# sparkfun_qwiic_rfid

*Release 2.0.0*

**SparkFun Electronics**

**Feb 25, 2021**

# CONTENTS:

Python module for the SparkFun RFID Qwiic Reader

This module is also compatible with the following products:

- SparkFun RFID Qwiic Kit

This python package is a port of the existing SparkFun Qwiic RFID Reader Arduino Library

This package can be used in conjunction with the overall SparkFun qwiic Python Package

New to qwiic? Take a look at the entire SparkFun qwiic ecosystem.

# ONE

# CONTENTS

- *Supported Platforms*
- *Dependencies*
- *Installation*
- *Documentation*
- *Example Use*

# SUPPORTED PLATFORMS

The Qwiic RFID Python package current supports the following platforms:

- Raspberry Pi
- NVidia Jetson Nano
- Google Coral Development Board

# DEPENDENCIES

This driver package depends on the qwiic I2C driver: Qwiic_I2C_Py

# DOCUMENTATION

The SparkFun Qwiic RFID module documentation is hosted at ReadTheDocs

# INSTALLATION

## 5.1 PyPi Installation

This repository is hosted on PyPi as the sparkfun-qwiic-rfid package. On systems that support PyPi installation via pip, this library is installed using the following commands

For all users (note: the user must have sudo privileges):

```
sudo pip install sparkfun-qwiic-rfid
```

For the current user:

```
pip install sparkfun-qwiic-rfid
```

To install, make sure the setuptools package is installed on the system.

Direct installation at the command line:

```
python setup.py install
```

To build a package for use with pip:

```
python setup.py sdist
```

A package file is built and placed in a subdirectory called dist. This package file can be installed using pip.

```
cd dist
pip install sparkfun-qwiic-rfid-<version>.tar.gz
```

# EXAMPLE USE

See the examples directory for more detailed use examples.

```python
from __future__ import print_function
import qwiic_rfid
import time
import sys

def run_example():

    print("\nSparkFun Qwiic RFID Reader Example 1")
    my_RFID = qwiic_rfid.Qwiic_RFID()

    if my_RFID.begin() == False:
        print("\nThe Qwiic RFID Reader isn't connected to the system. Please check
→your connection", file=sys.stderr)
        return

    print("\nReady to scan some tags!")

    while True:
        val = input("\nEnter 1 to get tag ID and scan time: ")

        if int(val) == 1:
            print("\nGetting your tag ID...")
            tag = my_RFID.get_tag()
            print("\nTag ID: " + tag)

            scan_time = my_RFID.get_prec_req_time()
            # If this time is too precise, try:
            # scan_time = my_RFID.get_req_time()
            print("\nScan Time: " + str(scan_time))

        time.sleep(0.02)

if __name__ == '__main__':
    try:
        run_example()
    except (KeyboardInterrupt, SystemExit) as exErr:
        print("\nEnding Example 1")
        sys.exit(0)
```

# TABLE OF CONTENTS

## 7.1 API Reference

### 7.1.1 qwiic_rfid

Python module for the Qwiic RFID Reader.

This python package is a port of the existing [SparkFun Qwiic RFID Arduino Library]([https://github.com/sparkfun/](https://github.com/sparkfun/) SparkFun_Qwiic_RFID_Arduino_Library)

This package can be used in conjunction with the overall [SparkFun qwiic Python Package]([https://github.com/](https://github.com/) sparkfun/Qwiic_Py)

New to qwiic? Take a look at the entire [SparkFun qwiic ecosystem]([https://www.sparkfun.com/qwiic](https://www.sparkfun.com/qwiic)).

**class** qwiic_rfid.**QwiicRFID**(*address=None*, *i2c_driver=None*)

>**Parameters**
>
>>- **address** – The I2C address to use for the device. If not provied, the default address is used.
>>
>>- **i2c_driver** – An existing i2c driver object. If not provided a driver object is created.
>
>**Returns** The RFID device object.
>
>**Return type** Object

**begin**()
>Initialize the operation of the Qwiic GPIO
>
>>**Returns** Returns true if the initialization was successful, otherwise False.
>>
>>**Return type** void

**change_address**(*new_address*)
>Changes the I2C address of the Qwiic RFID reader
>
>>**Parameters** **new_address** – the new address to set the RFID reader to
>>
>>**Return type** bool

**clear_tags**()
>Reads and clears the tags from the buffer
>
>>**Return type** void - does not return anything

**get_all_prec_times**(*time_array*)
>Gets all times in the buffer

Parameters **time_array** – list of upto 20 times the RFID tag was read from the I2C bus

**Return type** void - does not return anything

**get_all_tags**(*tag_array*)
Gets all the tags in the buffer

Parameters **tag_array** – list of upto 20 RFID tag numbers

**Return type** void - does not return anything

**get_prec_req_time**()
Gets the time when the RFID tag was last scanned

**Returns** Returns time in seconds

**Return type** int

**get_req_time**()
Gets the time when when RFID tag was last scanned

**Returns** Returns time in seconds

**Return type** int

**get_tag**()
Gets the current RFID tag

**Returns** Returns the RFID tag

**Return type** string

**is_connected**()
Determine if a Qwiic RFID device is connected to the system.

**Returns** True if the device is connected, otherwise False.

**Return type** void

**class** qwiic_rfid.**QwiicRFID**(*address=None*, *i2c_driver=None*)

**Parameters**

- **address** – The I2C address to use for the device. If not provied, the default address is used.

- **i2c_driver** – An existing i2c driver object. If not provided a driver object is created.

**Returns** The RFID device object.

**Return type** Object

**begin**()
Initialize the operation of the Qwiic GPIO

**Returns** Returns true if the initialization was successful, otherwise False.

**Return type** void

**change_address**(*new_address*)
Changes the I2C address of the Qwiic RFID reader

Parameters **new_address** – the new address to set the RFID reader to

**Return type** bool

**clear_tags**()
Reads and clears the tags from the buffer

---

**Return type** void - does not return anything

**get_all_prec_times**(*time_array*)
Gets all times in the buffer

**Parameters** **time_array** – list of upto 20 times the RFID tag was read from the I2C bus

**Return type** void - does not return anything

**get_all_tags**(*tag_array*)
Gets all the tags in the buffer

**Parameters** **tag_array** – list of upto 20 RFID tag numbers

**Return type** void - does not return anything

**get_prec_req_time**()
Gets the time when the RFID tag was last scanned

**Returns** Returns time in seconds

**Return type** int

**get_req_time**()
Gets the time when when RFID tag was last scanned

**Returns** Returns time in seconds

**Return type** int

**get_tag**()
Gets the current RFID tag

**Returns** Returns the RFID tag

**Return type** string

**is_connected**()
Determine if a Qwiic RFID device is connected to the system.

**Returns** True if the device is connected, otherwise False.

**Return type** void

## 7.2 Example 1

Listing 1: examples/qwiic_rfid_ex1_getTag.py

```python
# !/usr/bin/env python
# -----------------------------------------------------------------
# qwiic_rfid_ex1.py
#
# Basic example that reads tag and prints ID and the time requested
# when prompted by user.
# -----------------------------------------------------------------
#
# Written by Priyanka Makin @ SparkFun Electronics, January 2021
#
# This python library supports the SparkFun Electronics qwiic
# sensor/board ecosystem on a Raspberry Pi (and compatible) single
# board computers.
#
```

(continues on next page)

```
15   # More information on qwiic is at https://www.sparkfun.com/qwiic
16   #
17   # Do you like this library? Help support SParkFun. Buy a board!
18   # https://www.sparkfun.com/products/15191
19   #
20   # ==================================================================
21   # Copyright (c) 2021 SparkFun Electronics
22   #
23   # Permission is hereby granted, free of charge, to any person obtaining a copy
24   # of this software and associated documentation files (the "Software"), to deal
25   # in the Software without restriction, including without limitation the rights
26   # to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
27   # copies of the Software, and to permit persons to whom the Software is
28   # furnished to do so, subject to the following conditions:
29   #
30   # The above copyright notice and this permission notice shall be included in all
31   # copies or substantial portions of the Software.
32   #
33   # THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
34   # IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
35   # FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
36   # AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
37   # LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
38   # OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
39   # SOFTWARE.
40   #=================================================================================
41   # Example 1
42   #
43   # This example gets the latest tag scanned and it's associated time from the
44   # Qwiic RFID Reader when the user enters "1" into the terminal. The
45   # "scan" time is not the time of the day the RFID card was scanned but rather the
46   # time between when the card was scanned and when you the user requested the RFID
47   # tag from the Qwiic RFID Reader.
48
49   from __future__ import print_function
50   import qwiic_rfid
51   import time
52   import sys
53
54   def run_example():
55
56       print("\nSparkFun Qwiic RFID Reader Example 1")
57       my_RFID = qwiic_rfid.QwiicRFID()
58
59       if my_RFID.begin() == False:
60           print("\nThe Qwiic RFID Reader isn't connected to the system. Please check␣
     ↪your connection", file=sys.stderr)
61           return
62
63       print("\nReady to scan some tags!")
64
65       while True:
66           val = input("\nEnter 1 to get tag ID and scan time: ")
67
68           if int(val) == 1:
69               print("\nGetting your tag ID...")
70               tag = my_RFID.get_tag()
```

```python
71          print("\nTag ID: " + tag)

73          scan_time = my_RFID.get_prec_req_time()
74          # If this time is too precise, try:
75          # scan_time = my_RFID.get_req_time()
76          print("\nScan Time: " + str(scan_time))


78      time.sleep(0.02)


80 if __name__ == '__main__':
81      try:
82          run_example()
83      except (KeyboardInterrupt, SystemExit) as exErr:
84          print("\nEnding Example 1")
85          sys.exit(0)
```

## 7.3 Example 2

Listing 2: examples/qwiic_rfid_ex2_getAllTags.py

```python
1  # !/usr/bin/env python
2  # -----------------------------------------------------------------
3  # qwiic_rfid_ex2.py
4  #
5  # Basic example that reads RFID tags and prints all the IDs and the
6  # times requested in the buffer when prompted by user.
7  # -----------------------------------------------------------------
8  #
9  # Written by Priyanka Makin @ SparkFun Electronics, January 2021
10 #
11 # This python library supports the SparkFun Electronics qwiic
12 # sensor/board ecosystem on a Raspberry Pi (and compatible) single
13 # board computers.
14 #
15 # More information on qwiic is at https://www.sparkfun.com/qwiic
16 #
17 # Do you like this library? Help support SParkFun. Buy a board!
18 # https://www.sparkfun.com/products/15191
19 #
20 # ==================================================================
21 # Copyright (c) 2021 SparkFun Electronics
22 #
23 # Permission is hereby granted, free of charge, to any person obtaining a copy
24 # of this software and associated documentation files (the "Software"), to deal
25 # in the Software without restriction, including without limitation the rights
26 # to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
27 # copies of the Software, and to permit persons to whom the Software is
28 # furnished to do so, subject to the following conditions:
29 #
30 # The above copyright notice and this permission notice shall be included in all
31 # copies or substantial portions of the Software.
32 #
33 # THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
34 # IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
```

```
35  # FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
36  # AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
37  # LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
38  # OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
39  # SOFTWARE.
40  #===============================================================================
41  # Example 2
42  #
43  # This example code demonstrates how to get every RFID card off of the Qwiic RFID
44  # reader. The Qwiic RFID reader can hold up to 20 cards and their scan times at a
45  # time. If more than 20 are read then the first card is overwritten. If you
46  # expect this to be the case then getting the scan time will help to see when the
47  # RFID card was scanned. A brief note about time. This time is not the time when the
48  # RFID card was scanned but the time between when the RFID card was scanned and it
49  # was requested by you, the user. There are two time functions available for large
50  # reads: getAllTimes() and getAllPrecTimes(), both return time in seconds but the
51  # second option gives you a time with two decimal point precision (0.00)
52
53  from __future__ import print_function
54  import qwiic_rfid
55  import time
56  import sys
57
58  def run_example():
59
60      print("\nSparkFun Qwiic RFID Example 2\n")
61      my_RFID = qwiic_rfid.QwiicRFID()
62
63      if my_RFID.begin() == False:
64          print("The Qwiic RFID Reader isn't connected to the system. Please check your␣
    ↪connection", \
65              file=sys.stderr)
66          return
67
68      print("\nReady to scan some tags!")
69
70      all_tags = [None] * my_RFID.MAX_TAG_STORAGE
71      all_times = [None] * my_RFID.MAX_TAG_STORAGE
72
73      while True:
74          val = input("\nEnter 1 to get all ID's and scan times: ")
75
76          if int(val) == 1:
77
78              my_RFID.get_all_tags(all_tags)
79              my_RFID.get_all_prec_times(all_times)
80
81              for i in range(0, my_RFID.MAX_TAG_STORAGE):
82                  print("\nRIFD Tag: " + all_tags[i])
83                  print("\nScan Time: " + str(all_times[i]))
84
85          time.sleep(0.02)
86
87  if __name__ == '__main__':
88      try:
89          run_example()
90      except (KeyboardInterrupt, SystemExit) as exErr:
```

```
91        print("\nEnding Example 2")
92        sys.exit(0)
```

## 7.4 Example 3

Listing 3: examples/qwiic_rfid_ex3_changeI2Caddress.py

```python
1   # !/usr/bin/env python
2   # -----------------------------------------------------------------
3   # qwiic_rfid_ex3.py
4   #
5   # Example that takes user input to change the I2C address of Qwiic RFID
6   # -----------------------------------------------------------------
7   #
8   # Written by Priyanka Makin @ SparkFun Electronics, January 2021
9   #
10  # This python library supports the SparkFun Electronics qwiic
11  # sensor/board ecosystem on a Raspberry Pi (and compatible) single
12  # board computers.
13  #
14  # More information on qwiic is at https://www.sparkfun.com/qwiic
15  #
16  # Do you like this library? Help support SParkFun. Buy a board!
17  # https://www.sparkfun.com/products/15191
18  #
19  # ==================================================================
20  # Copyright (c) 2021 SparkFun Electronics
21  #
22  # Permission is hereby granted, free of charge, to any person obtaining a copy
23  # of this software and associated documentation files (the "Software"), to deal
24  # in the Software without restriction, including without limitation the rights
25  # to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
26  # copies of the Software, and to permit persons to whom the Software is
27  # furnished to do so, subject to the following conditions:
28  #
29  # The above copyright notice and this permission notice shall be included in all
30  # copies or substantial portions of the Software.
31  #
32  # THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
33  # IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
34  # FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
35  # AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
36  # LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
37  # OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
38  # SOFTWARE.
39  #=====================================================================================
40  # Example 3
41  #
42  # This example code demonstrates how to change the address of the Qwiic RFID Tag
43  # Reader to one of your choosing. There is a set range of available addresses from
44  # 0x07 to 0x78, so make sure your chosen address falls within this range.
45
46  from __future__ import print_function
47  import qwiic_i2c
```

```python
48  import qwiic_rfid
49  import time
50  import sys
51
52  def run_example():
53
54      print("\nSparkFun Qwiic RFID Reader Example 3")
55      my_RFID = qwiic_rfid.QwiicRFID()
56
57      if my_RFID.begin() == False:
58          print("\nThe Qwiic RFID Reader isn't connected to the system. Please check
    ↪your connection", file=sys.stderr)
59          return
60
61      print("\nReady!")
62
63      print("\nEnter a new I2C address for the Qwiic RFID Reader to use.")
64      print("\nDon't use the 0x prefix. For instance if you wanted to")
65      print("\nchange the address to 0x5B, you would type 5B and hit enter.")
66
67      new_address = raw_input("\nNew Address: ")
68      new_address = int(new_address, 16)
69
70      # Check if the user entered a valid address
71      if new_address > 0x08 and new_address < 0x77:
72          print("\nCharacters received and new address valid!")
73          print("\nAttempting to set RFID reader address...")
74
75          my_RFID.change_address(new_address)
76          print("\nAddress successfully changed!")
77          # Check that the RFID Reader acknowledges on new address
78          time.sleep(0.02)
79          if my_RFID.begin() == False:
80              print("\nThe Qwiic RFID Reader isn't connected to the system. Please
    ↪check your connection", file=sys.stderr)
81
82          else:
83              print("\nRFID Reader acknowledged on new address!")
84
85      else:
86          print("\nAddress entered not a valid I2C address")
87
88  if __name__ == '__main__':
89      try:
90          run_example()
91      except (KeyboardInterrupt, SystemExit) as exErr:
92          print("\nEnding Example 3")
93          sys.exit(0)
```

# INDICES AND TABLES

- genindex
- modindex
- search

# PYTHON MODULE INDEX

## q