



UoP Racing Team

UoP7e Racecar

2019

Part Name	TELEMETRY
Section	POWERTRAIN

TELEMETRY SHARING SOFTWARE

Introduction

A custom software was created in order to share the live telemetry data received from **UoP7e Racecar** to all team members in order to have an immediate update of the situation of the car and don't rely on one single PC. For example the cooling sub-team can watch the temperature by themselves instead of constantly asking the member which operated the host PC.

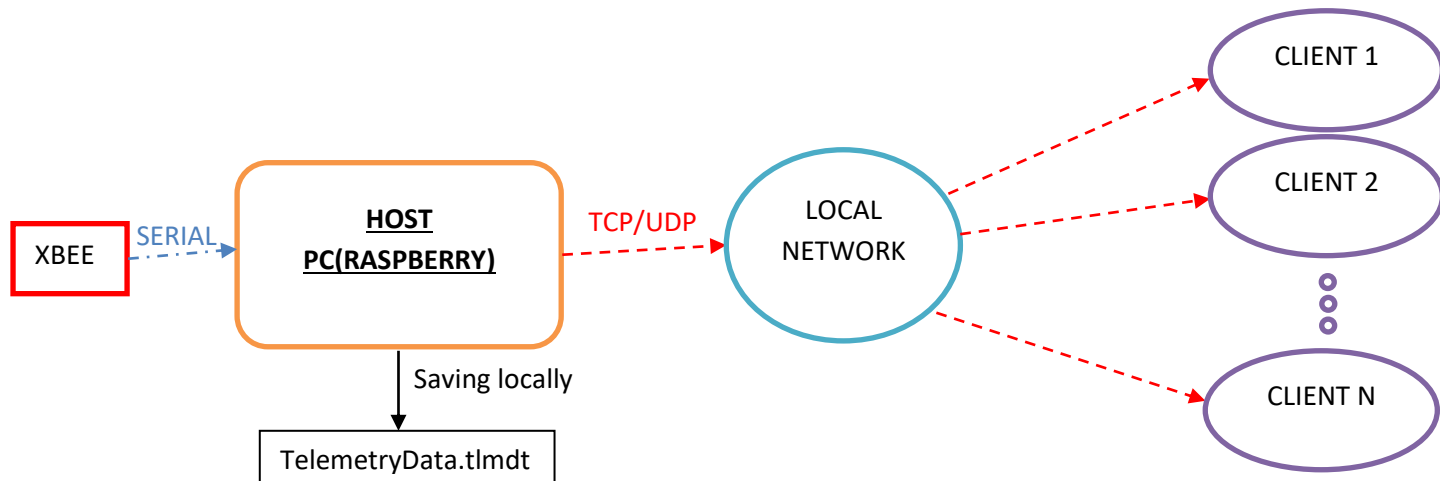


Figure 1. Block Structure of software

Structure

The **XBEE** module receives wirelessly the telemetry data from the car ,then it sends them via Serial to the COM(USB) port of the host. The server running on the host is responsible for 3 things:

1. Read Serial data from XBEE
2. Save them locally to “TelemetryData.tlmdt” for later use
3. Send them to the clients

1.

After scanning of ports the user selects the one attached to Xbee ,the Serial protocol is set at a baud rate of 115200 and the connection starts.The data are read at chunks of 20 Bytes(after testing,this value resulted to the best transmit rate)

2.

The data are received locally in raw form to the “TelemetryData.tlmdt” which is going to be used for processing and debugging later.

3.

Each client must know the local IP Address of the host (it is shown in the execution of the server.py).A predefined **Socket port:1234** has been selected for this software.After a TCP

connection is established between server and client ,the client can now receive data .Then the data are transmitted to a virtual COM(Serial) Port ,in the same port the **TELEMERTY GUI** (created by Manos) is going to be listening and finally receive the data from the car.

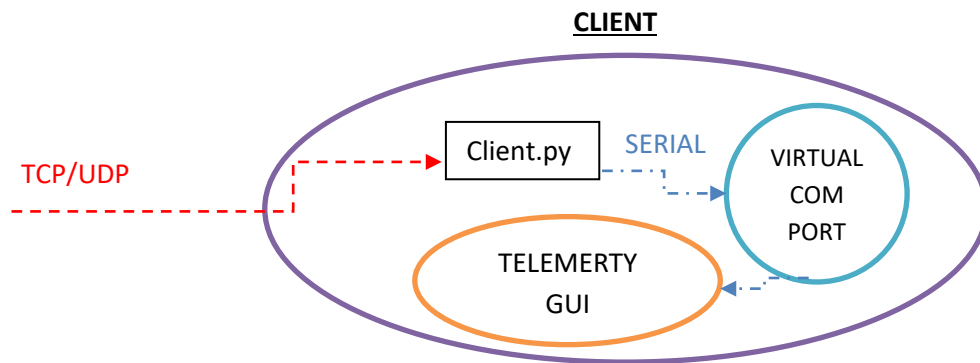


Figure 2. Block Structure of Client

Instructions

Host:

Run **server.py** ,select from the list and write down the COM port which is going to be used.

Client:

Run **client.py** ,enter the IP of server (it is shown in the server side program in the first line) and select from the list and write down the **VIRTUAL** COM port which is going to be used for connection with the **TELEMERTY GUI**.

Managing Virtual COM Ports

Windows

The [Virtual Serial Port Tools](#) software is used to create 2 virtual COM Ports and connect them together,since Windows won't allow 2 applications access the same Serial port at the same time.The software is free with the only limitation that you cant choose the name of the create Ports and are not permanent after turn off.(e.g The software transmits to **COM1** which is connected via the local bridge to **COM2** and **TELEMERTY GUI** listens to **COM2**)

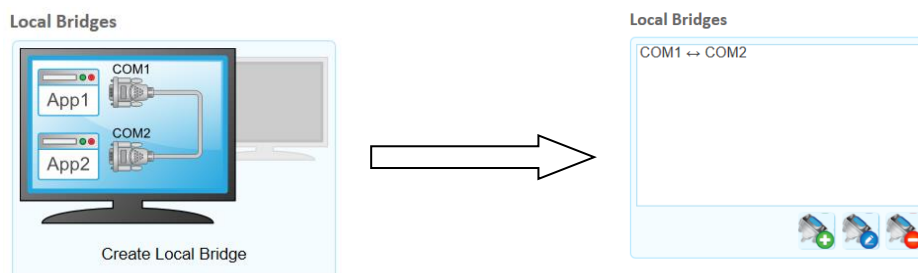


Figure 3.Example of Bridge between 2 Virtual COM Ports

Linux

Haven't been able to test to Linux yet but after researching [socat](#) seems to be solving the problem.