**Perun kernel**

Description

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

This paper describes in detail an application which acts as a kernel of first UFP UAVs.

Jakub Mnich – MikroCpp (Universal Flying Platforms) 2 November 2016

1. **General description**

Perun provides the following functionalities to the system:

* FFDEKernel instance.
* Global logger for important messages.
* Internet / LAN communication gate.
* Command distribution server.

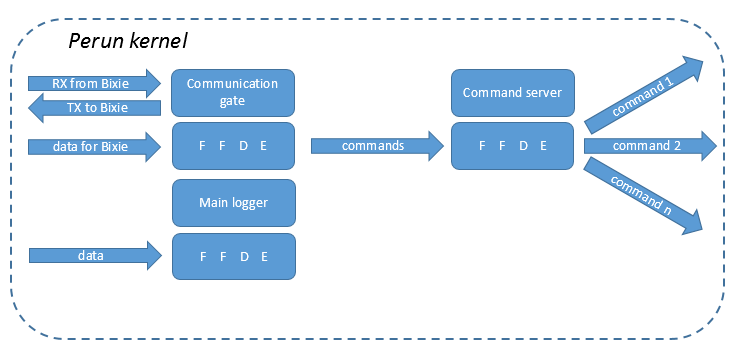
These services are automatically launched on startup without any additional initialization required. They provide the following functionalities:

Fig. - diagram

FFDEKernel **[1]**– required to launch FFDE Network on the UAV. Requires no attention from the user whatsoever.

Global logger – simple logger capable of saving small amount of important information: configuration messages, warnings, errors etc. It should not be used to store large amount of data generated on regular basis like sensor readings.

Global communication gate – allows to exchange data with Bixie Station after it binds with a socket created by the gate. When it receives commands it passes them to Global Command Server for redirection.

Global command server – directs commands from communication gate to right recipients. They are transmitted via FFDE Network.

Perun’s internal structure allows for simple integration of additional modules, shall any become necessary in future.

1. **Usage**

Before Perun can be used it needs to be started like any Java application. After that it initializes automatically.

Each of the Perun’s modules can be interfaced through FFDE. Accessing their functionalities is described in the following section.

1. Global logger

*FFDE ID: “mainLog”*

Any information sent to this logger through any FFDE Pipeline will automatically be recorder under the sender’s FFDE Server identifier.

Logger is controlled by its FFDE Master – it can execute two simple commands passed to it as first lines of messages from the master:

* “flushToGate”

Automatically sends all data stored in the logger to the global communication gate which retransmits it to a Bixie Station. First line of this message is “systemLogFlush”. It does not remove the log’s content.

* “clearLog”

This command destroys all data stored by the logger.

1. Global communication gate

*FFDE ID: “communicationGate”*

The gate automatically retransmits any information it gets through FFDE to the connected Bixie Station. If transmission is requested and no ground station is available the gate will put an error message in the global logger. No feedback is provided to the sender. That is why it is recommended to check the gate’s state first.

Message with this string in its first line “confirmGroundConnection” and an FFDE Node’s name in the second will not be retransmitted. Instead the gate will send a message to the node specified in the second line. The message would contain “communicationGate” string in its first line and “gateConnected” or “gateNotConnected” string in the second.

The gate automatically retransmits all incoming messages to the Global command server.

1. Global command server

*FFDE ID: “globalCommandServer”*

This module is responsible for directing information transmitted from ground to right recipients.

In order to receive data feed from the command server a recipient is required to register by creating FFDE Pipeline to the command server and transmitting through it desired command identifiers (one per message – only the first line of data will be extracted).

After the registration is completed the command server will provide automatic command feed through FFDE Pipeline created before.

**References:**

1. **Jakub Mnich** (2016) FFDE Network – Foundation For Data Exchange. *MikroCpp documentation repository*.