The proposed GNU Radio design interface is shown in the figure 2.

One of the acquisition systems developed at Jicamarca is called JARS (Jicamarca Acquisition Radar System), an eight-channel digital reception system that can collect data at a rate of 1 MHz per channel. This system is in charge of acquiring the modulated signal, digitize it and send it to the acquisition computer through a PCI (Peripheral Component Interconnect) interface.

Currently, a new version of JARS is being developed: JARS 2.0. This acquisition system uses an ethernet interface to communicate with the computer and it uses a similar software than the previous version to the acquisition control. This software is named JROAcquisition, and is coded using C and C++ language. The different libraries, dependencies and data types used in JROAcquisition make the compilation process a tedious task.

We chose GNU Radio to design and implement an interface to manage JARS 2.0 because we will be able to simulate the different configuration of the radar and see which is the best to detect any kind of specific phenomenon before operating the radar in high power.

This interface will be coded inside the first block named: Jarsmode. This first block will give us the data acquired for the JARS 2.0 as a data type std: complex, which it means a sampled quadrature signal. The second block is the Throttle block. This is a core block which establish the rate at which the samples will pass through the blocks in the flow graph.

The third block is named CoherIntegrRoj. This block will do a coherent integration of the obtained samples. This block will be a new and personalized out-out-three block, which means it is not part of the core blocks which come with GNU Radio. The last block is the QT GUI Time sink. This is a core block which show us the sampled quadrature signal, after be processed, in time.

We also have the option to save the acquired data in hdf5 format because of GNU Radio has blocks which can do this for us, as Digital RF sink.

This will be designed as an Out-of-tree block of GNU Radio, which means it will be a new and personalized block out of the general blocks given for the GNU Radio core. This interface will configure the JARS 2.0 with the help of a metadata file with extension .rscp, which contains all the parameters needed for the configuration of the radar, as: IPP (inter pulse period), NSA (numbers of samples), DH (height resolution), etc.

This interface will follow the structure of an object oriented programming (OPP). The different stages that conform this interface will be shown in the next flow diagram.