

---

# Adafruit RTTTL Library Documentation

*Release 1.0*

**Scott Shawcroft**

**Jan 20, 2021**



---

## Contents

---

<b>1</b>	<b>Dependencies</b>	<b>3</b>
<b>2</b>	<b>Installing from PyPI</b>	<b>5</b>
<b>3</b>	<b>Usage Example</b>	<b>7</b>
<b>4</b>	<b>CPX Usage Example</b>	<b>9</b>
<b>5</b>	<b>Contributing</b>	<b>11</b>
<b>6</b>	<b>Documentation</b>	<b>13</b>
<b>7</b>	<b>Table of Contents</b>	<b>15</b>
7.1	Simple test .....	15
7.2	adafruit_rtttl .....	16
<b>8</b>	<b>Indices and tables</b>	<b>17</b>
	<b>Python Module Index</b>	<b>19</b>
	<b>Index</b>	<b>21</b>



This plays `RTTTL` melodies.



# CHAPTER 1

---

## Dependencies

---

This driver depends on:

- [Adafruit CircuitPython](#)
- [Adafruit CircuitPython Waveform](#)

Please ensure all dependencies are available on the CircuitPython filesystem. This is easily achieved by downloading the [Adafruit library and driver bundle](#).





## CHAPTER 2

---

### Installing from PyPI

---

On supported GNU/Linux systems like the Raspberry Pi, you can install the driver locally [from PyPI](#). To install for current user:

```
pip3 install adafruit-circuitpython-rtttl
```

To install system-wide (this may be required in some cases):

```
sudo pip3 install adafruit-circuitpython-rtttl
```

To install in a virtual environment in your current project:

```
mkdir project-name && cd project-name
python3 -m venv .env
source .env/bin/activate
pip3 install adafruit-circuitpython-rtttl
```



## CHAPTER 3

---

### Usage Example

---

This plays Frosty the Snowman:

```
import board
import adafruit_rtttl

adafruit_rtttl.play(board.A0, "Snowman:d=8,o=5,b=200:2g,4e.,f,4g,2c6,b,c6,4d6,4c6,4b,
↪a,2g.,b,c6,4d6,4c6,4b,a,a,g,4c6,4e.,g,a,4g,4f,4e,4d,2c.,4c,4a,4a,4c6,4c6,4b,4a,4g,
↪4e,4f,4a,4g,4f,2e.,4e,4d,4d,4g,4g,4b,4b,4d6,d6,b,4d6,4c6,4b,4a,4g,4p,2g")
```



## CHAPTER 4

---

### CPX Usage Example

---

This plays Frosty the Snowman on a Circuit Playground Express (we must enable onboard speaker):

```
import board
from digitalio import DigitalInOut, Direction
import adafruit_rtttl

spkrenable = DigitalInOut(board.SPEAKER_ENABLE)
spkrenable.direction = Direction.OUTPUT
spkrenable.value = True

adafruit_rtttl.play(board.A0, "Snowman:d=8,o=5,b=200:2g,4e.,f,4g,2c6,b,c6,4d6,4c6,4b,
↪a,2g.,b,c6,4d6,4c6,4b,a,a,g,4c6,4e.,g,a,4g,4f,4e,4d,2c.,4c,4a,4a,4c6,4c6,4b,4a,4g,
↪4e,4f,4a,4g,4f,2e.,4e,4d,4d,4g,4g,4b,4b,4d6,d6,b,4d6,4c6,4b,4a,4g,4p,2g")
```



## CHAPTER 5

---

### Contributing

---

Contributions are welcome! Please read our [Code of Conduct](#) before contributing to help this project stay welcoming.





## CHAPTER 6

---

### Documentation

---

For information on building library documentation, please check out [this guide](#).



---

Table of Contents

---

## 7.1 Simple test

Ensure your device works with this simple test.

Listing 1: examples/rtttl\_simpletest.py

```
1  # SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
2  # SPDX-License-Identifier: MIT
3
4  # The MIT License (MIT)
5  #
6  # Copyright (c) 2017 Scott Shawcroft for Adafruit Industries
7  #
8  # Permission is hereby granted, free of charge, to any person obtaining a copy
9  # of this software and associated documentation files (the "Software"), to deal
10 # in the Software without restriction, including without limitation the rights
11 # to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
12 # copies of the Software, and to permit persons to whom the Software is
13 # furnished to do so, subject to the following conditions:
14 #
15 # The above copyright notice and this permission notice shall be included in
16 # all copies or substantial portions of the Software.
17 #
18 # THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
19 # IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
20 # FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
21 # AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
22 # LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
23 # OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
24 # THE SOFTWARE.
25
26 # This is tested on the CircuitPlayground Express
27
```

(continues on next page)

(continued from previous page)

```
28 import digitalio
29 import board
30 import adafruit_rtttl
31
32 enable = digitalio.DigitalInOut(board.SPEAKER_ENABLE)
33 enable.switch_to_output(value=True)
34
35 adafruit_rtttl.play(
36     board.SPEAKER,
37     "itchy:d=8,o=6,b=160:c,a5,4p,c,a,4p,c,a5,c,a5,"
38     + "c,a,4p,p,c,d,e,p,e,f,g,4p,d,c,4d,f,4a#,4a,2c7",
39 )
40 adafruit_rtttl.play(
41     board.SPEAKER,
42     "Phantom:d=4,o=5,b=140:c,f,c,d#.,8c#,2c#,a#4,"
43     + "d#,8a#4,2c,c,f,c,d#.,8c#,2c#,a#4,d#.,8a#4,2c,p,c,f,g#,c.6,8a#,2a#,a#,d#.6,8a#,"
44     + "2c6,p,c6,2f.6,8d#6,8c#6,8c6,8a#,8g#,8g,8f,2e,c#,c#.,8c,2c",
45 )
```

## 7.2 adafruit\_rtttl

Play notes to a digitalio pin using ring tone text transfer language (rtttl).

- Author(s): Scott Shawcroft

`adafruit_rtttl.play` (*pin, rtttl, octave=None, duration=None, tempo=None*)

Play notes to a digitalio pin using ring tone text transfer language (rtttl). :param ~digitalio.DigitalInOut pin: the speaker pin :param rtttl: string containing rtttl :param int octave: represents octave number (default 6 starts at middle c) :param int duration: length of notes (default 4 quarter note) :param int tempo: how fast (default 63 beats per minute)

## CHAPTER 8

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



### **a**

`adafruit_rtttl`, [16](#)





## A

`adafruit_rtttl` (*module*), 16

## P

`play()` (*in module adafruit\_rtttl*), 16