

---

# **asyncserial Documentation**

***Release 0.1.0b3***

**Matthias Riegler**

**Apr 22, 2018**



---

## Contents

---

<b>1</b>	<b>Installation</b>	<b>3</b>
<b>2</b>	<b>Content</b>	<b>5</b>
2.1	asyncserial package . . . . .	5
2.1.1	asyncserial.Serial . . . . .	5
<b>3</b>	<b>Examples</b>	<b>7</b>



*asyncserial* is a wrapper for the *pyserial* library providing an async interface based on *async def* and *await*.



# CHAPTER 1

---

## Installation

---

```
pip install asyncserial
```





## CHAPTER 2

---

### Content

---

### 2.1 `asyncserial` package

#### 2.1.1 `asyncserial.Serial`



## CHAPTER 3

---

### Examples

---

```
import asyncio
from asyncserial import Serial

loop = asyncio.get_event_loop()

test_serial = Serial(loop, "/dev/ttyACM0", baudrate=115200)

async def test():
    await test_serial.read() # Drop anything that was already received
    while True:
        line = await test_serial.readline() # Read a line
        print("[+] Serial read: {}".format(line))
        await asyncio.sleep(0) # Let's be a bit greedy, should be adjust to your needs

asyncio.ensure_future(test())

print("[+] Starting eventloop")
loop.run_forever()
```