Scholarship in Practice

Decision 1: Choosing a 50 MS/s sample rate software defined radio

The first decision the group made was choosing the sample rate at which our radio would operate at. The software defined radio needed to produce a minimum sample rate, which was calculated by the range that we wanted the SDR-SAR to be able to distinguish objects at. The range that was given to us by our sponsor was 40 feet, and so a sample rate of 50MS/s was chosen to be able to distinguish between targets as well as give us some room for error. The choice proved to be sufficient, as we were actually able to increase resolution from 40 feet to around 22 feet.

Decision 2: Purchasing the B200 Software Defined Radio

The decision to purchase the B200 SDR was based on two main factors: cost and performance. As decided previously, we needed a radio that could sustain 50MS/s sample rates. Due to the limited budget and general high price of SDRs that use an ethernet port to communicate with the host PC, we were limited to a USB based radio. The B200 proved to be most cost effective and also advertised a 50MS/s rate which fit our criteria. The choice was suboptimal because the USB communication protocol stopped us from being able to stream large files to the radio. Given a larger budget, an ethernet-based SDR would have been much better and allow us more flexibility.

Decision 3: Using snapshots for SAR imaging instead of streaming

For the original plans of purchasing the USRP B200 radio, the team wanted to be able to stream the image of the metal object using the SAR. Upon purchasing the USRP B200 radio, the team found that there was a limitation with the radio purchased for transmitting and receiving data at the same time. Relying on USB 3.0 that the USRP B200 radio had meant that data speed was restricted coming from the computer to the radio. As a result, the team decided to take snapshots instead as it would be within the tolerance the radio purchased could handle. Streaming could only be feasible if another, more powerful radio, would be used. As the budget does not allow for more expensive hardware, snapshots of the scene instead of a continuous stream of data is the best choice available.

Decision 4: Using MATLAB to process the data obtained from the B200

While originally the group meant to utilize GNURadio Companion to process all of the data collected from the radio as it would be a more unified system, this was decision was replaced by using MATLAB to process and filter anything the radio provided. This was because it was easier to code in MATLAB rather than make the custom C++/Python modules that GNURadio would have needed. Furthermore, it was better as less processing needed to be done by the SDR and GNURadio, which was left to only acquire data. This decision would be

repeated if doing it all over again, especially since the snapshot method utilized means that you do not need to make any data processing in real time, but instead it can be done in chunks.

Decision 5: Using an Amp with VCO or just the VCO for attaining a 7.8GHz center frequency

The team made the correct decision here, as it wasn't needed to incorporate an amp with the VCO. The VCO alone could attain the 7.8GHz center frequency. Through programming the VCO using TICS Pro, the center frequency could be set accordingly for the upconvert to 10 GHz with the SDR. From these choices, the team learned the importance of planning the design before buying components assumed needed. If the scenario were played out again, the team would avoid combining the amp with the VCO for achieving the desired center frequency.

Decision 6: Using Linux

While the entire project could have been done using software that runs on the Windows operating system, our team chose to use Ubuntu for the data acquisition part of the project. This decision was made because the GNURadio Companion software that communicates with the radio is much more stable on Linux when compared to Windows. Even if becoming familiar with Linux took a little bit of work, the stability and efficiency of the platform made for a much better choice than Windows.

Decision 7: Dual Boot or Virtual Machine for installing Linux

For this decision, the team decided to install Linux using the dual boot method. This decision turned out to be the correct one, as it allowed for running all the software required for programming the SDR with no issues. While both installation methods for Linux require a set allotted space, the risks of using a virtual machine outweigh the challenges of setting up a dual boot. The virtual machine method could have random crashes while using it. Furthermore, trying to locate source files for GNURadio Companion would have been more difficult to find. The risks for setting up the dual boot consisted of potentially bricking the PC, as the BIOS needed to be updated for the laptops in order to install Linux. However, locating the source files needed was much easier and there were no crashes throughout any tests done.