Real-time monitoring and diagnostic of a vehicle

This program will monitor the status of various vehicle components, in real-time, using a number of sensors on board and then provide the driver the information from the sensors using a display. This is done by implementing a set of real-time periodic tasks that will frequently obtain data from in-car sensors and display information to the driver. The sensor data will normally be obtained by a OBD-II port on a vehicle but due to the current situation a dataset is provided in the zip file that has the periodic data from the sensors.

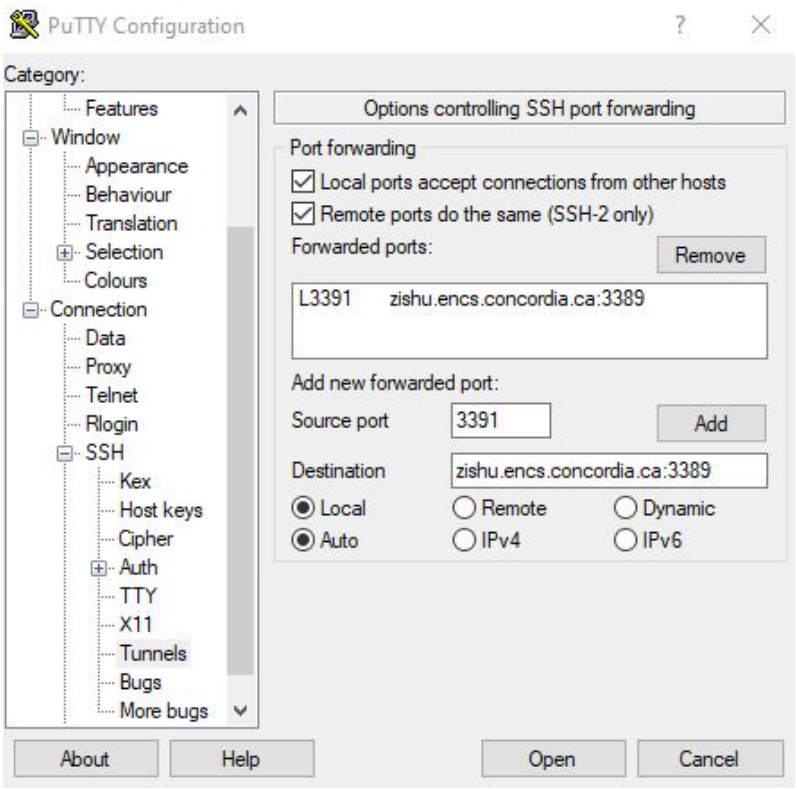
Prerequisites:

Before you continue, ensure you have met the following requirements:

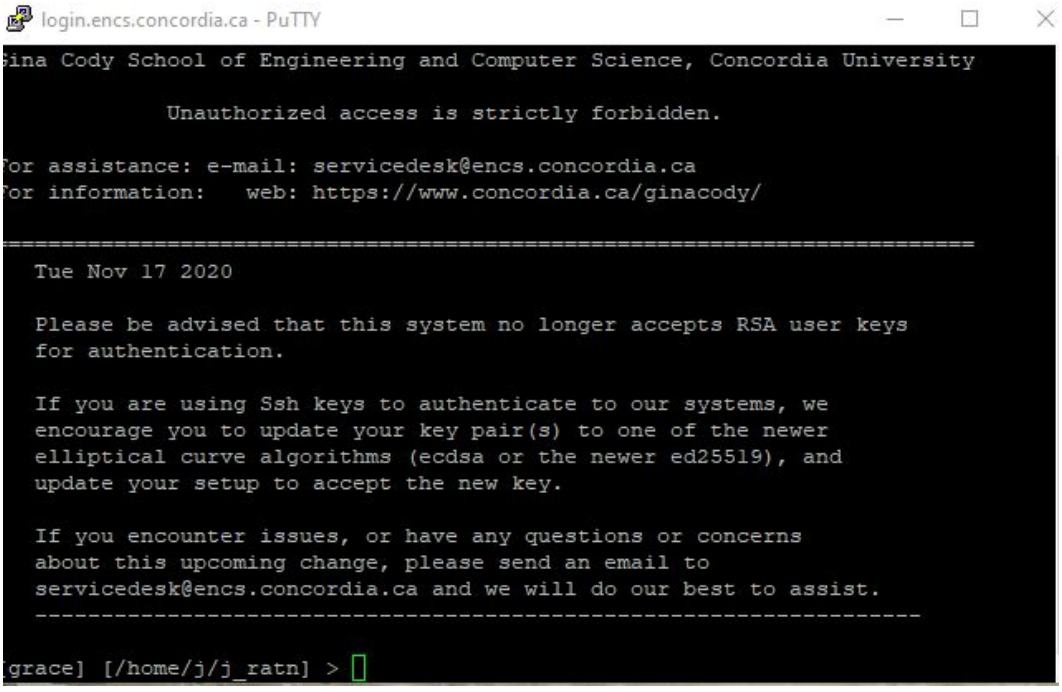
- You have setup Concordia VPN and are connected to it
- You have gotten the name of a available remote real-time workstation from the webpage: https://users.encs.concordia.ca/~realtime/workstation/check_rt_ws.html
- You have installed the latest version of Putty

Connect to Remote Desktop:

- Run putty and setup the SSH Tunnel to the ENCS machine
    - Run Putty
    - Under "SSH"-> Tunnels -> check both the boxes of "Local ports accept connections from other hosts" and "Remote ports do the same"
    - Under "SSH"-> Tunnels -> enter source port as "3391" and the destination with the name of the available workstation you have.
        - E.g. "zishu.encs.concordia.ca:3389"
    - Click “Add”

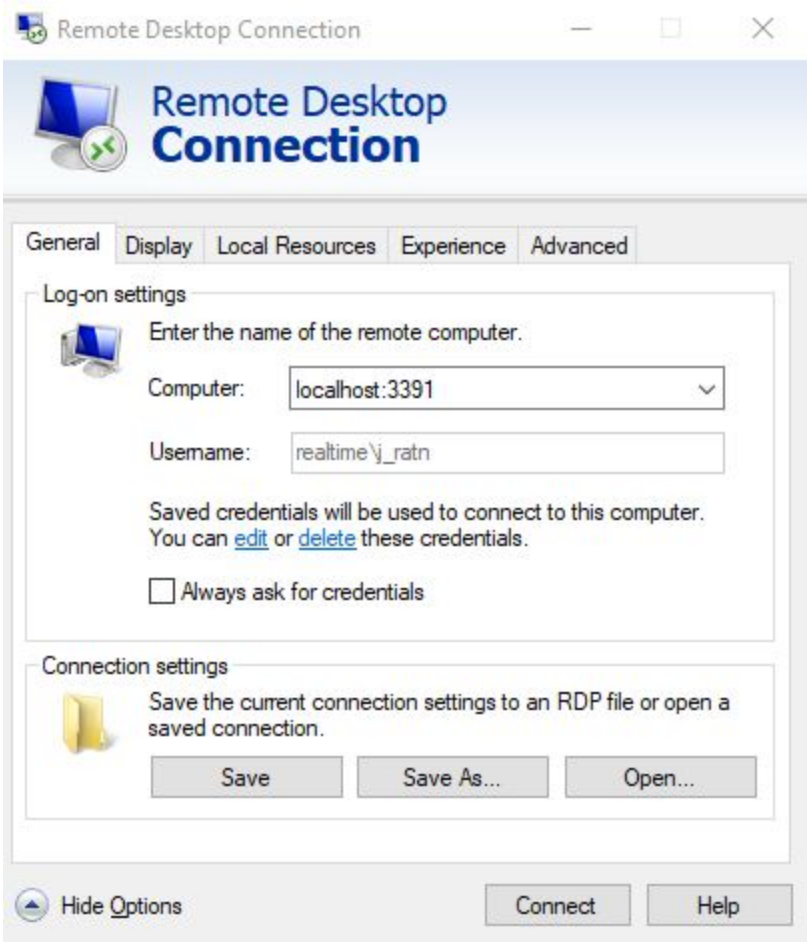


- Under "Session" -> Host Name as login.encs.concordia.ca, Port 22 -> “Open”
- Enter your ENCS username and password



- Minimize window

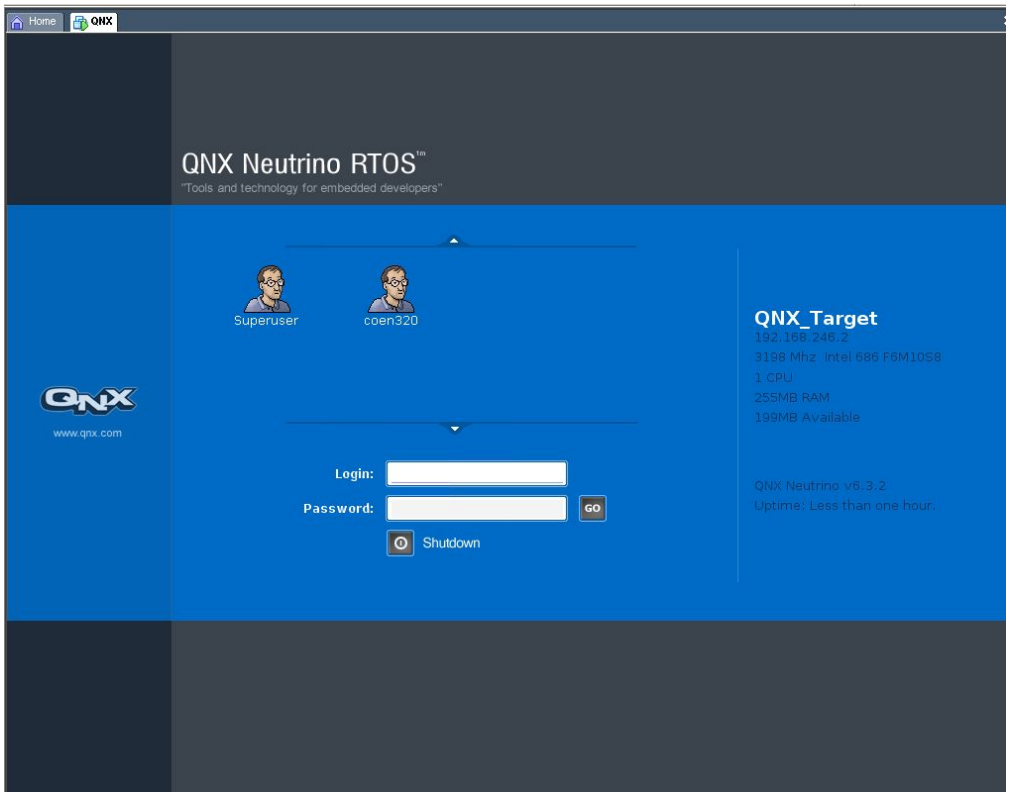
- Connect to remote desktop:
    - Open RDP
        - Search “RDP” in the start menu
        - Run RDP
    - Enter "localhost:3391" , click ‘Show Options” , input “realtime\your_realtime_id”, press “Connect”.
        - The realtime id is your ENCS id



    - Enter your realtime password
- You are now connected to a real-time workstation

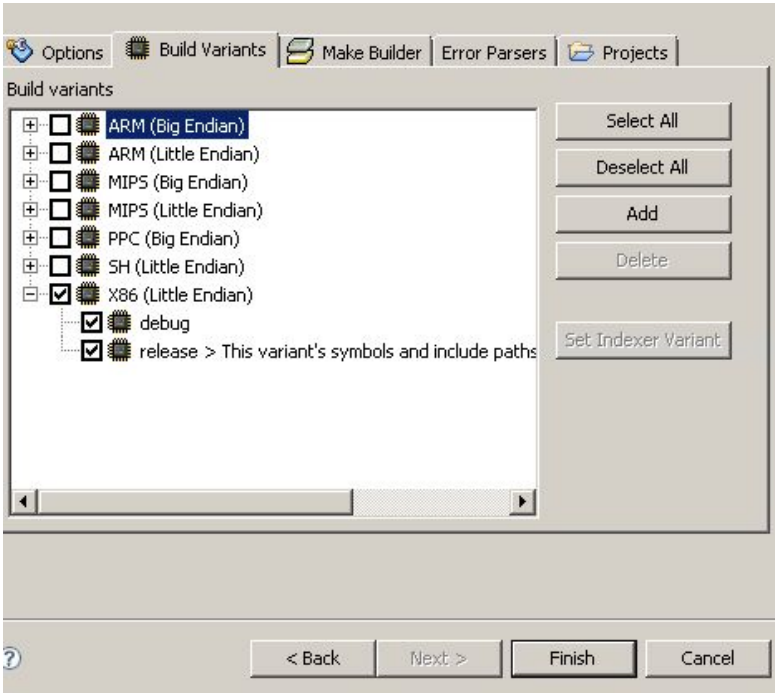
Run Vmware:

- Search “Vmware” in the start menu -> run Vmware -> “QNX” → “Start this virtual machine”
- Let the VM get setup, until login is prompted
    - No login is required
- The VM is now activated
- Minimize window



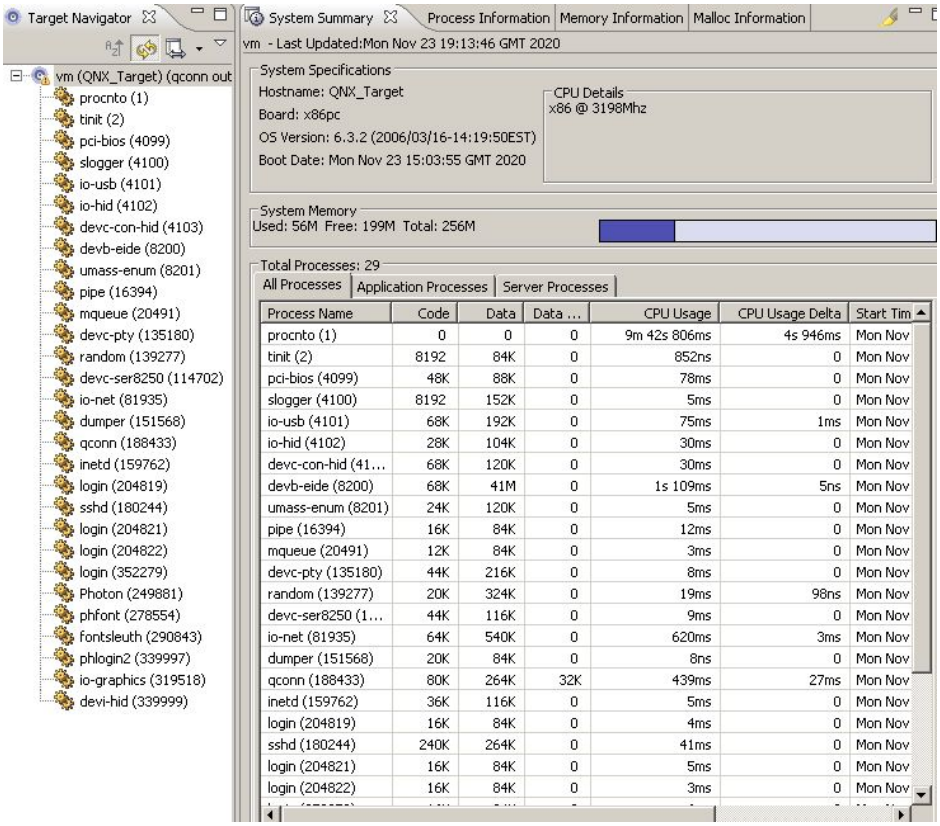
Run Real-Time Monitoring Program:

- Click the QNX Momentics IDE icon on the workstation desktop to start the IDE.
- Create the project:
    - Select “New” → QNX C++ Project → name project → click “next”
    - Choose “x86” in the Build Variants tab to get a PC as target → click “finish”



    - Replace the .cc file in the new project with the COEN320Project.cc included in this zip file

- Create a target project:
    - Window → Open Perspective → QNX System Information
    - Press the right mouse button → select New QNX Target → name the target system → enter the VM’s IP address “192.168.246.2” → click finish
    - Return to c++ perspective



- Place dataset in the Virtual machine target:
    - Open Target File System Navigator.
        - From the main menu, select Window > Show View > Other > Select QNX Targets > double-click Target File System Navigator.
    - Place the dataset.txt in the tmp folder in the Target File System Navigator
        - Drag from extracted zip file to tmp folder in Target File System Navigator
- Compile program:
    - Right-click the project name in the C/C++ Projects view → “Properties” → “QNX C/C++ Project” → “Build Variants” → “+” symbol next to x86 → check both the debug and release variants.
    - Right-click the project name → Build Project
- Run program:
    - Open the dropdown menu beside the Run icon and choose “run..”
    - In popup window:
        - Select “C/C++ QNX QConn (IP) ” → “New launch configuration”
        - On the main tab:
            - “Search project” → choose binary of the project without “_g” → “Ok” → “Apply” → “Run”

- You can view the result in the console:
    - The console displays the data from the sensors periodically, infinitely.
    - The program will continue getting the data from the sensors for ever, hence once it gets at the end of the dataset it will continue reading the data from the sensors on the last line infinitely
    - Terminate the execution when this happens or at any desired time.
    - 

    - The display gets updated every 0.5 seconds by default, the period of the display can be changed on line 21 of the code to be any desired period in microseconds.
    - ```
      #include <string>

      //period of the CSV reader
      #define SleepTime 1
      //period of the thread that displays the variables in microsecond
      #define DispTime 500000 //1000000
      //buffer to save variables
      #define BUFFER_SIZE 9

      std::string buffer[BUFFER_SIZE];
      ```

Reference:

1. https://users.encs.concordia.ca/~realtime/docs/ECE_Real-time_QNX_LAB_%20User_Guide.pdf
      ```