March TATER Monthly Status Report
3-31-2018

Contents

1 Problem Statement

Our goal is to design a method to monitor and characterize the electromagnetic emissions of a microprocessor during boot to determine if foreign code has been injected. Limited by the types of instructions that have the most impact, the data acquisition system and analysis algorithm will be modeled accordingly. The finished product will consist of a system which can physically capture electromagnetic emissions with a custom antenna, collected using an amplifier paired with an ADC acquisition platform, and process collected data using a custom algorithm to create an EM profile of a processor's boot.

2 Hardware Summary

This month involved the adaptation of the ZC706 reference design to the KC705. Although the ADC data does not seem to be entering memory correctly yet, the following accomplishments have been made:

- Replacement of Zynq processing system with MicroBlaze and connections to other peripherals adapted.
- SPI uses Xilinx Quad SPI driver IP rather than internal Zynq SPI interface.
- JESD204B link configured to operate with the KC705 (establishes DATA ready phase).
- ADC DMA engine removed to support limited KC705 memory.
- System constraints updated and ported to KC705.
- Software driver ported from ZC706 configuration to new hardware design.
- Addition of Ethernet TCP packet transmission for gathered samples (100MB) in memory takes 20sec to complete.
- Hard and soft implementations exist as standalone projects, simply drop in the AD **hdl** and **no-OS** directory respectively and run **make**

2.1 Challenges

There have been numerous challenges with actually getting the required samples necessary, as further illustrated by this support answer we received from Analog Devices:

https://ez.analog.com/message/339471-address-width-of-axi-utiladcfifo-configuration

We need to be able to use the **axi_adcfifo** as stated in order to meet the memory bandwidths and meet timing as well without dropping samples. Removing the previous **util_adcfifo** solved one issue, however these are some ideas we are attempting in order to get it to function properly:

- Because we only need one DMA request issued, add an AXI Interconnect with **axi_adcfifo** and MicroBlaze data cache as masters and DDR as slave. – This results in lockup in the DMA end-of-transfer signaling in software. We completely removed the DMA core as there is no need for it, and connected GPIO request pins directly to the MicroBlaze. Currently debugging as it seems the most promising method.
- Use standard Xilinx FIFOs and DMA as opposed to custom Analog Device implementations. – The maximum DMA request size using these IPs is 8MB; which would cause the FIFO to fill faster than being flushed and lose samples.
- Use BRAM and DMA to transfer from the data placed by **axi_adcfifo** in DDR to the MicroBlaze. – Also debugging, however even though samples are "written" to DDR, when DDR is read back the contents have not been modified.

Another issue we have been attempting to work around is related to our other Analog Device question:

https://ez.analog.com/message/339013-error-using-axi-jesd204-jesd204upcommonv306

Because Analog Devices is constantly pushing to both their master and development branch almost daily, it is difficult to ensure a design is actually meant to be compatible. We are currently using the latest released version with 2017.4, and there don't appear to be any critical problems with that, however it is difficult to ensure this.

We have attempted to use the earlier 2015.3 version of Vivado as well, however the libraries fail to build on previous Analog Device versions because the closest that they actually had is 2015.2.1, which completely restricts the build process.

2.2 Printout of configuration of ADC and transfer requests:

```
waltz@Waltz ~/Documents/git/tater $ ./script.sh
ad9528 clock configured correctly!
ad9680 configured via spi correctly!
QPLL ENABLE
Rx link is enabled
Measured Link Clock: 250MHz
Link status: DATA
SYSREF captured: Yes
adc_setup adc core initialized (1000 MHz)
ad9680 - PN9 sequence mismatch
ad9680 - PN23A sequence mismatch
memory transfer request started.
transfering 100MB via ethernet...
complete.
```

Figure 1: UART output produced by software driver detailing various parts of configuration of the hardware components.

Note: We aren't entirely sure exactly why the AD9680 is failing the internal self tests (PN). We believe this may be related to the JESD204B link, but we have quadruple checked the pins against the schematic and it is achieving the correct sync across all lanes. Perhaps a software driver issue, or related to not reading samples from memory correctly.

2.3 Block Design

Shown below in Figure 3 is the current state of the reference design. This design can be viewed and built by downloading the hardware design on GitLab, placing it into the **projects** directory in this release: https://github.com/analogdevicesinc/hdl/releases and building with **make**. (**make open** will open the project in Vivado).

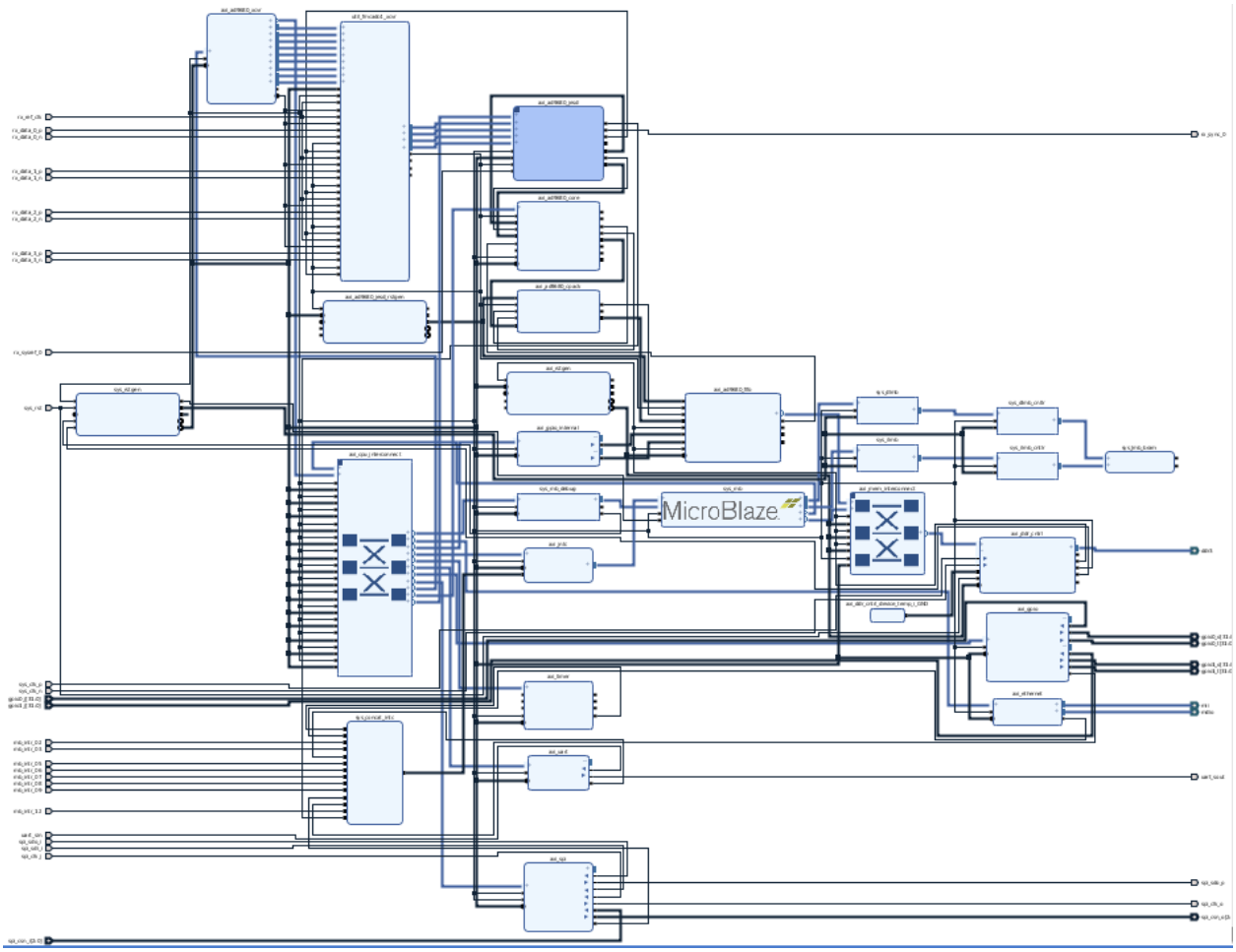

Figure 2: FPGA Block Design

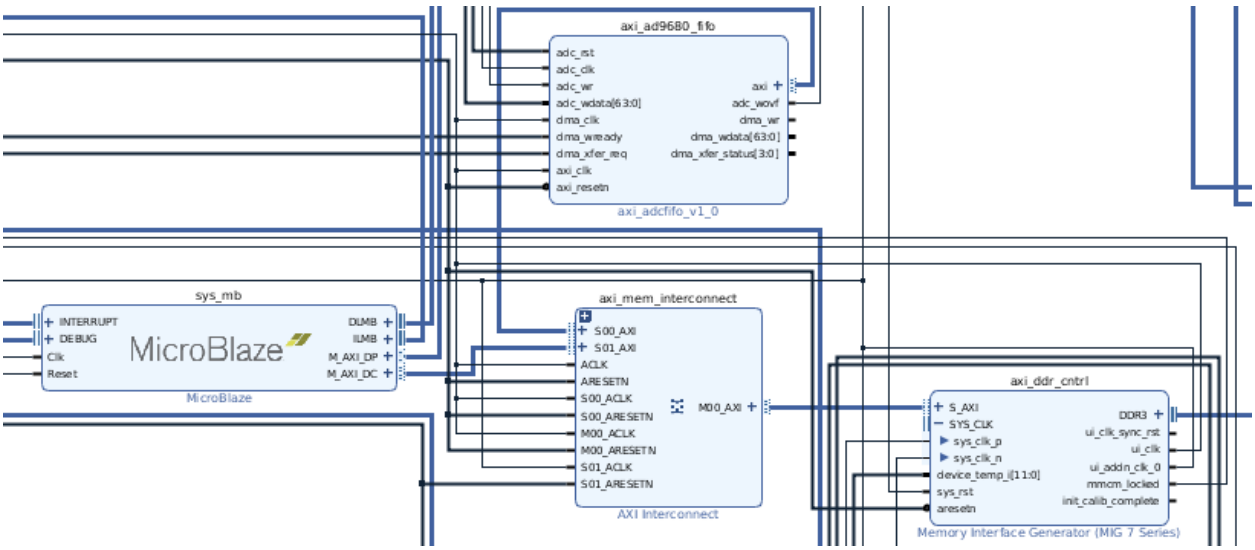

Figure 3: FPGA Memory Interface Connections (ui_clk = 200MHz, ui_addn_clk_0 = 100MHz)

3 Software Summary

This month, we worked on combining program files for usability, adding comments for future reference, documentation, along with helpful and user friendly program outputs. Upon further examination and testing against gathered hardware captures, the team has determined that the current alignment method will continue to work satisfactorily. We plan to ensure this once we establish the final sample collection method.

4 Antenna Summary

We originally were going to buy the Pulse Larsen Antenna, but when making the purchase found an even more suitable antenna for our needs. It satisfies our size constraints and is similar in structure to the probes we have been using, while still obtaining the frequency range we need, which this month we defined as a 1-250 MHz bandwidth, which can be observed below in the following captures:

Antenna link: https://www.digikey.com/product-detail/en/114990711/1597-1409-ND/6235100

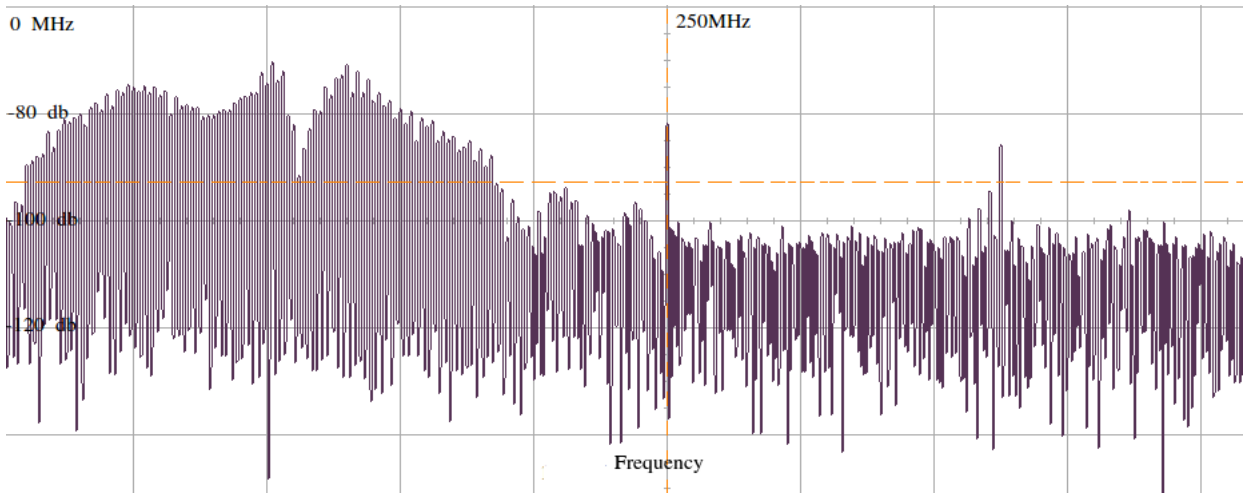

Figure 4: EM received when probes are placed in proximity of the processor.

The below Figure 4 shows the EM produced by the processor as compared to the noise level in Figure 5. Between 1-250MHz, a sharp increase in db is shown on the left. Although it may still may be useful to capture the full EM spectra, the critical signals are produced in this range.

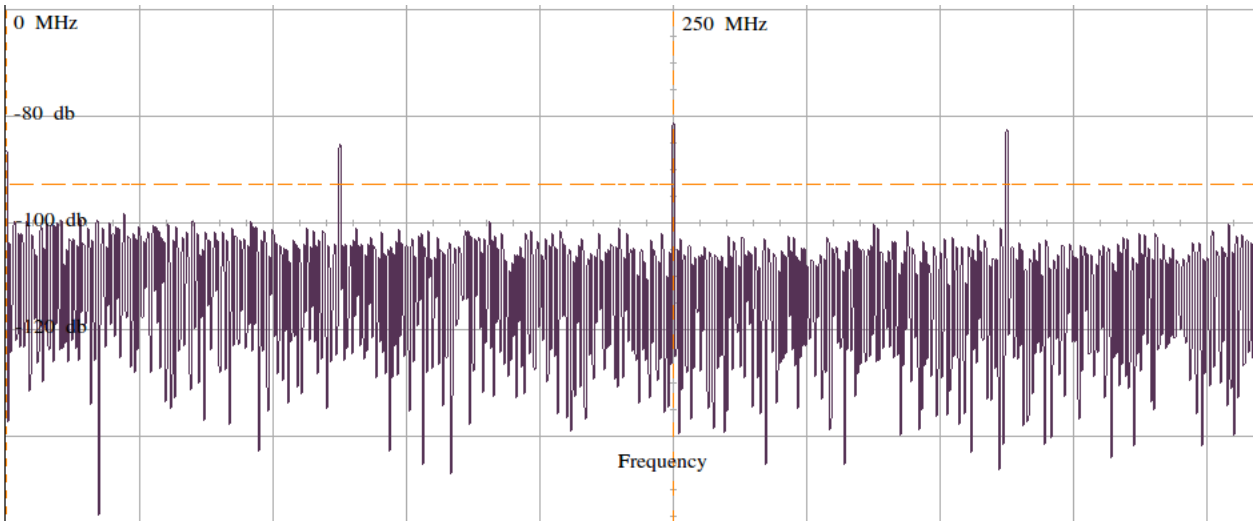

Figure 5: EM received in static environment; disconnected probes

4.1 Next Steps

We will need to implement a permanent antenna setup that will allow us to achieve consistent results. This can be done by ensuring that no components in our setup are able to move unless we need them to, such as external wires. As the antenna is similar in structure to our probes, we can improve upon the design we have been using to achieve this goal.

5 Moving Forward

We plan to write documentation including an updated Acceptance Test Plan, user guide, and final report. In addition, we are cleaning up the code and adding Doxygen-style comments. We are also planning to prepare for design expo which entails a poster board and technical presentation as well.

6 Risks

Main risks, severity, impact, and how we are mitigating the risks:

6.1 Risk 0

FPGA design is providing multiple interesting challenges.

`Severity`: High

`Mitigation`: We have acquired a backup oscilloscope in order to properly demo for expo in terms of time constraints. We are continuing to diagnose issues with data rates and memory issues as described above.

6.2 Risk 1

Detection of potential modifications. Once we have the hardware setup finalized, this will be quickly testable against our test suite.

`Severity`: High

`Status`: This information will also be provided during the acceptance test plan and the user guide as a large table and all performed variations.