Ryan Hansen
Last Edited 5/6/20

SDR – Introduction

Foreword

Hello, if you are reading this you might be interested in continuing the development of CySat or you might be curious about this subsystem and how it fits into the entire cube satellite system as a whole. In this document I will cover things that I learned about the SDR system and do my best to explain it in a way that anyone being handed this project can understand. If you have taken or plan on taking CPRE-488, it will greatly help your understanding of the hardware that comprises the SDR.

Purpose

SDR stands for Software Defined Radio. I will not get to in depth into what this means but essentially you taken analog inputs and pipe it into software and then perform processing on those signals in order to get a desired result. CySat's main mission is to collect moisture data on the surface of the earth from space. I won't dive to into detail here because it's fairly irrelevant to the software side of things, but a passive antenna or Radiometer is used to collect the vibrations of molecules on the earth. By tuning into specific frequency for H20 you can detect an amplitude of the signal and infer an amount from a controlled data set. For CySat, the analog signals piped into the SDR are coming from an array of LNAs or Low Noise Amplifiers. The LNAs boost the analog signal coming from the Radiometer. The goal of the SDR is to sort out the radio signals for a specific spectrum and then perform some processing on that and save it as digital numbers in a file. All of the previously mentioned operations have already been done by a program called GNU Radio which runs on the SDR. Most of the remaining work left for the SDR surrounds the control flow needed to send the collected data files back to earth.

Hardware

Alright let's talk a little bit more about what the SDR actually is. The manuals for this hardware are in the document's subfolder for the SDR Board if you would rather get information from there. The SDR used on CySat is an embedded FPGA with ARM processor system that has an Analog Component converter integrated onto the board. The core of the system is the Xilinx Zynq FPGA. This FPGA has a small ARM processor inside of it that allows software to run on it and have direct signal maps to hardware and configurable digital logic that can be set up to talk to hardware ports. The FPGA on the SDR is set up to run an embedded version of Linux. This allows files to be developed and ran on the FPGA as if they were running on a PC. Most of the low-level programming and hardware config has been completed. Again, if you have taken CPRE-488 you will have some experience with this type of hardware. If you are unsure of what an FPGA is, I suggest doing some research to get an idea.

Continued Development

As previously mentioned, the work SDR is not complete. There is a pretty good solid foundation that has been set by me and the previous teams that have worked on this project before. Continued development should not be too difficult. Check out the other documents listed as "SDR – Overview" and "SDR – Next Steps" for much more detailed information on the next steps for working on this hardware.