ABSTRACT

MDSplus is a software tool dedicated to data acquisition, storage and analysis used for complex scientific experiments. We will show how to set up a very simple experiment, manage data retrieval, storage and consumption using MDSplus and Python. JupyterLab is used as the interactive development environment.

BACKGROUND

As a data acquisition software, designed as a new paradigm for data analysis, MDplus has been extensible used as one of the main software tools to acquire and organize the vast amount of data coming from magnetic fusion energy programs. The intent of this poster is to show that it can easily be used for any kind of data-acquisition systems: from the most complex to the simplest.

METHODS / IMPLEMENTATION

METHODOLOGY

A custom device is used as the data provider. This device is an open-source electronics platform hardware that consists of a microcomputer (Fig. 1) and sensors (Fig. 2).

Fig. 1. Arduino UNO microcontroller.  
Fig. 2. Garmin Lidar-Lite r3HP (left) and Temperature/humidity sensors.

Fig. 3 shows the JupyterLab IDE. It allows for code development, notebooks and data visualization.

IMPLEMENTATION

MDSplus Device represents the hardware in the data acquisition system. Information associated with a given device will be stored in a set of nodes of the pulse file. Figure 5 shows what the tree structure looks like for our particular device.

The code that represent the device (see Figure 3), written in Python, has the following structure:

- **The Device**: an Python subclass that defines the Arduino device.
- **The Tree** structure and nodes: defined in MDSplus parts[].
- **The Port** communication with the device and sensors: PySerial library.
- **The INIT** and **STOP** methods: to start and stop the data acquisition.
- **The TREND** method: for acquisition using launchd/systemd.
- **The STREAM** method: for acquisition using the data streaming.

Additionally, an Arduino **sketch** needs to be uploaded into the microcontroller.

HARDWARE

The device (Fig. 6) itself is composed by the following:

- An Arduino UNO: a microcontroller board (Figure 1)
- A Garmin LIDAR-Lite v3HP sensor. (Figure 2)
- A DHT11 temperature and humidity sensor.

Fig. 6. The Device.

REFERENCES and ACKNOWLEDGEMENTS


This work was funded under DOE cooperative agreement DE-SC0012470.