name and value. This is useful for checking register settings.

Command ‘w’, Memory Watch window reads 256 words starting with location 0x0000 of each device. This is useful to see all the system registers at a glance. Using sub-commands allow moving up and down in memory space. This is more useful to see large areas of memory such as RT control blocks, BC message lists or interrupt log tables.

Command ‘1’, writes to a system register 0x0000-0x004F by device.

Command ‘2’, writes to any register/memory 0x0000-0x7FFF by device.

Command ‘n’ commands the BC to transmit 15 messages.

Use command ‘n’ or ‘N’ to command the BC to transmit 15 messages. If an external RT is connected to the break out board and the address is 3 the external RT should receive 15 messages. Nothing will be shown on the console menu. To verify the BC is transmitting, un-connect the external RT and enable an internal RT using command ‘b’ and also Press ‘k’ and ‘t’ . These commands enable RT3 of Dev0, so RT traffic data is displayed on the console after message reception. When command ‘n’ is pressed, the RT traffic data should be shown, similar to what is presented below. There is no menu command to disable an internal RT once it’s been enabled but the card and program can be reset by using command ‘4’ to start similar to the state when the Demo program is first launched.

```
>n
>
```

```
Dev0 MSG #0000.  TIME = 00040628us    BUS A   TYPE0: BC to RT
        CMD1 1BC0 --> 03-R-30-00
        DATA 0101  0202  0303  0404  0505  0606  0707  0808
             0909  1010  1111  1212  1313  1414  1515  1616
             1717  1818  1919  2020  2121  2222  2323  2424
             2525  2626  2727  2828  2929  3030  3131  3232

        STA1 1800

Dev0 MSG #0001.  TIME = 00041324us    BUS B   TYPE1: RT to BC
        CMD1 1FC0 --> 03-T-30-00
        STA1 1800
        DATA 0101  0202  0303  0404  0505  0606  0707  0808
             0909  1010  1111  1212  1313  1414  1515  1616
             1717  1818  1919  2020  2121  2222  2323  2424
             2525  2626  2727  2828  2929  3030  3131  3232


Dev0 MSG #0002.  TIME = 00042020us    BUS A   TYPE1: RT to BC
        CMD1 1FC0 --> 03-T-30-00
        STA1 1800
        DATA 0101  0202  0303  0404  0505  0606  0707  0808
             0909  1010  1111  1212  1313  1414  1515  1616
             1717  1818  1919  2020  2121  2222  2323  2424
             2525  2626  2727  2828  2929  3030  3131  3232


Dev0 MSG #0003.  TIME = 00042720us    BUS A   TYPE0: BC to RT
        CMD1 1BC0 --> 03-R-30-00
        DATA 0101  0202  0303  0404  0505  0606  0707  0808
             0909  1010  1111  1212  1313  1414  1515  1616
             1717  1818  1919  2020  2121  2222  2323  2424
             2525  2626  2727  2828  2929  3030  3131  3232

        STA1 1800

Dev0 MSG #0004.  TIME = 00043416us    BUS B   TYPE1: RT to BC
        CMD1 1FC0 --> 03-T-30-00
        STA1 1800
        DATA 0101  0202  0303  0404  0505  0606  0707  0808
             0909  1010  1111  1212  1313  1414  1515  1616
             1717  1818  1919  2020  2121  2222  2323  2424
             2525  2626  2727  2828  2929  3030  3131  3232


Dev0 MSG #0005.  TIME = 00044112us    BUS A   TYPE1: RT to BC
        CMD1 1FC0 --> 03-T-30-00
        STA1 1800
        DATA 0101  0202  0303  0404  0505  0606  0707  0808
             0909  1010  1111  1212  1313  1414  1515  1616
             1717  1818  1919  2020  2121  2222  2323  2424
             2525  2626  2727  2828  2929  3030  3131  3232


Dev0 MSG #0006.  TIME = 00044814us    BUS A   TYPE0: BC to RT
        CMD1 1BC0 --> 03-R-30-00
        DATA 0101  0202  0303  0404  0505  0606  0707  0808
             0909  1010  1111  1212  1313  1414  1515  1616
```

```
          1717  1818  1919  2020  2121  2222  2323  2424
          2525  2626  2727  2828  2929  3030  3131  3232

       STA1 1800

Dev0 MSG #0007.  TIME = 00045510us    BUS B   TYPE1: RT to BC
       CMD1 1FC0 --> 03-T-30-00
       STA1 1800
       DATA 0101  0202  0303  0404  0505  0606  0707  0808
          0909  1010  1111  1212  1313  1414  1515  1616
          1717  1818  1919  2020  2121  2222  2323  2424
          2525  2626  2727  2828  2929  3030  3131  3232


Dev0 MSG #0008.  TIME = 00046206us    BUS A   TYPE1: RT to BC
       CMD1 1FC0 --> 03-T-30-00
       STA1 1800
       DATA 0101  0202  0303  0404  0505  0606  0707  0808
          0909  1010  1111  1212  1313  1414  1515  1616
          1717  1818  1919  2020  2121  2222  2323  2424
          2525  2626  2727  2828  2929  3030  3131  3232


Dev0 MSG #0009.  TIME = 00046906us    BUS A   TYPE0: BC to RT
       CMD1 1BC0 --> 03-R-30-00
       DATA 0101  0202  0303  0404  0505  0606  0707  0808
          0909  1010  1111  1212  1313  1414  1515  1616
          1717  1818  1919  2020  2121  2222  2323  2424
          2525  2626  2727  2828  2929  3030  3131  3232

       STA1 1800

Dev0 MSG #0010.  TIME = 00047602us    BUS B   TYPE1: RT to BC
       CMD1 1FC0 --> 03-T-30-00
       STA1 1800
       DATA 0101  0202  0303  0404  0505  0606  0707  0808
          0909  1010  1111  1212  1313  1414  1515  1616
          1717  1818  1919  2020  2121  2222  2323  2424
          2525  2626  2727  2828  2929  3030  3131  3232


Dev0 MSG #0011.  TIME = 00048298us    BUS A   TYPE1: RT to BC
       CMD1 1FC0 --> 03-T-30-00
       STA1 1800
       DATA 0101  0202  0303  0404  0505  0606  0707  0808
          0909  1010  1111  1212  1313  1414  1515  1616
          1717  1818  1919  2020  2121  2222  2323  2424
          2525  2626  2727  2828  2929  3030  3131  3232


Dev0 MSG #0012.  TIME = 00049000us    BUS A   TYPE0: BC to RT
       CMD1 1BC0 --> 03-R-30-00
       DATA 0101  0202  0303  0404  0505  0606  0707  0808
          0909  1010  1111  1212  1313  1414  1515  1616
          1717  1818  1919  2020  2121  2222  2323  2424
          2525  2626  2727  2828  2929  3030  3131  3232

       STA1 1800
```

```
Dev0 MSG #0013.  TIME = 00049696us    BUS B   TYPE1: RT to BC
      CMD1 1FC0 --> 03-T-30-00
      STA1 1800
      DATA 0101  0202  0303  0404  0505  0606  0707  0808
         0909  1010  1111  1212  1313  1414  1515  1616
         1717  1818  1919  2020  2121  2222  2323  2424
         2525  2626  2727  2828  2929  3030  3131  3232


Dev0 MSG #0014.  TIME = 00050392us    BUS A   TYPE1: RT to BC
      CMD1 1FC0 --> 03-T-30-00
      STA1 1800
      DATA 0101  0202  0303  0404  0505  0606  0707  0808
         0909  1010  1111  1212  1313  1414  1515  1616
         1717  1818  1919  2020  2121  2222  2323  2424
         2525  2626  2727  2828  2929  3030  3131  3232
```

Command ‘a’ commands the BC to continuously transmit a set of repeating messages every 100ms – see the QSG for a partial listing. Use command ‘x’ to stop the messages.

Command ‘l’ inserts three predefined low priority BC messages in on bus B into the message sequence while running the BC Async demo (command ‘a’). This will only occur once.

```
Dev0 MSG #1694.  TIME = 00070698us    BUS B   TYPE0: BC to RT
            CMD1 0822 --> 01-R-01-02
            DATA DEAD  BEEF
            STA1 0800

 Dev0 MSG #1695.  TIME = 00071054us    BUS B   TYPE1: RT to BC
            CMD1 0C2F --> 01-T-01-15
            STA1 0800
            DATA BBBB  0202  1414  0404  0505  0606  0707  0808
                 0909  1010  1111  1212  1313  1414  1515

 Dev0 MSG #1696.  TIME = 00071210us    BUS B   TYPE0: BC to RT
            CMD1 0825 --> 01-R-01-05
            DATA CAFE  C0DE  0303  0404  0505
            STA1 0800
```

Command ‘h’ inserts a predefined high priority BC message into the message sequence while running the BC Asyn demo similar to command ‘l’ but is repeatable. This shows the inserted message #1533 between two other messages.

```
Dev0 MSG #1532.  TIME = 00129598us    BUS A   TYPE2: RT to RT
            CMD1 182A --> 03-R-01-10
            CMD2 0C2A --> 01-T-01-10
            STA1 0800
            DATA BBBB  0202  1414  0404  0505  0606  0707  0808
                 0909  1010
            STA2 1800

HoltBCSendAsyncMsgHP returns 0
>
 Dev0 MSG #1533.  TIME = 00042566us    BUS B   TYPE0: BC to RT
```

```
            CMD1 0822 --> 01-R-01-02
            DATA DEAD  BEEF
            STA1 0800

 Dev0 MSG #1534.  TIME = 00098222us    BUS A   TYPE0: BC to RT
            CMD1 1822 --> 03-R-01-02
            DATA 0005  0002
            STA1 1800
```

Command ‘S’ displays the current SMT command and data buffer addresses.

This example shows the necessary steps to enable the BC and both RT’s in both devices and begin transmitting from both BC’s using command ‘a’ and ‘A’. Both devices are only available with a -2F card so with a -1F card there’s no need to enter the upper case commands and no Dev1 messages will be presented in the output shown below. Notice some messages are from Dev0 and some are from Dev1.

```
>> b
>c
>k (RTMT Demo)
>t (Traffic Enabled)
>B
>C
>K
RTMT Demo
>a
>A (this will be seen much later inter-mixed in the messages below)

With a -2F card a mixed of messages from Dev0 and Dev1 are displayed.

 Dev0 MSG #0283.  TIME = 00130428us    BUS A   TYPE2: RT to RT
            CMD1 182A --> 03-R-01-10
            CMD2 0C2A --> 01-T-01-10
            STA1 0800
            DATA BBBB  0202  1414  0404  0505  0606  0707  0808
                 0909  1010
            STA2 1800

 Dev1 MSG #0284.  TIME = 00090538us    BUS A   TYPE0: BC to RT
            CMD1 1822 --> 03-R-01-02
            DATA 0005  0002
            STA1 1800

 Dev1 MSG #0285.  TIME = 00090842us    BUS A   TYPE2: RT to RT
            CMD1 182A --> 03-R-01-10
            CMD2 0C2A --> 01-T-01-10
            STA1 0800
            DATA BBBB  0202  1414  0404  0505  0606  0707  0808
                 0909  1010
            STA2 1800

 Dev0 MSG #0286.  TIME = 00099052us    BUS A   TYPE0: BC to RT
            CMD1 1822 --> 03-R-01-02
            DATA 0005  0002
            STA1 1800

 Dev0 MSG #0287.  TIME = 00099356us    BUS A   TYPE2: RT to RT
            CMD1 182A --> 03-R-01-10
            CMD2 0C2A --> 01-T-01-10
            STA1 0800
```

```
DATA BBBB  0202  1414  0404  0505  0606  0707  0808
     0909  1010
STA2 1800
```

Note: Command ‘9’ (Only after reset) can be used to change the RT addresses of the four RT’s. The BC demos, ‘n’ and ‘a’ transmits messages to RT addresses 1 and 3. The BC message RT address is set in the BC message functions: bcAsync() and MajorMinorframe() in the demo code file “demo.c”. To alter these RT addresses in the BC command words the addresses need to be modified in the code.

Command ‘0’ toggles between Demo and User board. The default is “Demo” mode. Command ‘f’ reads and displays the states of the FPGA control signals going to both Dev0 and Dev1. The signals are listed in Table 1 in the next section. See the next section for an explanation of Demo and User mode.

```
Press 'M' for menu, or press any valid menu key. >> f

Dev 0 values:
 P_BC_TRIG = 0
 P_NMR = 1
 P_TEST = 0
 P_BCENA = 1
 P_NTRUM = 1
 P_RT1ENA = 1
 P_RT2ENA = 1
 P_INHIBIT = 0
 P_INPUTCONTROL demo/user mode = 0          0=demo, 1=user
 P_SPAREINPUT = 0
 P_MTPKTRDY = 1
 P_ACTIVE = 0
 P_RT1MC8 = 1
 P_RT2MC8 = 1
 P_READY  = 1
Dev 1 values:
 P_BC_TRIG = 0
 P_NMR = 1
 P_TEST = 0
 P_BCENA = 1
 P_NTRUM = 1
 P_RT1ENA = 1
 P_RT2ENA = 1
 P_INHIBIT = 0
 P_INPUTCONTROL demo/user mode= 0           0=demo, 1=user
 P_SPAREINPUT = 0
 P_MTPKTRDY = 1
 P_ACTIVE = 0
 P_RT1MC8 = 1
 P_RT2MC8 = 1
 P_READY  = 1
>
```

Hardware Description

The Holt Mini PCIe card features two Holt HI-2130 dual-redundant multi-terminals on a 50.95mm x 30mm form factor per the PCI Express Mini Card Electromechanical Specification. A Xilinx XC7A12T FPGA serves as the interface between two HI-2130s and the PCIe Gen2 bus. The host communicates with the card using the built-in PCIe drivers in the Linux OS once a PCIe link is established after power up.

FPGA HI-2130 Dev0 and Dev1 control signals

A common parallel 16-bit address and data bus is implemented in the FPGA to communicate to the two HI-2130 devices (Dev0/Dev1). These signals are shown on page 2 of the schematics.

The terminal control signals for both devices (HI-2130) are connected to FPGA I/O which are controlled by the host using a memory mapped software write function to access the I/O registers 0x8000 – 0x8013. Signals that are writable are set high by writing a 1 or a 0 to set it low. For example to set BCENA high for device 0: The Chan parameter specifies device 0 or device 1 by setting the parameter to 0 or 1.

```
HoltRegWrite(Chan, P_BCENA, 1);  // Chan=0 for Dev0.
```

Demo Mode vs. User Mode

For ease of software demonstration a Demo Mode is selected when location INPUT_CONTROL (0x800D) is set low and a User Mode is selected when it is high. The power up default is Demo Mode.

In Demo Mode some of the control signals are preset in their enabled states by default so the terminals are all enabled: BCENA =1, RTENA=1 and MTENA=1 etc.

In User Mode the user can set any combination of BC, RT or MT enable control signals to enable or disable any of these terminals of each device. In User Mode two control signals, BCENA and BCTRIG, are sourced from the J4 connector pin.

To select User Demo Mode or User Mode use the write function:

```
HoltRegWrite(Chan, INPUT_CONTROL, 0); // 0=demo mode (default)
HoltRegWrite(Chan, INPUT_CONTROL, 1); // 1=user mode
```

When demo software is configured for ‘Demo mode - default’ the BCENA0 and BCENA1 pins from the FPGA to the devices are settable by software. When in “User” mode both BC enables follow the J4 connector states. By default, both BC enables are disabled. To enable a BC the user must ground the BE Enable pin on the connector. The BCTRIG0 and BCTRIG1 inputs are provided but are not currently used in the demos.

Console menu command ‘0’ can be used to select between User and Demo mode. Console menu command ‘f’ reads all the signals from each device and displays the states of each signal.

Table 1 – Memory mapped I/O Device control signals

Device Signal	Source	Function	Address	Software read/write	Use
BC_TRIG	FPGA->Device	BC trigger pulse	0x8000	Read/Write	Set High then low (1/0)
NMR	FPGA	Resets Device	0x8002	Read/Write	Set Low then High
TEST	FPGA	Test pin state	0x8003	Read/Write	Set High for Test mode. See 6130 data sheet for test mode.
BCENA	FPGA	Enables BC	0x8004	Read/Write	Set high to enable BC
MTRUN	FPGA -> Device	Enables MT	0x8009	Read/Write	Set high to enable MT
RT1ENA	FPGA -> Device	Enables RT1	0x800A	Read/Write	Set High to Enable RT1
RT2ENA	FPGA -> Device	Enables RT2	0x800B	Read/Write	Set High to Enable RT2
TXINHA/TXINHB	J4 Connector	Inhibits bus	0x800C	Read only	Hard-wired to J4
INPUT_CONTROL	FPGA	Set Demo mode or User mode	0x800E	Read/Write	0=Demo mode (default) 1=User mode
SPARE_INPUT	J4 Connector	Spare input	0x800E	Read only	Readable Spare input.
MTPKTRDY	Device -> FPGA	Monitor Packet ready device output	0x800F	Read only	Optional read
ACTIVE	Device -> FPGA	Active state	0x8010	Read only	Optional read
RT1MC8	Device->FPGA	RT1 Mode Code 8	0x8011	Read only	Optional read
RT2MC8	Device->FPGA	RT2 Mode Code 8	0x8012	Read only	Optional read
READY	Device->FPGA	Indicates Ready state	0x8013	Read only	Optional read. Used by demo software to determine when devices are ready after MR.

J-TAG & FPGA Boot

The boot SPI flash memory is programmed at the factory but can be reprogrammed by the user with a DIGILENT JTAG-HS2 USB programming cable (not provided) using the Xilinx Vivado tool.

When the FPGA successfully boots from the preprogrammed SPI Flash, D2 LED will light up. To reboot the FPGA configuration sequence after reprogramming, press SW1 button. For normal operation this switch can be ignored.

The FPGA uses three signals: M0, M1 and M2 inputs for configuration options. M0 and M1 are hard wired so only M2 can be changed on this design. But default, a pull-down resistor keeps M2 low which configures the FPGA to boot from the SPI Flash. For reprogramming, J2 pin 8 (see Figure 3 – Board References) should be pulled high which will drive M2 high. For normal use, these connections don't need to be altered by the user, only if the user wants to customize the Verilog design and wants to reprogram the SPI boot Flash. Reprogramming instructions are provided later in this document.

Table 2 – FPGA boot options

Function	M2- M1 - M0	Notes
Direct FPGA programming	0 – 0 - 1	Programs FPGA (volatile)
Programs SPI Boot Flash (default configuration)	1 – 0 - 1	Programs SPI Boot Flash (non volatile) FPGA boots from Flash

There are two green LEDs driven by the FPGA to provide some initialization status. LED D2 lights up when the FPGA initialization sequence is complete and asserts the DONE_0 signal high. LED D10 lights up when the FPGA is in a configuration reset state and will turn off when the initialization is complete. Xilinx provides a complete technical guide on the FPGA configuration sequence. See Xilinx UG470.

HI-2130 Parallel Interfaces

A 16-bit parallel interface implemented in the FPGA is used to interface to both HI-2130 devices. The address and data bus and a few other signals are common between both devices. Unique chip select lines nCE0 and nCE1 are used to select between them.

Table 3 - HI-2130 Common Interface Signals to FPGA

HI-2130	FPGA BANK	PRIMARY CONNECTOR	COMMENTS
Address pins: [A15:A0]	14	-	Address bus
Data bus pins: [D15:D0]	14	-	Bidirectional data bus
nRE	14	-	Read strobe (Intel mode)
nWE	14	-	Write strobe (Intel mode)
MCLK50	34	-	Master 50MHz input clock from FPGA
MTTCLK	34	-	Optional clock, TBD
TTCLK	34	-	Optional clock, TBD

Table 4 - HI-2130 Device 0

HI-2130	FPGA BANK	J4 CONNECTOR	COMMENTS
nCS0	14	-	Chip Select 0
nMR0	34	-	Master Reset 0
nIRQ0	34	-	Interrupt output
TEST0	34	-	Test mode when asserted high.
RT1ENA0	34	-	RT1 Enable controlled by FPGA.
RT2ENA0	34	-	RT2 Enable controlled by FPGA
RT1MC80	34	-	RT1MC8 output, input to FPGA
RT2MC80	34	-	RT2MC8 output, input to FPGA
CH0BCENAB0 (BCENA)	34	YES	Default enabled by pull-up resistor. Can be set low by the connector or the FPGA can override it.
CH0BCTRIG (BCTRIG)	34	YES	Normally low. Can be pulsed at connector or the FPGA can override it.
MTRUN0	34	-	Input, controlled by FPGA.
READY0, ACTIVE0	34	-	Outputs, Inputs to FPGA. Function TBD.
MTPKTRDY0	34	-	Output, Not used or TBD.
TXINHA,TXINHB		YES	Both inhibit inputs connected to CH0INHIBIT0 on connector. Default is enabled. Assert low to disable both Transmitters.

Table 5 - HI-2130 Device 1

HI-2130	FPGA BANK	J4 CONNECTOR	COMMENTS
nCS1	14	-	Chip Select 1
nMR1	34	-	Master Reset 1
nIRQ1	34	-	Interrupt output
TEST1	34	-	Test mode when asserted high.
RT1ENA1	34	-	RT1 Enable controlled by FPGA.
RT2ENA1	34	-	RT2 Enable controlled by FPGA
RT1MC81	34	-	RT1MC8 output, input to FPGA
RT2MC81	34	-	RT2MC8 output, input to FPGA
CH1BCENAB1 (BCENA)	34	YES	Default enabled by pull-up resistor. Can be set low by the connector or the FPGA can override it.
CH1BCTRIG (BCTRIG)	34	YES	Normally low. Can be pulsed at connector or the FPGA can override it.
MTRUN1	34	-	Input, controlled by FPGA.
READY1, ACTIVE1	34	-	Outputs, Inputs to FPGA. Function TBD.
MTPKTRDY1	34	-	Output, Not used or TBD.
TXINHA,TXINHB	-	YES	Both inhibit inputs connected to CH1INHIBIT1 on the connector. Default is enabled. Assert high to disable both Transmitters.

J4 connector input pins: Transmit Inhibit, BCENAx, BCTRIG are ESD protected but the DC steady state voltage should not exceed 3.6V.

Power supply

The 3.3aux power from the PCIe connector powers a double buck converter that generates the 1V and 1V8 power rails for the FPGA. A 'power good1' signal is used to enable two linear voltage regulators that supply 1V and 1V2 to FPGA GBT PCIe transceiver rails. The 1V8 is used to turn on a MOSFET switch used to switch on a separate 3V3 supply to meet Xilinx power sequencing recommendations to power the SPI Flash, MEMS oscillator and the FPGA bank 14, 15 and 34 rails.

Table 6 – Power Supply Power

Supply Voltage	Schematic Name	Function	Maximum Capacity	Expected board usage
1V	VCCINT_1V	FPGA logic	800mA	200mA
1.8V	VCCAUX_1V8	FPGA Aux functions (JTAG)	800mA	100mA
3.3V	VCCO3V3	FPGA I/O rails, Flash, Osc and Misc.	1000mA	200mA
3.3V (PCIe conn.)	3V3aux	2 x HI-2130	1600mA	1553 messaging duty cycle dependant
1.5V (PCIe conn.)	MGTAVCC1V MGTAVCTT1V2	Provides clean power to the GBT (PCIe bus)	150mA 150mA	<100mA <100mA

Table 7 – Power and Thermal bench measurements transmitting into 69Ω

Active Channels 0=U7 1=U8	TX Duty Cycle %	ICC Amps (3.3V) 69 ohm load	U7 (2130) C° Temp. Board flat open-air	U8 (2130) C° Board flat open-air
0	100	1.07	70	56
0 & 1	100	1.86	82	78
0 & 1	100	1.86	50 with fan	45 with fan
0 & 1	100	1.78	81	78
0 & 1	50	1.02	65	63
0	50	.655	59	52
0	10	.34	49	45
0 & 1	10	.42		
0 & 1	IDLE	.26	46	44

Note: ICC will be less with typical 75Ω bus coupler impedance.

Table 8 – J4 Connector Pins

20 Pin	NAME	DESCRIPTION
1	CHANNEL 0 APOS	MIL-STD-1553 CH0 A+ (BUSA)
2	CHANNEL 0 ANEG	MIL-STD-1553 CH0 A- (nBUSA)
3	Chassis GND	Mounting screw – no other connection.
4	CHANNEL 0 BPOS	MIL-STD-1553 CH0 B+ (BUSB)
5	CHANNEL 0 BNEG	MIL-STD-1553 CH0 B- (nBUSB)
6	Chassis GND	Mounting screw – no other connection.
7	CHANNEL1 APOS	MIL-STD-1553 CH1 A+
8	CHANNEL 1 ANEG	MIL-STD-1553 CH1 A-
9	Chassis GND	Mounting screw – no other connection.
10	CHANNEL 1 BPOS	MIL-STD-1553 CH1 B+
11	CHANNEL 1 BNEG	MIL-STD-1553 CH1 B-
12	Chassis GND	Mounting screw – no other connection.
13	CH0INHBIT0	Channel 0 Transmit Inhibit. 10K pull-up. Connects to an inverter which Inverted direct connect to 2130 inhibit pins.
14	CH0 BCENAB	Channel 0 BC Enable. 10K pull-up.
15	CH0 BCTRIG	Channel 0 BC Trigger. 10K pull-up.
16	SPARE INPUT	Unused.
17	CH1INHBIT1	Channel 0 Transmit Inhibit. 10K pull-up. Inverted direct connect to 2130 inhibit pins.
18	CH1 BCENAB	Channel 1 BC Enable. 10K pull-up.
19	CH1 BCTRIG	Channel 1 BC Trigger. 10K pull-up.
20	Logic GND	

Table 9– Card Test points

Test Points	Signal	Schematic Page	Schematic page, Function
TP1	ACTIVE0, Device0 HI-2130	3	Asserts high during any BC/RT/SMT command
TP2	Device 0 APOS	3	APOS (BUSA)
TP3	Device 0 ANEG	3	ANEG(nBUSA)
TP4	Device 0 BPOS	3	BPOS(BUSB)
TP5	Device B BNEG	3	BNEG(nBUSB)
TP6	ACTIVE1, Device1 HI-2130	4	Asserts high during any BC/RT/SMT command
TP7	Device 1 APOS	4	APOS (BUSA)
TP8	Device 1 ANEG	4	ANEG(nBUSA)
TP9	Device 1 BPOS	4	BPOS(BUSB)
TP10	Device 1 BNEG	4	BNEG(nBUSB)
TP11	FPGA GPIO	5	Available for user implementation
TP12	ADC 12-bit	5	Optional ADC, user implementation

TP1 and TP6 (feed-through only) are the HI-2130 ACTIVE outputs which are useful when checking out new software or any other changes to the design. ACTIVE asserts high during any BC, RT or SMT command and serves as a good starting point to check when debugging changes to the software or FPGA design. See board references for test point board placement.

System Clocks

A 100MHz PCIe system clock is input to the FPGA from the PCIe connector. This clock signal is input to the GBT clock input pair required for PCIe and also serves as the general clock for the FPGA logic.

An optional 100MHz clock is also provided on-board by U13 MEMS oscillator module. This clock is currently not used.

GPIO

A single GPIO pin from the primary connector is resistor isolated and ESD protected and routed to a FPGA pin. This could be an input or output – user defined.

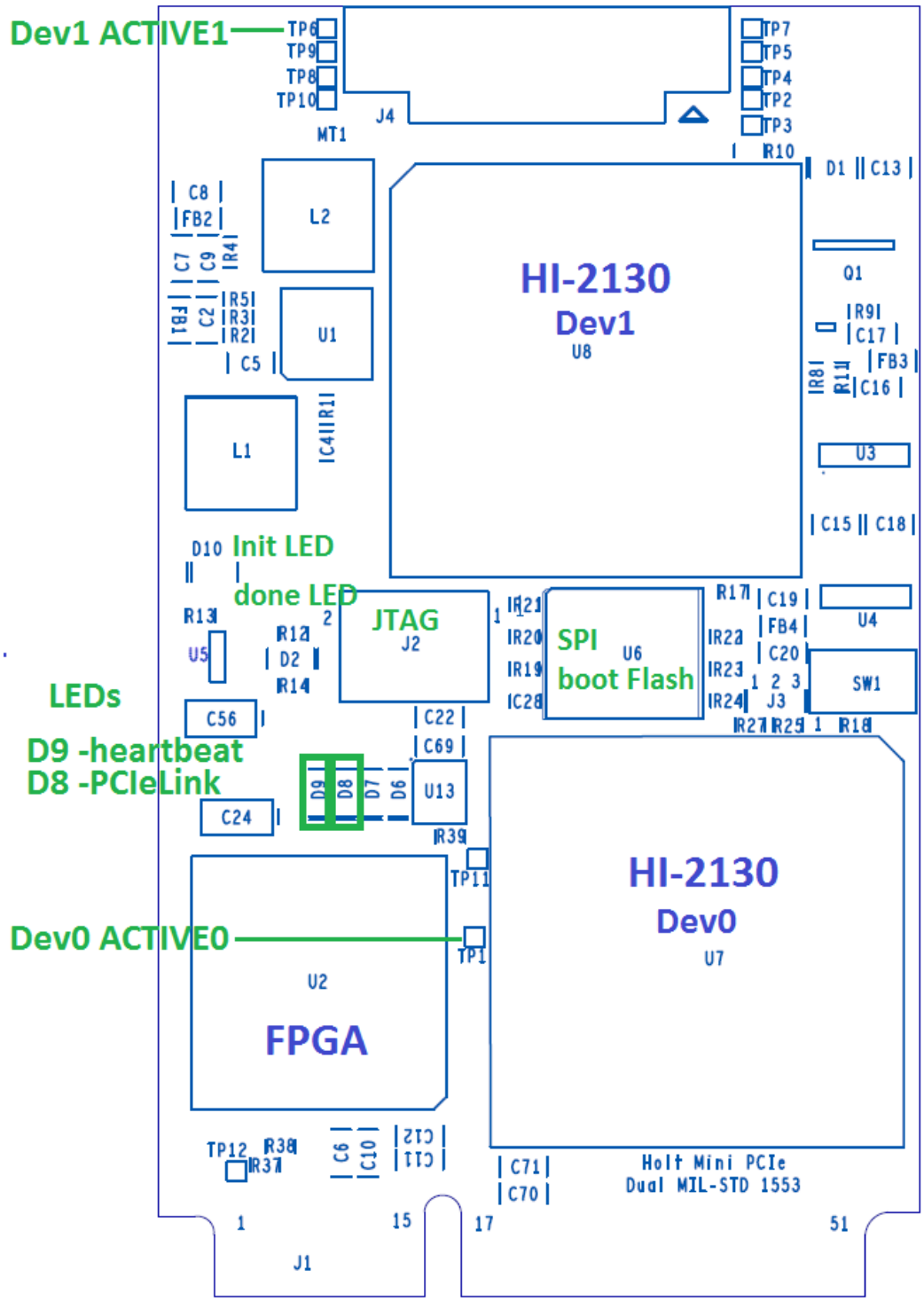
ADC 12-bit converter

An ADC input is provided on a test point but this feature is not implemented in the current design. The user has the option to implement this if there is a requirement.

FPGA Spare Pins

See the schematic for spare pins.

Figure 3 – Board References



Summary

This technical guide summarized the steps necessary to install the Holt Mini PCIe Eclipse project and import the project into Eclipse to allow building the projects and rerunning them. See technical guide AN-MPCIeVivado Technical Guide for information on the Xilinx FPGA design.

Holt Integrated Circuits, Inc.
PCB P/N: HV047
Rev.A

Bill of Materials
EV-2130mPCIe-1F
Evaluation Board

Item	Qty	Description	Reference	Digikey P/N	Mfg P/N
1	1	PCB, Bare, Evaluation Board	N/A		N/A
2	2	Cap Cer 10pF 10V COG/NPO 0402 SMD	C5,C9	399-8939-1-ND	Kemet C0402C100C8GACTU
3	2	Cap Cer 0.01uF 16V X7R 0402 SMD	C6,C10	399-6297-1-ND	Kemet C0402T103K4RACTU
4	22	Cap Cer 0.1uF 10V X7S 0201 SMD	C4,C14,C21,C23,C28,C35,C37,C39,C40,C41,C42,C44,C46,C47,C48,C49,C50,C54,C60,C65,C66,C67	490-14450-1-ND	Murata GRM033C71A104KE14D
5	1	Cap Cer 0.1uF 10V X7R 0402 SMD	C22	478-7891-1-ND	AVX 0402ZC104KAT2A
6	4	Cap Cer 0.22uF 10V X5R 0402 SMD	C11,C12,C70,C71	490-3910-1-ND	Murata GRM155R61A224KE19D
7	12	Cap Cer 0.47uF 25V X5R 0402 SMD	C27,C29,C30,C31,C33,C34,C52,C59,C62,C63,C64,C69	490-12270-1-ND	Murata GRT155R61E474ME01D
8	4	Cap Cer 1uF 16V X5R 0402 SMD	C15,C16,C18,C19	490-12255-1-ND	Murata GRT155R61C105KE01D
9	4	Cap Cer 2.2uF 16V X5R 0402 SMD	C3,C8,C17,C20	445-9076-1-ND	TDK C1005X5R1C225M050BC
10	6	Cap Cer 4.7uF 10V X5R 0402 SMD	C26,C32,C55,C58,C61,C68	445-13820-1-ND	TDK C1005X5R1A475K050BC
11	4	Cap Cer 10uF 10V X5R 0402 SMD	C1,C2,C7,C13	490-13211-1-ND	Murata GRJ155R60J106ME11D
12	10	Cap Tant 47uF 20% 6.3V 300mOhm 0603	C24,C25,C36,C38,C43,C45,C51,C53,C56,C57	478-9699-1-ND	AVX F380J476MMAAXEH3
13	1	Res 100, 1%, 1/20W 0201 SMD	R7	P122654CT-ND	Panasonic ERJ-1GNF1000C
14	1	Res 226K 1%, 1/20W 0201 SMD	R2	P122842CT-ND	Panasonic ERJ-1GNF2263C
15	1	Res 255K 1%, 1/20W 0201 SMD	R5	P122874CT-ND	Panasonic ERJ-1GNF2553C
16	1	ReS 340K 1%, 1/20W 0201 SMD	R3	P122949CT-ND	Panasonic ERJ-1GNF3403C
17	2	Res 511K 1%,1/20W 0201 SMD	R1,R4	P123057CT-ND	Panasonic ERJ-1GNF5113C
18	1	Res 0, 5%, 1/20W 0201 SMD	R40	P15979CT-ND	Panasonic ERJ-1GN0R00C
19	1	Res 10, 5%, 1/20W 0201 SMD	R39	P123219CT-ND	Panasonic ERJ-1GNJ100C
20	1	Res 15, 5%, 1/20W 0201 SMD	R6	P123244CT-ND	Panasonic ERJ-1GNJ150C
21	1	Res 220, 5%, 1/20W 0201 SMD	R14	P123276CT-ND	Panasonic ERJ-1GNJ221C
22	3	Res 330, 5%,1/20W 0201 SMD	R8,R12,R28	P123304CT-ND	Panasonic ERJ-1GNJ331C
23	5	Res 470, 5%, 1/20W 0201 SMD	R10,R41,R42,R43,R44	P123332CT-ND	Panasonic ERJ-1GNJ471C
24	5	ReS 1K, 5%, 1/20W 0201 SMD	R15,R25,R26,R27,R38	P123221CT-ND	Panasonic ERJ-1GNJ102C
25	1	Res 3.3K, 5%, 1/20W 0201 SMD	R9	P123305CT-ND	Panasonic ERJ-1GNJ332C
26	8	Res 4.7K, 5%, 1/20W 0201 SMD	R13,R18,R19,R20,R21,R22,R23,R24	P4.7AECT-ND	Panasonic ERJ-1GNJ472C
27	1	Res 9.1K, 5%, 1/20W 0201 SMD	R37	P123382CT-ND	Panasonic ERJ-1GNJ912C
28	11	Res 10K, 5%, 1/20W 0201 SMD	R11,R16,R17,R29,R30,R31,R32,R33,R34,R35,R36	P122414CT-ND	Panasonic ERJ-1GNF1002C
29	12	Test Points	TP1-TP12 (DNI)	None	None
30	6	Ferrite 120 Ohm@100Mhz,0402 SMD	FB1,FB2,FB3,FB4,FB5,FB6	490-5192-1-ND	Murata BLM15BD121SN1D
31	2	Inductor fixed 1.5uH,1.75A .0.47 mOhm	L1,L2	732-1005-1-ND	Wurth EL 744031001

Holt Integrated Circuits, Inc.
PCB P/N: HV047
Rev.A

Bill of Materials
EV-2130mPCIe-1F
Evaluation Board

32	7	Green Led, Clear 0402 SMD	D1,D2,D6,D7,D8,D9,D10	1497-1219-1-ND	SunLED XZVG68W-2
33	1	Conn 8-Pin 0.8mm Micro Socket	J2	CLE-104-01-G-DV	Samtec CLE-104-01-G-DV
34	1	Through Hole 3-Pin Connector	J3 (DNI)	None	None
35	1	Conn Zero Insertion 20-pos,0.5mm FFC	J4	SAM14910CT-ND	Samtec ZF5S-20-01-T-WT-TR
36	1	Trans Mosfet P-Ch 30V 5A SOT23	Q1	SI2347DS-T1-GE3CT-ND	Vishay SI2347DS-T1-GE3
37	1	Trans NPN 40V 0.2A SOT416	Q2	MMBT3904TT1GOSCT-ND	On MMBT3904TT1G
38	2	TVS Diode 3.6V 8.8V 10USON	D4,D5	296-43875-1-ND	TI TPD4E02B04DQAR
39	1	IC Reg Linear 1V 150mA SOT23-5	U3	497-6871-1-ND	ST LD39015M10R
40	1	IC Reg Linear 1.2V 150mA SOT23-5	U4	497-6872-1-ND	ST LD39015M12R
41	2	IC Inverter 1CH 1-INP SC70-5	U5,U11	296-11600-1-ND	TI SN74LVC1G04DCKR
42	3	IC Inverter 2CH 2-INP SC70-6	U9,U10,U12	296-13262-1-ND	TI SN74LVC2G04DCKR
43	1	Embedded FPGA Artix-7 238-Pin .5mm	U2	XC7A12T-2CPG238I-ND	Xlinx XC7A12T-2CPG238I
44	1	IC Flash 64M SPI 108Mhz 8-WSON	U6	S25FL064LABNFM010-ND	Cypress S25FL064LABNFM010
45	1	IC Reg Buck Adj 0.8A TRPL 16-WFQFN	U1	LTC3545IUD#PBF-ND	Linear LTC3545IUD#PBF
46	1	HI-2130 121-BGA LBxx Low Profile	U7	HI-2130 LBxx	Holt HI-2130 LBxx
47	1	Mems Osc XO 100Mhz LVCMOS	U13	1473-30420-1-ND	SiTime SIT8918BE-73-33E-100G
48	1	SW SPST Momentary	SW1	SW1020CT-ND	Omron B3U-1000P

Holt Inc.
PCB P/N: HV051

Bill of Materials
mPCIe_breakout-1F
Rev.B

Item	Qty	Description	Reference	DigiKey	Mfr P/N
1	1	PCB, Bare, Eval Board	N/A	--------	NewTek # 13989
2	1	Conn FFC Bottom 5mm R/A	J2A	SAM14910CT-ND	Samtec ZF5S-20-01-T-WT-TR
3	1	Cable FFC 20 Pos 0.5mm 5" Long	J2B	WM11409-ND	Molex 0152660213
4	1	24-Inch Triax Plug Cable	CH0(AB)	None	MilesTek CA-2014-48
5	1	Aluminum Block 0.75"x2.5"x0.25"	Tie Block	None	OnlineMetals # 1142
6	3	Hex Nut 3/16" Steel 4-40	None	36-4694-ND	Keystone 4694
7	3	Washer Split Lock #4	None	36-4693-ND	Keystone 4693
8	3	Machine Screw Flat #4-40	None	36-9502-ND	Keystone 9502
9	4	Rubber Foot, Bumpon Black , .312 X.200 H	Four corners	SJ5746-0-ND	3M SJ61A1

Holt Integrated Circuits, Inc.
PCB P/N: HV047
Rev.A

Bill of Materials
EV-2130mPCIe-2F
Evaluation Board

Item	Qty	Description	Reference	Digikey P/N	Mfg P/N
1	1	PCB, Bare, Evaluation Board	N/A		N/A
2	2	Cap Cer 10pF 10V COG/NPO 0402 SMD	C5,C9	399-8939-1-ND	Kemet C0402C100C8GACTU
3	2	Cap Cer 0.01uF 16V X7R 0402 SMD	C6,C10	399-6297-1-ND	Kemet C0402T103K4RACTU
4	22	Cap Cer 0.1uF 10V X7S 0201 SMD	C4,C14,C21,C23,C28,C35,C37,C39,C40,C41,C42,C44,C46,C47,C48,C49,C50,C54,C60,C65,C66,C67	490-14450-1-ND	Murata GRM033C71A104KE14D
5	1	Cap Cer 0.1uF 10V X7R 0402 SMD	C22	478-7891-1-ND	AVX 0402ZC104KAT2A
6	4	Cap Cer 0.22uF 10V X5R 0402 SMD	C11,C12,C70,C71	490-3910-1-ND	Murata GRM155R61A224KE19D
7	12	Cap Cer 0.47uF 25V X5R 0402 SMD	C27,C29,C30,C31,C33,C34,C52,C59,C62,C63,C64,C69	490-12270-1-ND	Murata GRT155R61E474ME01D
8	4	Cap Cer 1uF 16V X5R 0402 SMD	C15,C16,C18,C19	490-12255-1-ND	Murata GRT155R61C105KE01D
9	4	Cap Cer 2.2uF 16V X5R 0402 SMD	C3,C8,C17,C20	445-9076-1-ND	TDK C1005X5R1C225M050BC
10	6	Cap Cer 4.7uF 10V X5R 0402 SMD	C26,C32,C55,C58,C61,C68	445-13820-1-ND	TDK C1005X5R1A475K050BC
11	4	Cap Cer 10uF 10V X5R 0402 SMD	C1,C2,C7,C13	490-13211-1-ND	Murata GRJ155R60J106ME11D
12	10	Cap Tant 47uF 20% 6.3V 300mOhm 0603	C24,C25,C36,C38,C43,C45,C51,C53,C56,C57	478-9699-1-ND	AVX F380J476MMAAXEH3
13	1	Res 100, 1%, 1/20W 0201 SMD	R7	P122654CT-ND	Panasonic ERJ-1GNF1000C
14	1	Res 226K 1%, 1/20W 0201 SMD	R2	P122842CT-ND	Panasonic ERJ-1GNF2263C
15	1	Res 255K 1%, 1/20W 0201 SMD	R5	P122874CT-ND	Panasonic ERJ-1GNF2553C
16	1	ReS 340K 1%, 1/20W 0201 SMD	R3	P122949CT-ND	Panasonic ERJ-1GNF3403C
17	2	Res 511K 1%,1/20W 0201 SMD	R1,R4	P123057CT-ND	Panasonic ERJ-1GNF5113C
18	1	Res 0, 5%, 1/20W 0201 SMD	R40	P15979CT-ND	Panasonic ERJ-1GN0R00C
19	1	Res 10, 5%, 1/20W 0201 SMD	R39	P123219CT-ND	Panasonic ERJ-1GNJ100C
20	1	Res 15, 5%, 1/20W 0201 SMD	R6	P123244CT-ND	Panasonic ERJ-1GNJ150C
21	1	Res 220, 5%, 1/20W 0201 SMD	R14	P123276CT-ND	Panasonic ERJ-1GNJ221C
22	3	Res 330, 5%,1/20W 0201 SMD	R8,R12,R28	P123304CT-ND	Panasonic ERJ-1GNJ331C
23	5	Res 470, 5%, 1/20W 0201 SMD	R10,R41,R42,R43,R44	P123332CT-ND	Panasonic ERJ-1GNJ471C
24	5	ReS 1K, 5%, 1/20W 0201 SMD	R15,R25,R26,R27,R38	P123221CT-ND	Panasonic ERJ-1GNJ102C
25	1	Res 3.3K, 5%, 1/20W 0201 SMD	R9	P123305CT-ND	Panasonic ERJ-1GNJ332C
26	8	Res 4.7K, 5%, 1/20W 0201 SMD	R13,R18,R19,R20,R21,R22,R23,R24	P4.7AECT-ND	Panasonic ERJ-1GNJ472C
27	1	Res 9.1K, 5%, 1/20W 0201 SMD	R37	P123382CT-ND	Panasonic ERJ-1GNJ912C
28	11	Res 10K, 5%, 1/20W 0201 SMD	R11,R16,R17,R29,R30,R31,R32,R33,R34,R35,R36	P122414CT-ND	Panasonic ERJ-1GNF1002C
29	12	Test Points	TP1-TP12 (DNI)	None	None
30	6	Ferrite 120 Ohm@100Mhz,0402 SMD	FB1,FB2,FB3,FB4,FB5,FB6	490-5192-1-ND	Murata BLM15BD121SN1D
31	2	Inductor fixed 1.5uH,1.75A .0.47 mOhm	L1,L2	732-1005-1-ND	Wurth EL 744031001

Holt Integrated Circuits, Inc.
PCB P/N: HV047
Rev.A

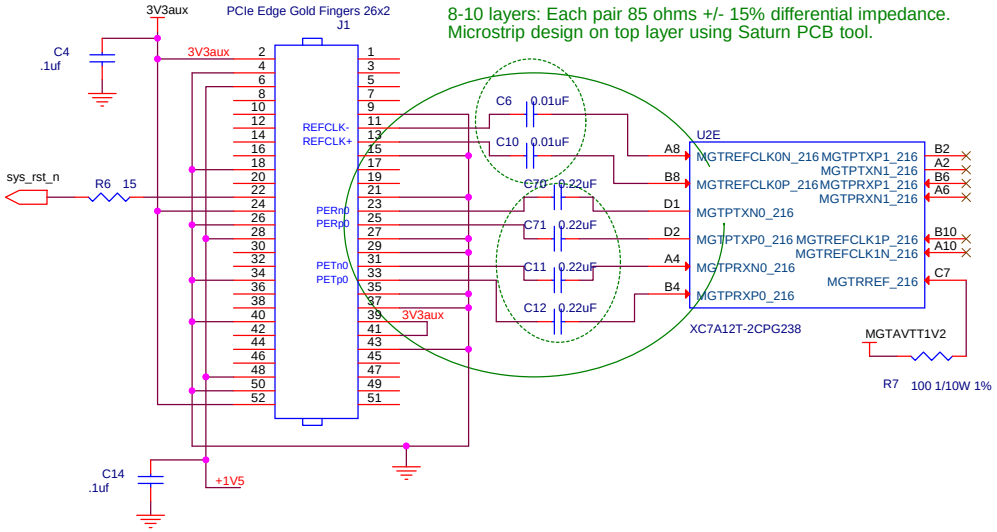
Bill of Materials
EV-2130mPCIe-2F
Evaluation Board

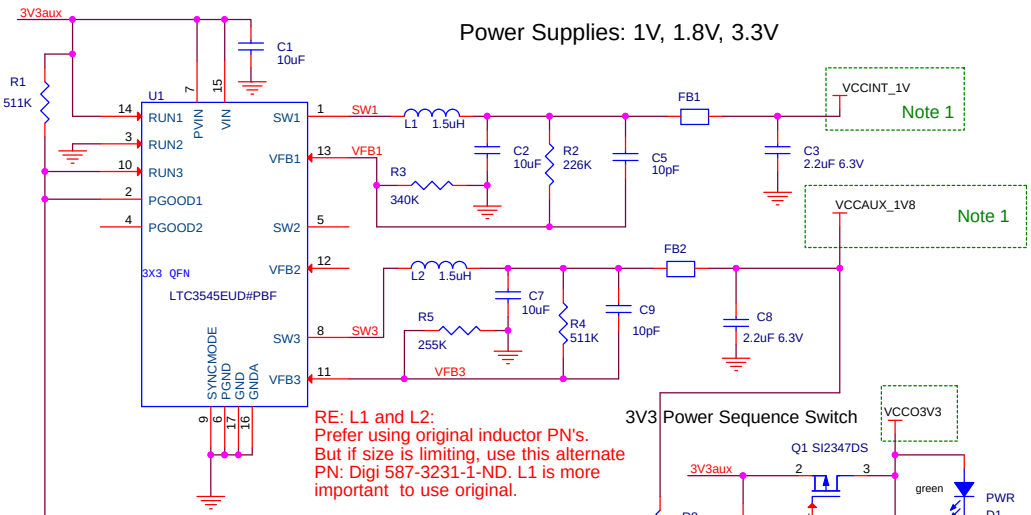
32	7	Green Led, Clear 0402 SMD	D1,D2,D6,D7,D8,D9,D10	1497-1219-1-ND	SunLED XZVG68W-2
33	1	Conn 8-Pin 0.8mm Micro Socket	J2	CLE-104-01-G-DV	Samtec CLE-104-01-G-DV
34	1	Through Hole 3-Pin Connector	J3 (DNI)	None	None
35	1	Conn Zero Insertion 20-pos,0.5mm FFC	J4	SAM14910CT-ND	Samtec ZF5S-20-01-T-WT-TR
36	1	Trans Mosfet P-Ch 30V 5A SOT23	Q1	SI2347DS-T1-GE3CT-ND	Vishay SI2347DS-T1-GE3
37	1	Trans NPN 40V 0.2A SOT416	Q2	MMBT3904TT1GOSCT-ND	On MMBT3904TT1G
38	2	TVS Diode 3.6V 8.8V 10USON	D4,D5	296-43875-1-ND	TI TPD4E02B04DQAR
39	1	IC Reg Linear 1V 150mA SOT23-5	U3	497-6871-1-ND	ST LD39015M10R
40	1	IC Reg Linear 1.2V 150mA SOT23-5	U4	497-6872-1-ND	ST LD39015M12R
41	2	IC Inverter 1CH 1-INP SC70-5	U5,U11	296-11600-1-ND	TI SN74LVC1G04DCKR
42	3	IC Inverter 2CH 2-INP SC70-6	U9,U10,U12	296-13262-1-ND	TI SN74LVC2G04DCKR
43	1	Embedded FPGA Artix-7 238-Pin .5mm	U2	XC7A12T-2CPG238I-ND	Xlinx XC7A12T-2CPG238I
44	1	IC Flash 64M SPI 108Mhz 8-WSON	U6	S25FL064LABNFM010-ND	Cypress S25FL064LABNFM010
45	1	IC Reg Buck Adj 0.8A TRPL 16-WFQFN	U1	LTC3545IUD#PBF-ND	Linear LTC3545IUD#PBF
46	2	HI-2130 121-BGA LBxx Low Profile	U7,U8	HI-2130 LBxx	Holt HI-2130 LBxx
47	1	Mems Osc XO 100Mhz LVCMOS	U13	1473-30420-1-ND	SiTime SIT8918BE-73-33E-100G
48	1	SW SPST Momentary	SW1	SW1020CT-ND	Omron B3U-1000P

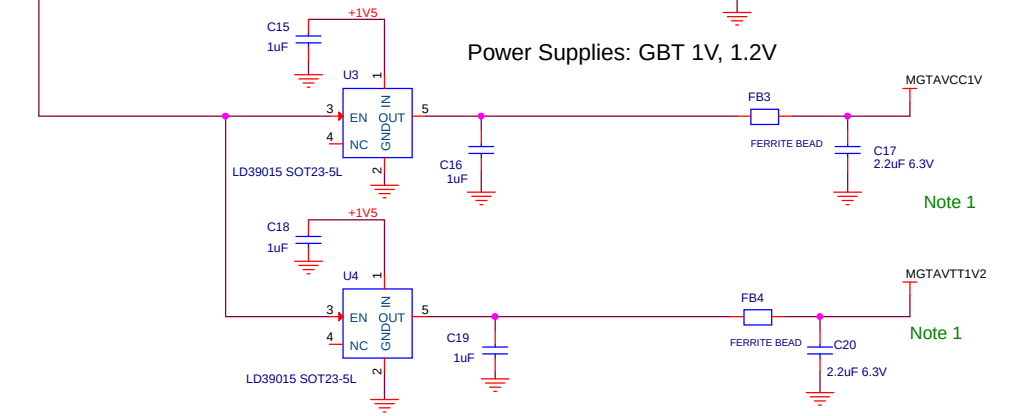
Holt Inc.
PCB P/N: HV051

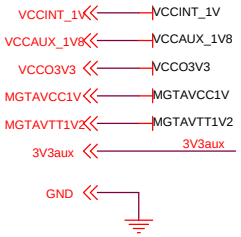
Bill of Materials
mPCIe_breakout-2F
Rev.B

Item	Qty	Description	Reference	DigiKey	Mfr P/N
1	1	PCB, Bare, Eval Board	N/A	--------	NewTek # 13989
2	1	Conn FFC Bottom 5mm R/A	J2A	SAM14910CT-ND	Samtec ZF5S-20-01-T-WT-TR
3	1	Cable FFC 20 Pos 0.5mm 5" Long	J2B	WM11409-ND	Molex 0152660213
4	2	24-Inch Triax Plug Cable	CH0(AB),CH1(AB)	None	MilesTek CA-2014-48
5	1	Aluminum Block 0.75"x2.5"x0.25"	Tie Block	None	OnlineMetals # 1142
6	3	Hex Nut 3/16" Steel 4-40	None	36-4694-ND	Keystone 4694
7	3	Washer Split Lock #4	None	36-4693-ND	Keystone 4693
8	3	Machine Screw Flat #4-40	None	36-9502-ND	Keystone 9502
9	4	Rubber Foot, Bumpon Black , .312	Four corners	SJ5746-0-ND	3M SJ61A1





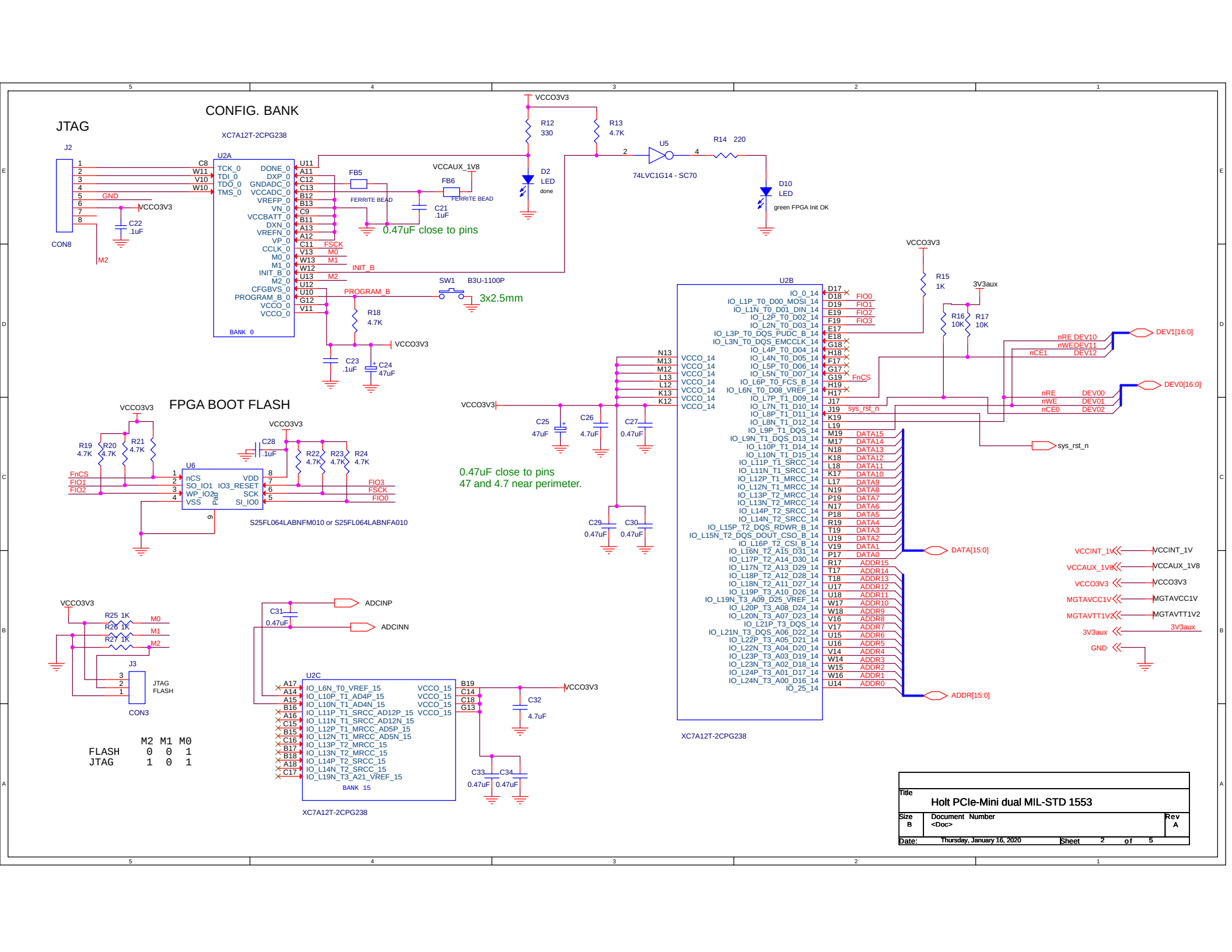


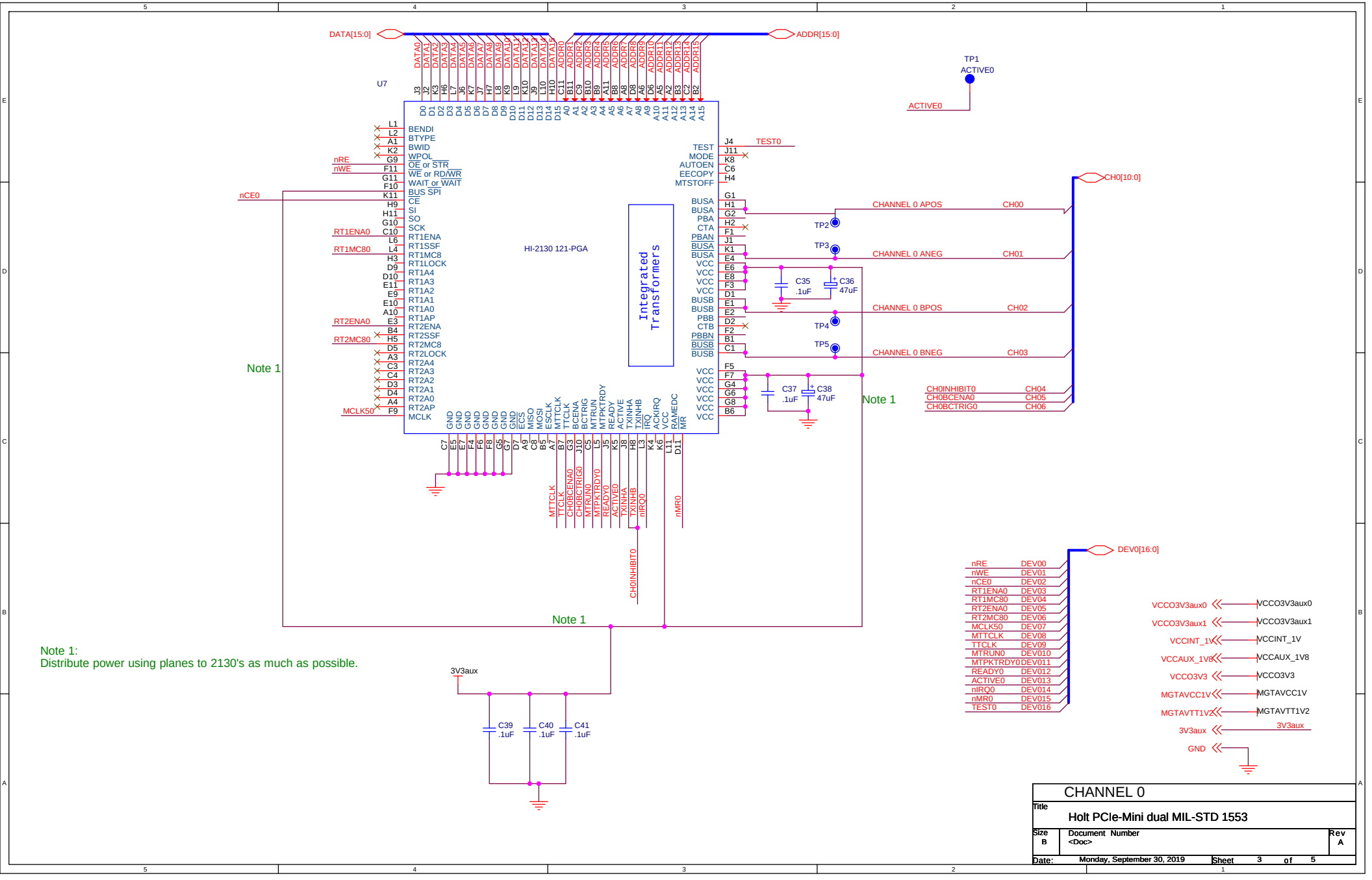


Where possible parts were selected
for bottom height requirement < 1.35mm
to allow back side placement.

Note 1:
Distribute power over planes to where they need to go under the
FPGA as much as possible.

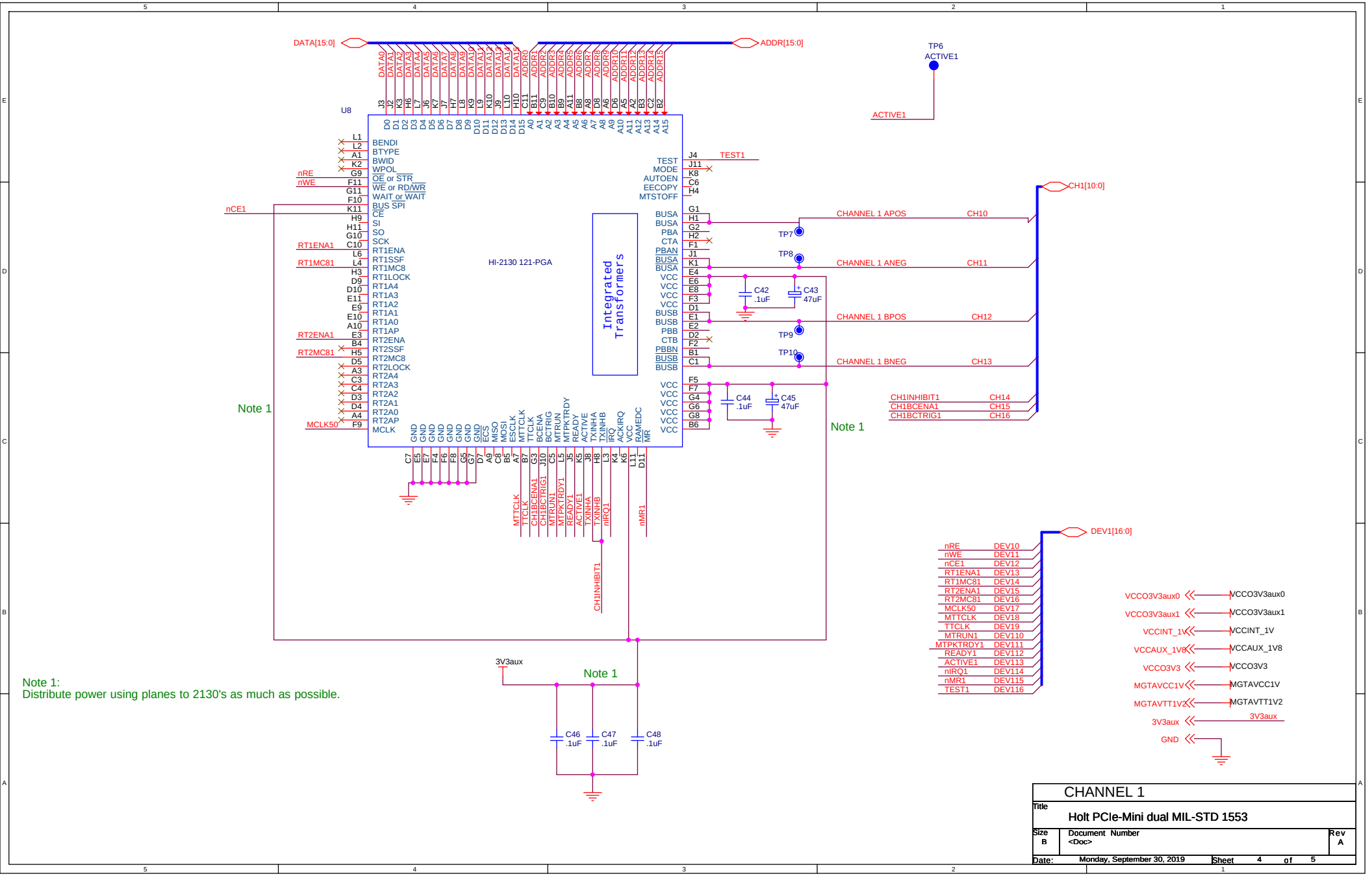
Title	Holt PCIe-Mini dual MIL-STD 1553	
Size B	Document Number <Doc>	Rev A
Date: Thursday, January 16, 2020	Sheet 1 of 5	


CONFIG. BANK
JTAG
XC7A12T-2CPG238
J2
CON8
U2A
BANK 0
VCCO3V3
VCCAUX_1V8
FB5
FB6
FERRITE BEAD
C21 .1uF
0.47uF close to pins
R12 330
R13 4.7K
R14 220
U5
74LVC1G14 - SC70
D2 LED done
D10 LED green FPGA Init OK
SW1 B3U-1100P
3x2.5mm
PROGRAM_B
INIT_B
R18 4.7K
C22 .1uF
C23 .1uF
C24 47uF
M2
FSCK
M0
M1
GND
FPGA BOOT FLASH
U6
S25FL064LABNFM010 or S25FL064LABNFA010
R19 4.7K
R20 4.7K
R21 4.7K
R22 4.7K
R23 4.7K
R24 4.7K
C28 .1uF
FnCS
FIO1
FIO2
FIO3
FSCK
FIO0
U2B
XC7A12T-2CPG238
R15 1K
R16 10K
R17 10K
3V3aux
nRE DEV10
nWE DEV11
nCE1 DEV12
DEV1[16:0]
nRE DEV00
nWE DEV01
nCE0 DEV02
DEV0[16:0]
sys_rst_n
DATA[15:0]
ADDR[15:0]
C25 47uF
C26 4.7uF
C27 0.47uF
C29 0.47uF
C30 0.47uF
0.47uF close to pins
47 and 4.7 near perimeter.
R25 1K
R26 1K
R27 1K
J3
JTAG FLASH
CON3
M2 M1 M0
FLASH 0 0 1
JTAG 1 0 1
ADCINP
ADCINN
C31 0.47uF
U2C
BANK 15
XC7A12T-2CPG238
C32 4.7uF
C33 0.47uF
C34 0.47uF
VCCINT_1V
VCCAUX_1V8
VCCO3V3
MGTAVCC1V
MGTAVTT1V2
3V3aux
GND
Title
Holt PCIe-Mini dual MIL-STD 1553
Size B
Document Number <Doc>
Rev A
Date: Thursday, January 16, 2020
Sheet 2 of 5



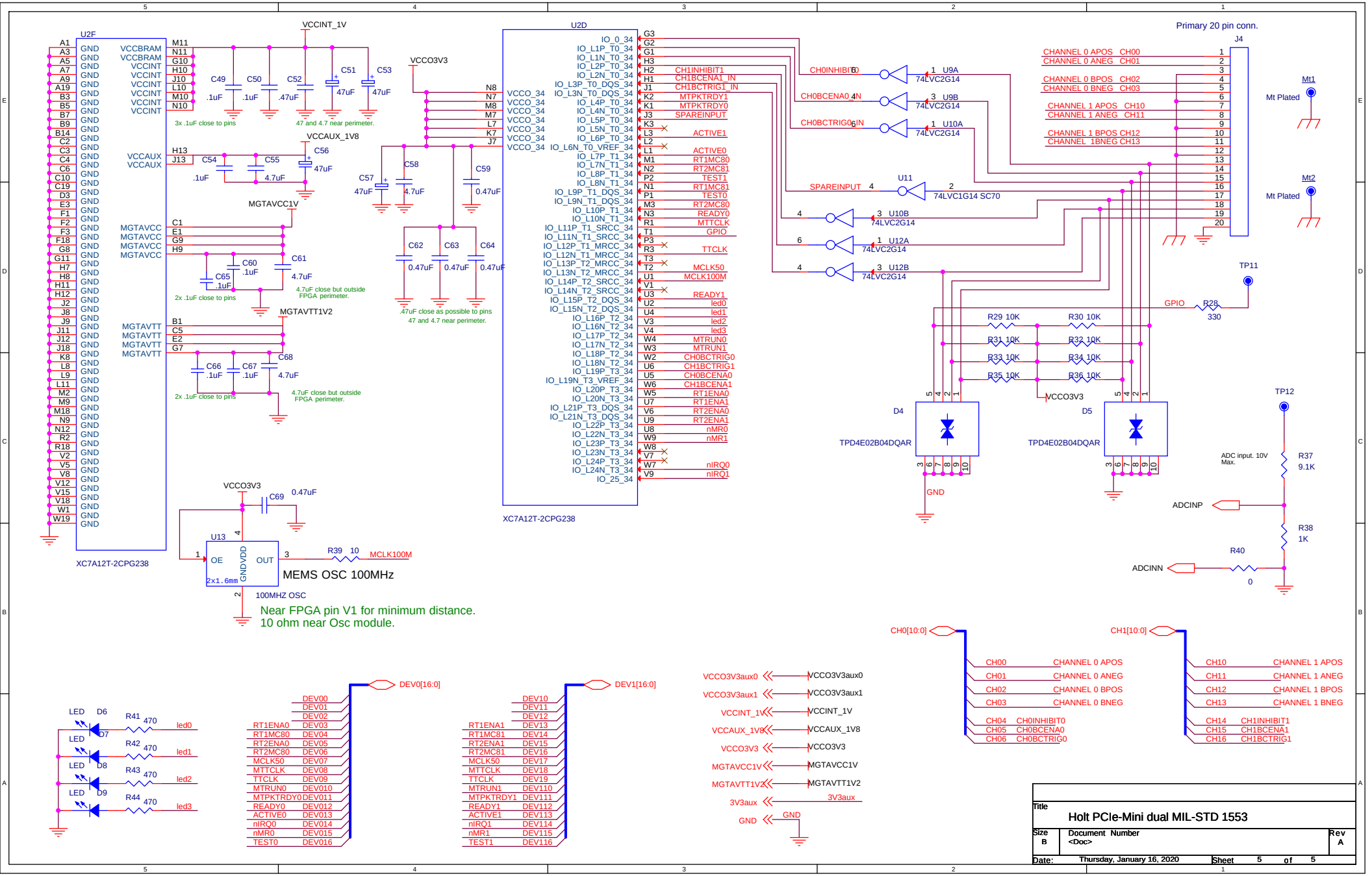
Note 1:
Distribute power using planes to 2130's as much as possible.

Title	CHANNEL 0
	Holt PCIe-Mini dual MIL-STD 1553
Size B	Document Number <Doc>
Rev A	
Date:	Monday, September 30, 2019
Sheet	3 of 5



Note 1:
Distribute power using planes to 2130's as much as possible.

CHANNEL 1			
Title: Holt PCIe-Mini dual MIL-STD 1553			
Size: B	Document Number: <Doc>		Rev: A
Date: Monday, September 30, 2019		Sheet 4 of 5	


U2F
XC7A12T-2CPG238
VCCINT_1V
VCCO3V3
VCCAUX_1V8
MGTAVCC1V
MGTAVTT1V2
3x .1uF close to pins
47 and 4.7 near perimeter.
2x .1uF close to pins
4.7uF close but outside FPGA perimeter.
U2D
XC7A12T-2CPG238
.47uF close as possible to pins
47 and 4.7 near perimeter.
Primary 20 pin conn.
J4
Mt1
Mt2
Mt Plated
CHANNEL 0 APOS CH00
CHANNEL 0 ANEG CH01
CHANNEL 0 BPOS CH02
CHANNEL 0 BNEG CH03
CHANNEL 1 APOS CH10
CHANNEL 1 ANEG CH11
CHANNEL 1 BPOS CH12
CHANNEL 1BNEG CH13
U9A
U9B
U10A
U10B
U12A
U12B
U11
74LVC2G14
74LVC1G14 SC70
CH0INHIBIT0
CH0BCENA0
CH0BCTRIG0
SPAREINPUT
TP11
GPIO
R28
330
R29 10K
R30 10K
R31 10K
R32 10K
R33 10K
R34 10K
R35 10K
R36 10K
D4
D5
TPD4E02B04DQAR
TP12
ADC input. 10V Max.
R37
9.1K
ADCINP
R38
1K
R40
0
ADCINN
U13
OE
GNDVDD
OUT
2x1.6mm
MEMS OSC 100MHz
100MHZ OSC
R39 10
MCLK100M
Near FPGA pin V1 for minimum distance.
10 ohm near Osc module.
LED D6
LED D7
LED D8
LED D9
R41 470
R42 470
R43 470
R44 470
led0
led1
led2
led3
DEV0[16:0]
DEV1[16:0]
CH0[10:0]
CH1[10:0]
CH00 CHANNEL 0 APOS
CH01 CHANNEL 0 ANEG
CH02 CHANNEL 0 BPOS
CH03 CHANNEL 0 BNEG
CH04 CH0INHIBIT0
CH05 CH0BCENA0
CH06 CH0BCTRIG0
CH10 CHANNEL 1 APOS
CH11 CHANNEL 1 ANEG
CH12 CHANNEL 1 BPOS
CH13 CHANNEL 1 BNEG
CH14 CH1INHIBIT1
CH15 CH1BCENA1
CH16 CH1BCTRIG1
VCCO3V3aux0
VCCO3V3aux1
VCCINT_1V
VCCAUX_1V8
VCCO3V3
MGTAVCC1V
MGTAVTT1V2
3V3aux
GND
Title
Holt PCIe-Mini dual MIL-STD 1553
Size B
Document Number <Doc>
Rev A
Date: Thursday, January 16, 2020
Sheet 5 of 5