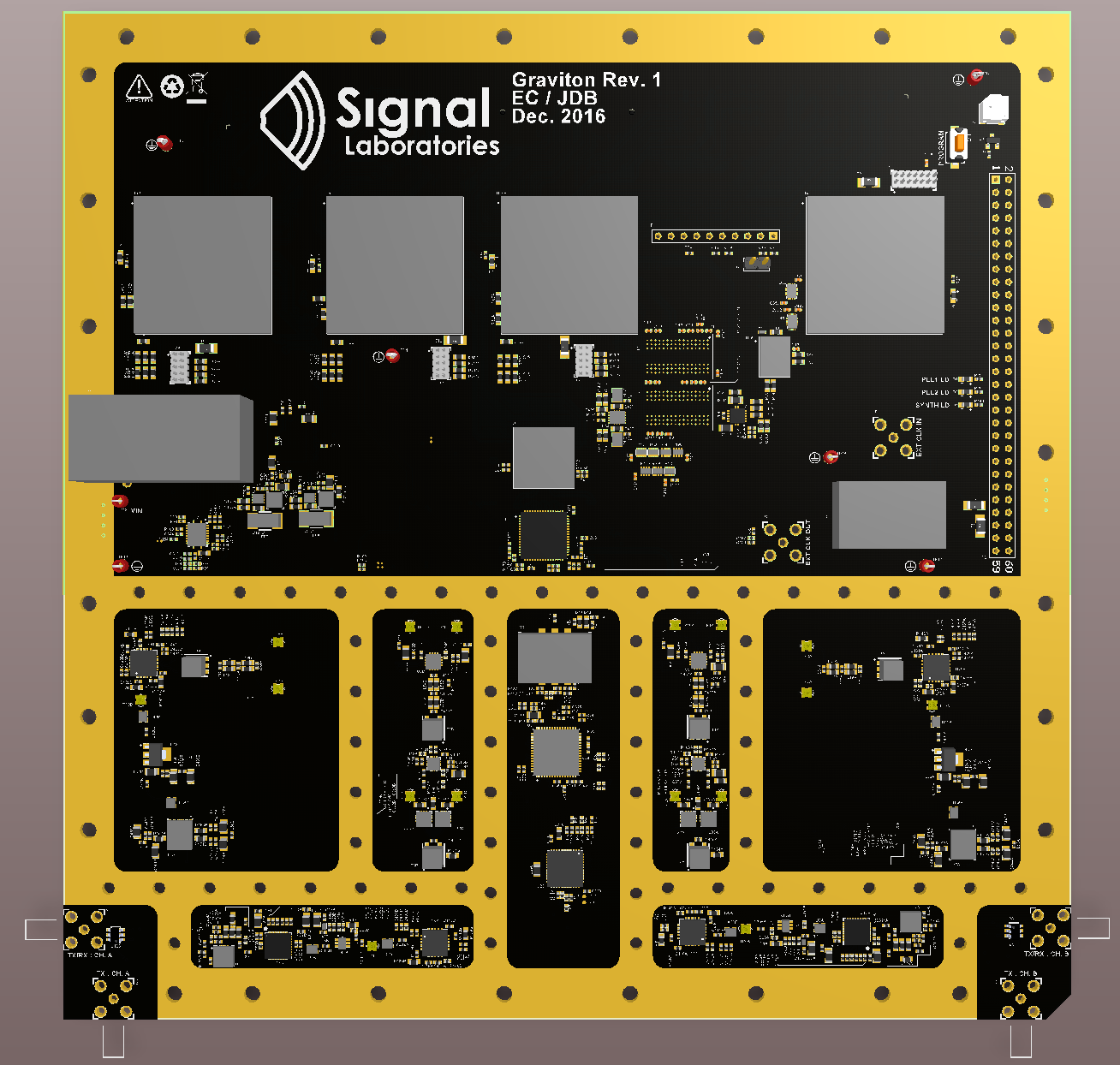
**Graviton Rev. 1**

User Guide



1. **Functional Description**

Graviton is a software-defined radio with a connector to a highly-expandable FPGA computing component. Its main parts are the Radio, Clock, Functional FPGAs, and Config FPGA. It is intended to provide all the needed radio interface peripherals for the Copper Suicide computing board. Together with Copper Suicide, it provides a complete client connection to the SigFi Network.

**Radio** – Each radio unit consists of a transmit and a receive chain connected by an RF switch to the same antenna SMA connector on the board. There is an optional capacitor population option to separate the TX and RX chains onto separate SMA connectors for doing things like simultaneous transmit and receive. There are such radio units on the board, one on the left and one on the right, each with a tx and rx chain. Both rx chains are connected to different channels of the same ADC. Likewise, both tx chains are connected to the different channels of the same DAC.

**Functional FPGAs** – There are three functional FPGAs on the board. The RX FPGA is connected to the Texas Instruments ADS42LB69 ADC and handles receiving data over a high speed DDR 16ch LVDS bus. The TX FPGA is connected to the Texas Instruments DAC3484 DAC and handles transmitting data over a high speed DDR 16ch bus. The Ethernet FPGA handles client communication and is connected to a Marvell 88E1512 Ethernet Gigabit Phy. The Ethernet FPGA is intended to deliver internet connectivity to the client, but can also be used for debugging in Graviton standalone mode, described later below. The Ethernet FPGA also has a 16-bit wide 8Gb SDRAM attached to it. Each of the FPGAs is connected by parallel busses to the Copper Suicide connector (see Interconnections, below). Copper Suicide does all the processing for the client box, communicating with the radios on Graviton through the TX and RX FPGAs, and serving up connectivity through Graviton’s Ethernet FPGA.

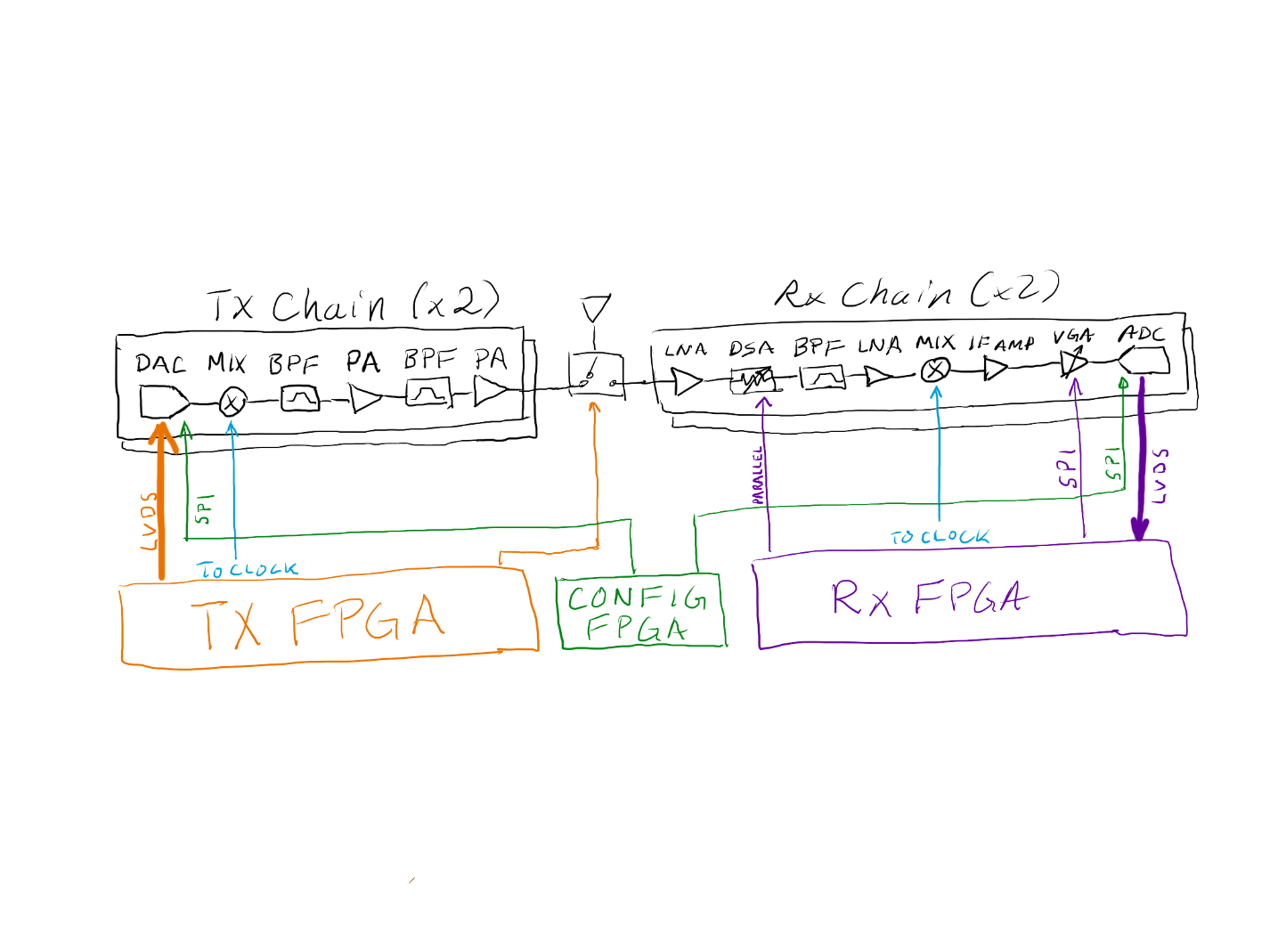
**Config FPGA –** The config FPGA retrieves and distributes the programming images for all the FPGAs on the Graviton board. It is attached to a 1Gb flash IC. It first programs itself, then reads out the images for all the Functional FPGAs, which it programs through a dedicated sysConfig bus connection to each of the FPGAs. The Config FPGA also handles the management interfaces for configuring the DAC, the board’s main PLL clock IC, and the RF synthesizer. For the bus voltage reasons, the ADC’s management bus is run directly from the RX FPGA. The Config FPGA can act as a supervisor and coordinator for the board through the GPMC interface to the Functional FPGAs.

**Interconnections –** The main data interconnects for the FPGAs are the SDR and DDR parallel busses. All of these busses are intended to go through the Copper Suicide board and go to the Copper Suicide connector. The SDR bus has 23 connections, and the DDR bus has 47 connections. The TX and RX FPGAs have one SDR and one DDR bus each. The Ethernet FPGA has two SDR and two DDR busses. On each of the SDR and DDR busses, two of the connections are routed as a differential pair and intended for use as a clocking signal. These busses are all point-to-point. There is also 23-bit wide management bus called the GPMC bus that is multi-drop and connects all the FPGAs together, including the Config FPGA. This bus also goes to the Copper Suicide connector. There is an 8-bit wide bus that runs between the TX and RX FPGAs and is intended for timing-critical coordination (if needed) between the transmit and receive machinery. There is a JTAG loop between all the FPGAs for debugging. If desired, there is a jumper on each FPGA to break the JTAG loop and isolate the JTAG to just that one FPGA.

**Graviton Standalone mode (with the Graviton Jumper board)** – The Graviton Jumper board is used to run Graviton in Standalone mode without a Copper Suicide board attached. It is a loopback board that connects each of the SDR and DDR busses from the RX and TX FPGAs to the two SDR and two DDR busses on the Ethernet FPGA. It also provides a 48V power connection.

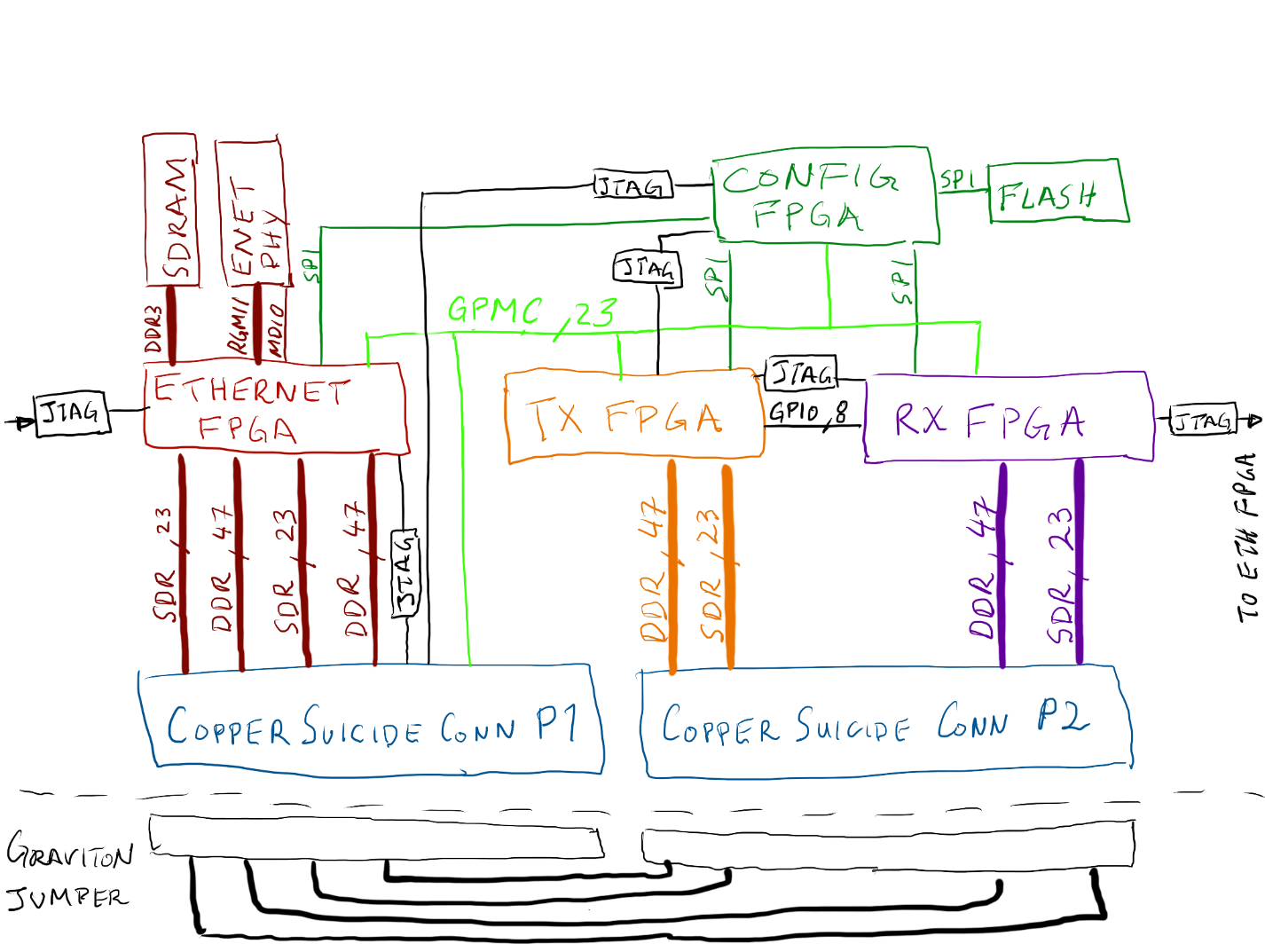
1. **RF Block Diagram**

Lorem Ipsum



1. **Digital Block Diagram**

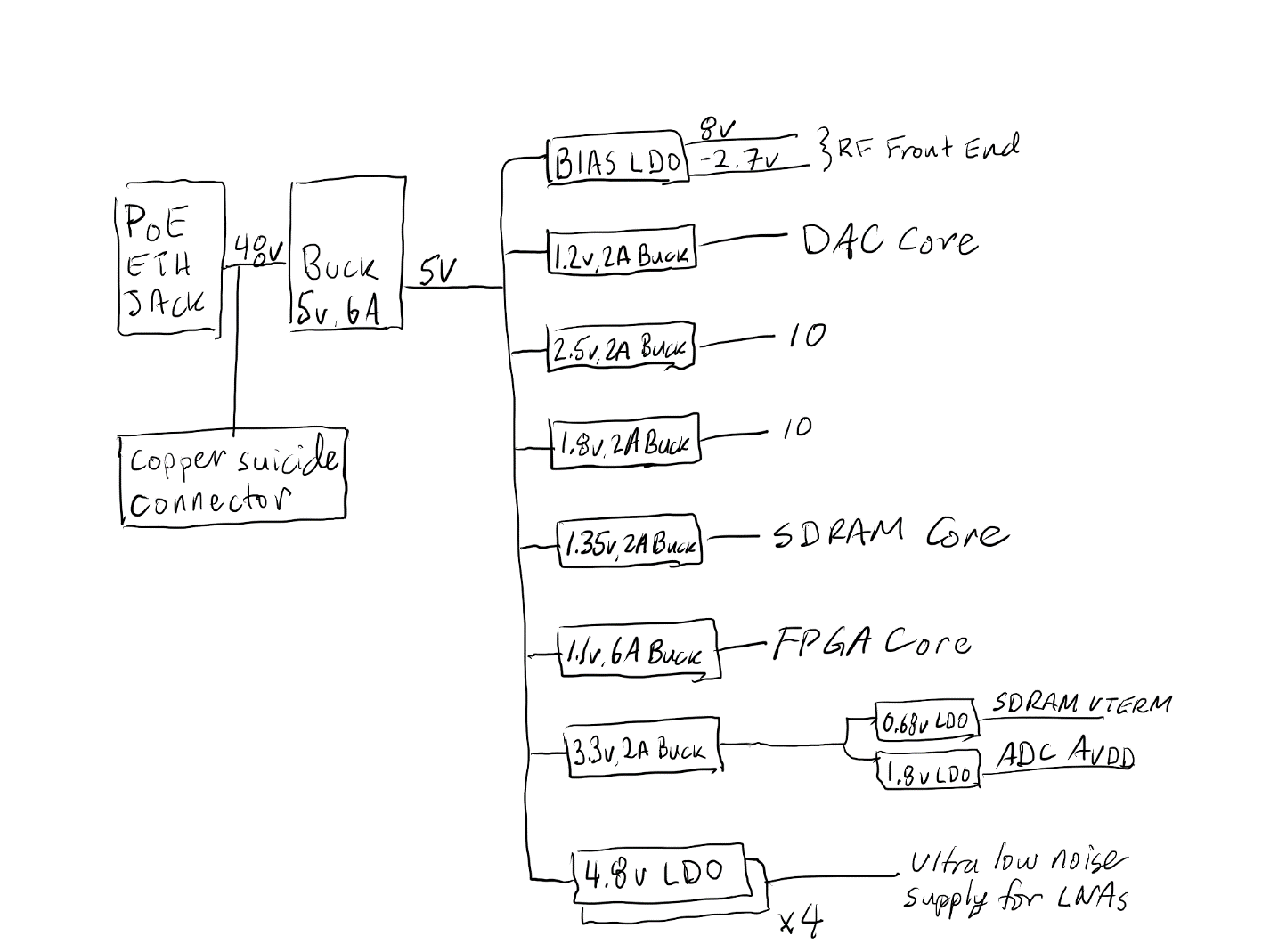
Lorem Ipsum



1. **Clock Distribution Block Diagram**



1. **Power**



1. **Debug Features**

**JTAG –** There is a JTAG loop connecting all the FPGAs (both functional and config) and looping through the Copper Suicide connector and the main JTAG connector on the board. Each FPGA also has an individual JTAG connector and a jumper whereby the loop can be broken and the individual FPGA can be addressed through its own JTAG connector.

**Buzzer –** The Config FPGA has a GPIO-activated buzzer that can be used to indicate faults or other debug information, particularly from inside a closed box.

**LEDs** – Each FPGA has a set of GPIO-activated LEDs.

**Config Debug Header –** All the major control busses (ADC, DAC, DSAs, clocks, etc) are wired out to a config debug header on the upper right side of the board.

**RF Test Points –** Key points in the RF chain are furnished with RF testpoint connectors. These connectors are normally closed. When an RF probe is attached, they open, breaking the RF path at the test point and rerouting the incoming signal into the probe.

1. **Resource Locations**

* All files checked into the git repo https://github.com/siglabs/graviton
* These are the packages sent to fab and assembly:
* https://github.com/siglabs/graviton/blob/master/hardware/20170111-GRAVITON-REV1-RC1-PKG-FAB.zip
* https://github.com/siglabs/graviton/blob/master/hardware/20170111-GRAVITON-REV1-RC1-PKG-ASSY.zip
* The fab house had a problem with the IPC file and I sent them a new one. The files currently in git include the new IPC.
* The schematic is up to date and here:
* <https://github.com/siglabs/graviton/blob/master/hardware/20170111-Graviton-REV1-RC1-DWG-SCH.pdf>
* The final stackup from Speedy Circuits is in the git here:
* <https://github.com/siglabs/graviton/blob/master/hardware/Final-Stackup-Speedy-Circuits-20170111-GRAVITON-REV1-RC1.xlsx>
* This document (User Guide) lives here: