

I am creating this document so that in the future if there is trouble figuring out RoboRIO communication and data collection, my attempts are outlined here and you can learn from my failures/successes. Currently, I am working on the SSH option and believe that I am close to full implementation. I have explored each of these options and made attempts with each one but have decided that SSH was probably the appropriate one to pursue for this project and therefore that is the only option with commits. (Other attempts were made in local files that I deleted once I decided they weren't worth pursuing).

## Options for RoboRIO Communication and Data Extraction:

### Direct Communication via USB

- **Description:** The RoboRIO is connected directly to the computer via USB, and communication is attempted through its local IP (172.22.11.2)
- **Attempts:**
  - **SSH (Secure Shell):**
    - **Success:** SSH communication was successfully established by connecting to the RoboRIO's default USB IP (172.22.11.2). Credentials used were typically admin for both username and password
    - **Tools:** The SSH.NET library was used in C# for SSH communication and SCP (Secure Copy Protocol) for file transfer
    - **Outcome:** Successfully connected, retrieved the list of log files from the RoboRIO, and downloaded log files using SCP
  - **Issues:**
    - Initial connection failures due to incorrect credentials or incorrect IP configurations. Ensuring the correct USB IP and RoboRIO settings was critical
    - Vulnerability warning in the SSH.NET library, which required updating the library version to 2020.0.2 to fix security issues

### SSH Over Ethernet

- **Description:** Using the Ethernet connection between the computer and RoboRIO, similar to the USB connection, communication is established over the network.
- **Attempts:**
  - SSH was attempted by connecting to the RoboRIO's Ethernet IP, which can be found in the network settings of the RoboRIO
  - **Success:** Communication and file transfer were successful using SSH and SCP over the Ethernet connection. The process was the same as the USB connection, with similar credentials
- **Challenges:**
  - Network configuration needed to be handled carefully, ensuring the correct IP address and network interfaces were used

## FTP/SFTP for Data Transfer

- **Description:** FTP and SFTP (Secure File Transfer Protocol) were explored as potential options for transferring log files from the RoboRIO
- **Attempts:**
  - SFTP was tested but required enabling additional services on the RoboRIO that were not natively available
  - **Challenges:**
    - Setting up an SFTP server on the RoboRIO would require additional custom configurations
    - Decided not to choose this option due to complexity

## CAN Bus Communication

- **Description:** The CAN Bus protocol is another possible way to communicate with the RoboRIO for advanced data collection
- **Attempts:** This option was explored conceptually but not implemented in practice during this phase. It is mechanically more complicated than I was able to deal with for this sprint
- **Challenges:** Requires additional hardware and deeper integration with the RoboRIO's CAN system

## Manual USB File Extraction

- **Description:** Another basic option is manually extracting files from the RoboRIO by plugging in a USB drive and copying files from the device's file system
- **Advantages:**
  - Simple and requires no special software or configuration
  - Useful in situations where remote access is not possible
- **Limitations:**
  - Not automated and requires physical access to the RoboRIO
  - Limited by file system permissions on the RoboRIO
- **Success:** Not implemented in this phase but remains a fallback option