

---

# **ligotimegps**

***Release 2.0.1***

**Apr 25, 2019**



---

## Contents

---

<b>1</b>	<b>Installation</b>	<b>3</b>
1.1	Pip . . . . .	3
1.2	Conda . . . . .	3
<b>2</b>	<b>Classes</b>	<b>5</b>
2.1	LIGOTimeGPS . . . . .	5
<b>3</b>	<b>Support</b>	<b>7</b>
	<b>Python Module Index</b>	<b>9</b>



*ligotimegps* provides a pure-python version of the `lal.LIGOTimeGPS` object, used to represent GPS times (number of seconds elapsed since GPS epoch) with nanoseconds precision.

The code provided here is much slower than the C-implementation provided by LAL, so if you really care about performance, don't use this module.

**Documentation contents:**

- *Installation*
- *Classes*
- *Support*



# CHAPTER 1

---

## Installation

---

### 1.1 Pip

```
python -m pip install ligotimegps
```

Supported python versions: 2.7, 3.4+.

### 1.2 Conda

```
conda install -c conda-forge ligotimegps
```

Supported python versions: 2.7, 3.5+.





---

`LIGOTimeGPS(seconds[, nanoseconds])`An object for storing times with nanosecond resolution

---

## 2.1 LIGOTimeGPS

**class** `ligotimegps.LIGOTimeGPS` (*seconds*, *nanoseconds*=0)Bases: `object`

An object for storing times with nanosecond resolution

Internally the time is represented as a signed integer *gpsSeconds* part and an unsigned integer *gpsNanoseconds* part. The actual time is always constructed by adding the nanoseconds to the seconds. So -0.5 s is represented by setting *seconds* = -1, and *nanoseconds* to 500000000.

### Parameters

- **seconds** (*int*, *str*) – the count of seconds
- **nanoseconds** (*int*, *str*, optional) – the count of nanoseconds

### Examples

```
>>> LIGOTimeGPS(100.5)
LIGOTimeGPS(100, 500000000)
>>> LIGOTimeGPS("100.5")
LIGOTimeGPS(100, 500000000)
>>> LIGOTimeGPS(100, 500000000)
LIGOTimeGPS(100, 500000000)
>>> LIGOTimeGPS(0, 10050000000)
LIGOTimeGPS(100, 500000000)
>>> LIGOTimeGPS(100.2, 300000000)
LIGOTimeGPS(100, 500000000)
>>> LIGOTimeGPS("0.000000001")
```

(continues on next page)

(continued from previous page)

```
LIGOTimeGPS(0, 1)
>>> LIGOTimeGPS("0.0000000012")
LIGOTimeGPS(0, 1)
>>> LIGOTimeGPS("0.0000000018")
LIGOTimeGPS(0, 2)
>>> LIGOTimeGPS("-0.8")
LIGOTimeGPS(-1, 200000000)
>>> LIGOTimeGPS("-1.2")
LIGOTimeGPS(-2, 800000000)
```

## Attributes Summary

<i>gpsNanoSeconds</i>	residual nanoseconds
<i>gpsSeconds</i>	Seconds since 0h UTC 6 Jan 1980
<i>nanoseconds</i>	residual nanoseconds
<i>seconds</i>	Seconds since 0h UTC 6 Jan 1980

## Methods Summary

<i>ns()</i>	Convert a <i>LIGOTimeGPS</i> to a count of nanoseconds as an int
-------------	--

## Attributes Documentation

### **gpsNanoSeconds**

residual nanoseconds

### **gpsSeconds**

Seconds since 0h UTC 6 Jan 1980

### **nanoseconds**

residual nanoseconds

### **seconds**

Seconds since 0h UTC 6 Jan 1980

## Methods Documentation

### **ns()**

Convert a *LIGOTimeGPS* to a count of nanoseconds as an int

When running python2.7 on Windows this is returned as `numpy.long` to guarantee long-ness.

## Examples

```
>>> LIGOTimeGPS(100.5).ns()
100500000000
```

## CHAPTER 3

---

### Support

---

To ask a question, report an issue, or suggest a change, please [open a ticket on GitHub](#).



---

## Python Module Index

---

|

ligotimegps, [1](#)



## G

gpsNanoSeconds (*ligotimegps.LIGOTimeGPS* attribute), 6

gpsSeconds (*ligotimegps.LIGOTimeGPS* attribute), 6

## L

LIGOTimeGPS (*class in ligotimegps*), 5

ligotimegps (*module*), 1

## N

nanoseconds (*ligotimegps.LIGOTimeGPS* attribute), 6

ns () (*ligotimegps.LIGOTimeGPS* method), 6

## S

seconds (*ligotimegps.LIGOTimeGPS* attribute), 6