Greetings all,

I spent Monday through Friday of this week at the Information Theory and Applications conference in La Jolla. It was like nerd paradise day camp, and I have a ton of great stories about entropy, comms theory, coding, and the application of info theory to elections, genomics, phylogeny, game theory, machine learning, recipe composition, neural nets, 5G and many other fields. I’ll save the funny stories for when we meet in person!

So about Phase 4. This week we had our second phone conference with AHA. Representatives from AMSAT and Virginia Tech were on the call. Discussions continued about using cores from AHA on the Phase 4B spacecraft. These cores would allow us to fully utilize the DVB-S2X standard on the transmitted downlink. Implementing this standard on the space segment dramatically reduces risk, reduces schedule, and increases satellite payload capability.

Adaptive modulation and a pilot channel scheme are both already fully specified in DVB-S2X. These are functions that we identified as necessary for successful system operation. In addition to Phase 4B, the Cube Quest Challenge project is moving forward with using coding and modulation specified by the very low SNR configurations of DVB-S2X. Since our radio supports Cube Quest Challenge, coordinating the air interface as closely as possible between the two projects is a huge win. Phase 3E can also use this same standard for a triple win.

Our job, as Phase 4 Ground, is to design an open source amateur radio implementation of the DVB-S2X standard.

Efforts to achieve a firm commitment to this standard from the space segment team successfully continued during the week. We were able to move from enthusiasm to agreement in talks with Dr. Jonathan Black and Dr. Robert Magwier. The working copy of the air interface has been updated with OQPSK for the uplink and DVB-S2X for the downlink.

What does it mean to say we are using DVB-S2X?  
  
DVB-S2X specifies physical layer and data link layer protocols. When we say physical layer, we mean literally what’s traveling along physically through the air. The standard specifies what type of waveforms are allowed. There are a variety! DVB-S2 specified QPSK and 8PSK (for broadcast applications) You can use these modulations in non-linear transponders driven near saturation. The standard also allows 16APSK and 32APSK. The X in S2X adds additional coding and modulation for the top end, for powerful clean channels, and at the bottom end, for very low SNR conditions. Very low SNR modes are what Cube Quest Challenge will use and possibly what SatChat1k modes for Phase 4 will use.

Source data is organized in useful ways by the data link protocols in the standard. We’re using something called Generic Stream Encapsulation. After this stage, the data is given forward error correction. The standard specifies the type of forward error correction coding, and the allowable rates. The particular coding used is a concatenated LDPC-BCH code. This stands for a low density parity check or LDPC code as the “inner” code and BCH code as the “outer” codes. LDPC codes are state-of-the art. Getting familiar with them provides a tremendous educational opportunity for all of us.

Code rate is how many bits go in over how many bits come out. Expressed as a fraction. A small amount of coding is sufficient when you have a strong clear channel. A large amount of coding helps to make your data durable if it has to go through a messy noisy channel. There is a list of code rates (at least eleven) in DVB-S2 to choose from ranging from ¼ to 9/10. The important idea here is that when your channel changes, the code rate is allowed to change. We aren’t stuck with worst case and end up leaving throughput on the table on a hugely expensive resource.

With DVB-S2X we have the opportunity to design an advanced radio system that can choose modulation and coding dynamically based on how good the channel happens to be. This means we can handle many communications challenges that we otherwise would not be able to address including a greatly expanded range of radio capability, how well the dishes are pointed, noise, interference, and anything else that might get in the way.

This is not a scary standard. It’s essentially “hey, organize your digital data like this, and then use this forward error correcting code. Support the use of these particular code rates. Use these modulation schemes. The framing and overhead will tell you what has been used on what frame. And, if you want to adaptively modulate and code, then here’s how you might want to tell the system what you want.”

Developing the algorithm to select what coding and modulations is being used depends on things we can measure in the system. Usually this is the ratio of carrier power to noise power or a function of that ratio. This is stuff we can do. It’s not rocket science, but it’s close to a rocket, so you can tell your friends you are close to being a rocket scientist.

If we run into some sort of brick wall, then our backup plan is to use some sort of constant coding and modulation scheme. That would be very well-traveled and very, very boring ground, but if you’re scared of standards and coding schemes with so many letters in them, rest assured, there is solid ground underneath us.

People from AHA will be here in San Diego in a week or two for a conference, and we’re going to meet up and have dinner to talk about things. I and others will be traveling to Austin Texas the week after that.

We need people that either have FPGA experience, or are willing to learn. Either one. You do not have to be an expert; you just have to be willing to accidentally become one along the way.

Our github is <https://github.com/phase4ground> and join our mailing list by volunteering for AMSAT technical service at <http://www.amsat.org/?page_id=1096>