

Zeo Raw Data API

BaseLink

Listens to the data coming from the serial port connected to Zeo.

The serial port is set at baud 38400, no parity, one stop bit. Data is sent Least Significant Byte first.

The serial protocol is:

- AncllLLTtsid
 - A is a character starting the message
 - n is the protocol “version”, ie “4”
 - c is a one byte checksum formed by summing the identifier byte and all the data bytes
 - ll is a two byte message length sent LSB first. This length includes the size of the data block plus the identifier.
 - LL is the inverse of ll sent for redundancy. If ll does not match ~LL, we can start looking for the start of the next block immediately, instead of reading some arbitrary number of bytes, based on a bad length.
 - T is the lower 8 bits of Zeo’s unix time.
 - tt is the 16-bit sub-second (runs through 0xFFFF in 1second), LSB first.
 - s is an 8-bit sequence number.
 - i is the datatype
 - d is the array of binary data

The incoming data is cleaned up into packets containing a timestamp, the raw data output version, and the associated data.

External code can be sent new data as it arrives by adding themselves to the callback list using the `addCallback` function. It is suggested, however, that external code use the `ZeoParser` to organize the data into events and slices of data.

class `ZeoRawData.BaseLink.BaseLink(port)`

Runs on a separate thread and handles the raw serial communications and parsing required to communicate with Zeo. Each time data is successfully received, it is sent out to all of the callbacks.

addCallback(callback)

Add a callback that will be passed valid data.

error(*msg*)

Function for error printing.

run()

Begins thread execution and raw serial parsing.

Parser

This module parses data from the BaseCapture module and assembles them into slices that encompass a range of data representative of Zeo's current status.

There are two different callbacks. One for slice callbacks and one that the module will pass events to.

class ZeoRawData.Parser.Parser

Interprets the incoming Zeo data and encapsulates it into an easy to use dictionary.

addEventCallback(*callback*)

Add a function to call when an Event has occurred.

addSliceCallback(*callback*)

Add a function to call when a Slice of data is completed.

clearSlice()

Resets the current Slice

update(*timestamp, timestamp_subsec, version, data*)

Update the current Slice with new data from Zeo. This function is setup to be easily added to the BaseLink's callbacks.

Utility

A collection of general purpose functions and datatypes.

ZeoRawData.Utility.dataTypes

A dictionary containing all types of data the base may send.

0x00 : *'Event'*

An event has occurred

0x02 : *'SliceEnd'*

Marks the end of a slice of data

0x03 : *'Version'*

Version of the raw data output

0x80 : *'Waveform'*

Raw time domain brainwave

0x83 : *'FrequencyBins'*

Frequency bins derived from waveform

0x84 : *'SQI'*

Signal Quality Index of waveform (0-30)

0x8A : *'ZeoTimestamp'*

Timestamp from Zeo's RTC

0x97 : *'Impedance'*

Impedance across the headband

0x9C : *'BadSignal'*

Signal contains artifacts

0x9D : *'SleepStage'*

Current 30sec sleep stage

`ZeoRawData.Utility.eventTypes`

A dictionary containing all the types of event that may be fired.

0x05 : *'NightStart'*

User's night has begun

0x07 : *'SleepOnset'*

User is asleep

0x0E : *'HeadbandDocked'*

Headband returned to dock

0x0F : *'HeadbandUnDocked'*

Headband removed from dock

0x10 : *'AlarmOff'*

User turned off the alarm

0x11 : *'AlarmSnooze'*

User hit snooze

0x13 : *'AlarmPlay'*

Alarm is firing

0x15 : *'NightEnd'*

User's night has ended

0x24 : *'NewHeadband'*

A new headband ID has been read

`ZeoRawData.Utility.filter60hz(A)`

Filters out 60hz noise from a signal. In practice it is a sinc low pass filter with cutoff frequency of 50hz.

`ZeoRawData.Utility.frequencyBins`

A dictionary containing all of the frequency bins.

0x00 : '2-4'

Delta

0x01 : '4-8'

Theta

0x02 : '8-13'

Alpha

0x03 : '13-18'

Beta

0x04 : '18-21'

Beta

0x05 : '11-14'

Beta (sleep spindles)

0x06 : '30-50'

Gamma

`ZeoRawData.Utility.getInt16(A)`

Creates a signed 16bit integer from a 2 item array

`ZeoRawData.Utility.getInt32(A)`

Creates a signed 32bit integer from a 4 item array

`ZeoRawData.Utility.getUInt16(A)`

Creates an unsigned 16bit integer from a 2 item array

`ZeoRawData.Utility.getUInt32(A)`

Creates an unsigned 32bit integer from a 4 item array

`ZeoRawData.Utility.sleepStages`

A dictionary containing the sleepstages output by the base.

0x00 : 'Undefined'

Sleepstage unsure

0x01 : *'Awake'*

Awake

0x02 : *'REM'*

Rapid eye movement(possibly dreaming)

0x03 : *'Light'*

Light sleep

0x04 : *'Deep'*

Deep sleep