NASA TMR RISC-V MCU for CWS

BYU IDAHO

ECEN 499 – Senior Project

Chris Porter, Kyle Toliver, Spencer Cheney, Jonah Boe

Objective

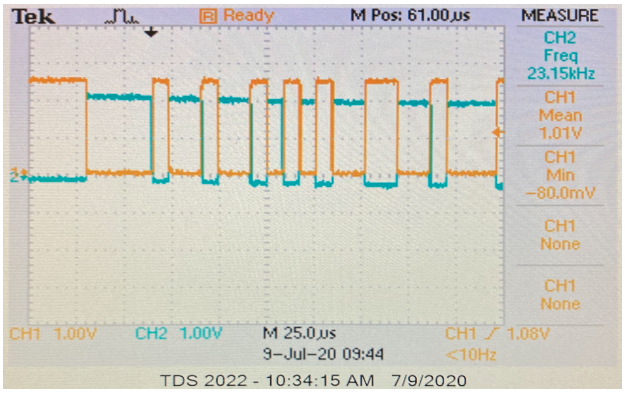
Using the PolarFire FPGA Eval kit and a custom PCB, benchmark various RISC-V core configurations in Triple Modular Redundancy (TMR) for use in NASA's Caution and Warning System (CWS) in the Portable Life Support System (PLSS) of the newest space suit (xEMU).

Hardware and Software

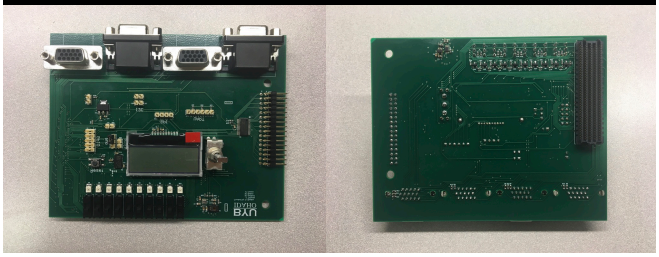
- **PolarFire FPGA**
- **Libero**
- **SoftConsole**
- **Custom PCB**

LVDS Full Duplex UART

UART has a baud rate which tell how often data is transmitted/read instead of having a common clock signal in our case this is 1,000,000 times per second. The transmit line will stay high until it is ready to transmit, once it is ready the signal will go low for a period equal to the baud rate and then transmit the data in 8 bits from the LSB to MSB. Once the data is transmitted it will send a parity bit to help with signal integrity that signals whether the data has an even amount of positive bits. Full duplex adds the specification that the device can transmit and receive at the same time through the use of FIFO queues. LVDS signal is a type of differential signal, meaning that there is a positive and negative signal where when the signal is high the positive is high and negative is and vice versa for a low signal. LVDS adds the specification that the signal will be in between 0.75 and 1.25 volts.

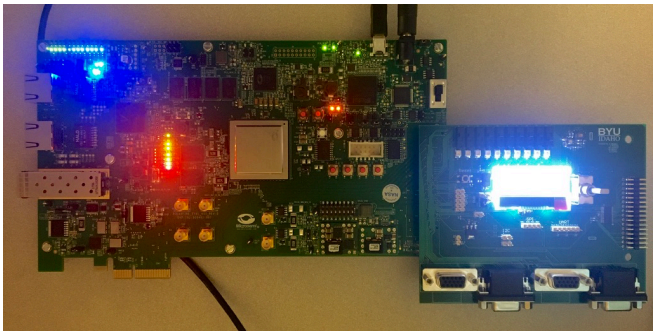


PCB



Assembly

Assembled custom PCB using both reflow and soldering iron methods. Tested PCB for shorts and errors and resolved all known issues.



Future Developments

- **Finish the other three processor designs.**
- **Fix memory controller compatibility issues between AHB processors and AXI memory.**
- **Finish the LVDS UART module in Libero**
- **Debug Softconsole code for LCD**
- **Finish implementing functions for peripheral devices in SoftConsole (Heart Rate, Accelerometer, Barometer)**
- **Test SPI and I2C receive functionality**

Conclusions

We assemble and tested the custom PCB. We start a LVDS UART module and fix other modules in Libero. In SoftConsole we developed functions to test LVDS Full Duplex UART, SPI, I2C, as well as the status of the 10 GPIO switches on the custom PCB. We started coding functions to display text on LCD screen and to control functionality through the onboard toggle switch.