Mapping The Galactic Plane

Kenji Doi, David Halperin, Mason Hoey, Patrick Jackson, John McDougal

11.28.18

**Introduction/Summary:**

The goal of this project was to construct a system that would allow us to capture the levels of hydrogen at 21-cm in our own galaxy. With this instrument, we would have to filter out frequencies other than those around 1420.4MHz to gather accurate data. An aluminum horn was constructed out of foam, foil and a container which allowed us to direct our system at an area in the sky. A combination of filters and amplifiers focused the antennas input to output the desired frequency levels of about 1420MHz photons emitted by hydrogen in the milky way.

After constructing the instrument, we attempted to use a raspberry pi paired with a spectrum analyzer to record the hydrogen levels continuously over time and store the data in a csv file. The csv file would be interpreted by python code and proceed to map the levels of hydrogen throughout the day as the galaxy passes overhead.

**Materials:**

The types of materials we used were broken up into the two categories below. We chose the materials for this project based on the low cost of the items as well as adaptability we would have given the items we purchased.

* Horn
  + Foam Board
  + Aluminum foil
  + Duct tape
  + Aluminum rectangular container (1 gallon)
* Electronic System
  + Antenna
  + (2) LNA
  + 1420MHz BPF

**Methods:**

* Rectangular pyramidal horn
* RF front end (antenna)
  + Testing reliability of electronics with VNA
* Recording software
  + Raspberry Pi 3 model B
    - Ubuntu Mate
    - GNURadio
* Spectrum analyzer
  + RTL-SDR
  + VNA
* Plotting software
  + Waterfall plot depicting freq channels vs time
  + 2 scatter plots depicting H1 vs time & H1 vs galactic coordinates

**Discussion/Results:**

We initially ran the Raspberry Pi with Raspbian but after coming across some trouble getting the RTL-SDR to work with GNURadio we switched to the suggested operating system, Ubuntu Mate, and installed GNURadio. We came across an issue installing the soapysdr package for GNURadio with the error stating it couldn’t be found. After completing the flowchart to create a waterfall plot, the RTL-SDR yielded no results. We proceeded to test the spectrum analyzer on another computer and found that the instrument wasn’t the problem, so to continue we decided we had to record the information manually and move away from the Raspberry Pi.

* Raspberry Pi initially running Raspbian operating system with GNURadio
  + Unable to find RTL-SDR
* Ubuntu Mate used as alternative operating system with GNURadio
  + Soapysdr package for GNURadio was unable to be found.
* RTL-SDR returned blank reading through Raspberry Pi
  + Testing confirmed error was with Raspberry Pi
* VNA used as alternative
  + Manually gather data

Attached to the horn we created the RF front end composed of an antenna combined with two LNA’s and a BPF. Joining the VNA with these components allowed us to test our system in which the RF front end proved to pass 1420MHz with a 63.8GHz band. Mistakenly, the instrument was tuned incorrectly and a half wave probe was tuned as opposed to the quarter wave probe that was required. Once this was corrected we were able to see multiple resonances near 1420MHz. After testing and debugging, the system appeared to be operating as expected.

* Using the VNA the RF front end passed 1420MHz with a 63.8GHz band (Figure 4.3)
* Accidentally, a half wave probe was tuned to absorb around 1420MHz.
  + Using rectangular can as a waveguide without the horn (Figure 4.2)
* Objective was to tune a quarter wave probe to absorb 1420MHz at the horn’s focus
* A more accurate antenna was tuned which received multiple resonances near 1420MHz (Figure 4.1)
  + Using rectangular can as a waveguide with the horn
* Upon testing the entire receiver (horn and antenna), no resonance was achieved within our 100kHz band centered at 1421MHz. (Figure 4.6)
* Afterwards, the system was debugged. The electronics were operating nominally and the antenna appeared to have a reasonable resonance outside of the horn. (Figures 4.7 and 4.8)

Constructing the horn

The original plan was to built a horn with an aluminum sheet, but instead we decided to change main material to a styrofoam sheet since ease of material processing. To make the horn electromagnetic reflective, we used a tin foils taped by a duct tape. The dimensions were cited from another project which was done by a Harvard student. A focus of this horn is a line instead of a point due to rectangular base. An issue which we faced was the strength of the horn, due to material weight, it was fragile, with risk of flowing away by wind. To solve this issue we added some scrap perpendicular metal stick in edges of the horn to add weight/enforce the structure. Another solution we add was a styrofoam sheet. We added on rectangle sheet slightly smaller than base and pushed that inside the horn to enforce the structure more.

With python, code was prepared to create a spectrum of the 21-cm hydrogen line within the galactic plane. The csv file containing the data from the Raspberry Pi would be used allowing the code to decipher the data and create a couple graphs.

* Graph 1: Waterfall Plot
  + Comparing different frequency channels over an extended period of time.
  + Center point: 1420MHz
  + Bandwidth: 100MHz
  + Hour domain: 12hrs
* Graph 2: Scatter Plot I
  + H1 vs time; time is set to Julian Days
* Graph 3: Scatter Plot II
  + H1 vs galactic coordinates

**Expected Results/Conclusion:**

In the end we ran out of time to fully complete the experiment and our instrument yielded no real results during testing. We were however able to adjust our horn and the RF front end and collect data in the lab. We concluded the antenna was tuned with only the can and not the can with the horn. This was then adjusted and resulted in the electronic components working normally with the antenna appearing to have a reasonable resonance outside of the horn.

The waterfall plot would have portrayed different frequency channels over an extended period of time. We would expect to see a greater concentration near the center of the waterfall (which is 1420MHz). The first scatter plot depicts H1 vs time; time is set to Julian Days. This plot would have shown us how hydrogen moves in the galactic plane over an extended period of time. The second scatter plot is comparing H1 vs galactic coordinates. From this, we should have seen different hydrogen concentrations in different parts of the galactic plane at one time.

**Charts and Figures:**

|  |
| --- |
|  |
| *Figure 4.1, Tuning the quarter wave probe just before testing it on the roof. There is resonance around 1420MHz.* |

|  |
| --- |
|  |
| *Figure 4.2, RF front end displaying resonance at 1420MHz* |

|  |
| --- |
|  |
| *Figure 4.3, Proof that the 1420MHz BPF (the BPF used in the experiment) was better for “seeing” the 21-cm line* |

|  |
| --- |
|  |
| *Figure 4.4, A comparison between the 1420MHz (light blue) and 1500MHz (dark blue) BPFs .* |

|  |
| --- |
|  |
| *Figure 4.5, Gathering an absorption reference of the horn from the ground. This reference data would later be compared to the spectrum generated by absorbing radiation from the sky. The reference is necessary in determining if the system was functioning properly.* |

|  |
| --- |
|  |
| *Figure 4.6, Unfortunately, actually running the experiment yielded no results. Only a garbled mess of frequencies was observed, and there were no resonances. Most importantly, there were no resonances at 1420MHz. The experiment was a failure, as denoted by a disappointed Patrick.* |

|  |
| --- |
|  |
| *Figure 4.7, Results of Debugging the probe outside of the rest of the system.* |

|  |
| --- |
|  |
| *Figure 4.8, Results of debugging the RF Front End.* |