pyrealtime Documentation

Release 0.1.12

Eric Whitmire

Sep 09, 2020

Contents:

CHAPTER 1

Indices and tables

- genindex
- modindex
- search

CHAPTER 2

API

pyrealtime.decode_layer.**comma_decoder**(*data*)

A simple parser that splits data by commas

Parameters **data** (*string*) – data to parse

Returns numpy array of floats

Return type np.ndarray

class pyrealtime.serial_layer.**ByteSerialReadLayer**(**args*, *num_bytes=1*, *preamble=None*, ***kwargs*)

class pyrealtime.serial_layer.**SerialReadLayer**(*baud_rate*, *device_name*, **args*, *auto_reconnect=True*, ***kwargs*)

Reads data from a serial port

Parameters

- **baud_rate** – Baud rate or serial port (e.g. 9600, 115200, etc). See pyserial documentation for more details
- **device_name** – Full or partial name of the device (e.g. 'COM2' or 'Arduino'). The port will be obtained using *find_serial_port()*.

classmethod **from_port**(*serial*, **args*, ***kwargs*)

Creates a layer from an existing serial object

Parameters **serial** – Serial port object, either created using pyserial or from *find_serial_port()*.

class pyrealtime.serial_layer.**SerialWriteLayer**(*port_in*, *baud_rate*, *device_name*, **args*, *auto_reconnect=True*, ***kwargs*)

Sends data to a serial port

Parameters

- **port_in** – Source of data to send
- **baud_rate** – Baud rate or serial port (e.g. 9600, 115200, etc). See pyserial documentation for more details

- **device_name** – Full or partial name of the device (e.g. 'COM2' or 'Arduino'). The port will be obtained using *find_serial_port()*.

classmethod from_port(*port_in*, *serial*, **args*, ***kwargs*)

Creates a layer from an existing serial object

Parameters **serial** – Serial port object, either created using pyserial or from *find_serial_port()*.

pyrealtime.serial_layer.**find_serial_port**(*name*)

Utility function to scan available serial ports by name. The returned object is intended to be passed to the *from_port()* constructor of *SerialReadLayer* or *SerialWriteLayer*.

Parameters **name** – The name of the serial port to scan for. It will return the first available port containing name.

Returns A closed serial port object

Python Module Index

p

Index