OVJ Proteus Software Milestones

The first major milestone for the OpenVnmrJ (OVJ) Proteus project is to be able to run ahX and ahXX experiments from OVJ and process the data. It is expected that this will take from 3 to 6 months. To achieve this goal, there will be several intermediate milestones, which are outlined below. As described in the software design document, there will be changes required in the OVJ PSG software, the “procs”, and the development of a new translation program (TEproc). The biggest unknowns for this project concern programming the Proteus system. Therefore, I will focus on resolving those unknowns first. The proposed design facilitates this by allowing the “universal” experiment codes (UCODES) file to be created by hand and then submitted to the Proteus system by the translation program (TEproc). Individual elements of a pulse sequence will be programmed and eventually chained together to achieve the final goal. For example, the first step will be to execute a variable length delay on the Proteus system. A UCODE file containing

```
  delay: 1
```

will be written and the TEproc program will read that file and send to Proteus the appropriate SCPI commands to execute a delay of one second. (It is my current understanding that the Proteus system is ultimately controlled via SCPI commands). It should be possible to execute many of the pulse elements by hand editing the UCODE file and modifying TEproc to send the Proteus SCPI commands. I envision the following series of steps.

Execute delay
Execute delay - pulse
Implement looping over delay - pulse
Execute delay - pulse - acquire
Implement time averaging via looping
Execute delay - pulse with markers to trigger amplifier blanking
Implement phase cycling, including real-time math functionality
Implement an arrayed experiment.

For each intermediate step, there will be questions that need to be resolved, such as:
How is the power of a pulse set?
How is the spectrometer frequency set?

At this point, there should be enough understanding of the Proteus system to implement the s2pul sequence from OVJ. This will involve modifying PSG to generate the UCODE file and modifying the procs to execute the TEproc translation program, and to receive the data from Proteus so that it can be processed by OVJ. Some system testing should then be feasible.

Continuing with the simple UCODE file and TEproc

Execute a shaped pulse
Execute a 2-channel sequence with pulses on each channel
Execute decoupling via the status command.
Execute programming decoupling, including cross-polarization.

At this point, it should be possible to implement the ahX and ahXX pulse sequences. PSG will need to be modified to output the correct UCODES. The first major milestone will be achieved.

As mentioned at the start, the main unknowns concern programming the Proteus system. The time required to do this is also unknown. We will get a better understanding as we progress. The initial development to send a delay command involves initiating network communications, being able to send and receive messages, translating a delay into the appropriate SCPI commands and sending these to Proteus.

The biggest unknowns at this first stage include using markers, looping, phase cycling, data acquisition, and experiment arraying. We will need assistance from Tabor to resolve these issues.

This first major milestone focuses on the RF transmit and receive functions of the Proteus system. Following this initial milestone, it might be interesting to see how the new capabilities of the Proteus system can be taken advantage of. There are additional requirements that are not directly related to RF issues. These include things such as steady-states, pre-acquisition delays, block-size processing, shimming, temperature control, spinning, status reporting, etc. These will be investigated at a future time.