

Python_RPi

Introduction

There are three examples under this folder.

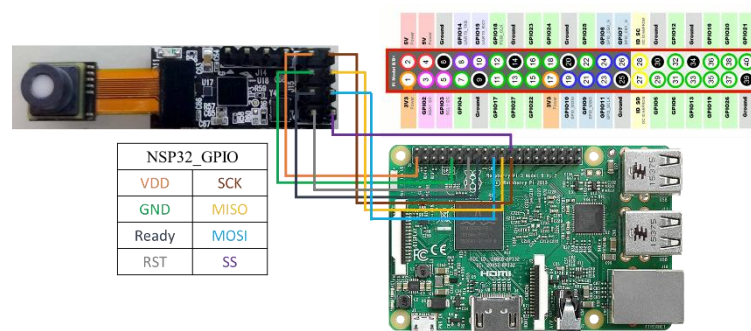
- 1) Beginner
A clean and simple example for beginners to start with NSP32, to demonstrate the basic usage of our API.
- 2) ConsoleDemo
A console program to demonstrate full functionalities of NSP32. Users can operate NSP32 by interactive console commands.
- 3) SpectrumMete
A GUI program to visualize the spectrum measurement using NSP32.

API Module Location

Our API module file is located at [/examples/NanoLambdaNSP32.py].

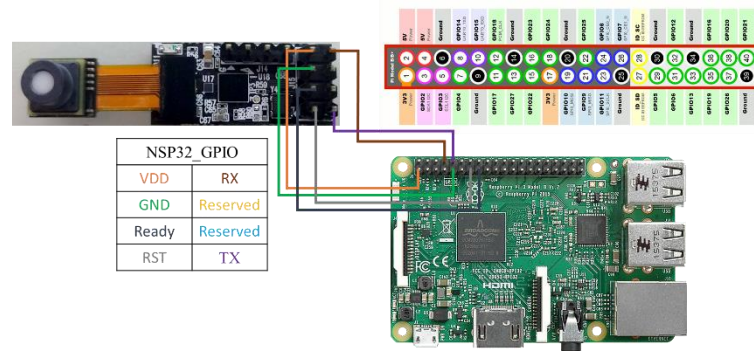
Hardware Setup

- 1) Tested on
 - Raspberry Pi 3 Model B with Rasbian Stretch version November 2018
 - Raspberry Pi Zero W with Rasbian Stretch version November 2018
- 2) Setup
 - i) Hardware connection with SPI



- ii) Hardware connection with UART

Python_RPi



The following table shows the general pin connections between NSP32 and RPi.

GPIO		Hardware	NSP32 Pin	RPi
		Power	VDD	VDD3V3
GND	GND		GND	
Data channel	SPI Signal	Wakeup/Reset	RST	GPIO27 (Pin 13)
		SPI SSEL	SS	SPI0_CE0_N
		SPI MOSI	MOSI	SPI0_MOSI
		SPI MISO	MISO	SPI0_MISO
		SPI SCK	SCK	SPI0_SCLK
		Ready	Ready	GPIO22 (Pin 15)
	UART Signal	Wakeup/Reset	RST	GPIO27 (Pin 13)
		UART RX	TX	UART0_RXD
		UART TX	RX	UART0_TXD
		Ready	Ready	GPIO22 (Pin 15)

Software Setup

1) Runs on

Python 3.5.3

2) Setup

i) Install Python 3.5 or above (Python 2 doesn't work).

Use the terminal command:

```
$sudo apt-get install python3
```

Note: The examples and the API utilizes the following modules, please make sure they are installed under your environment.

- RPi.GPIO [<https://pypi.org/project/RPi.GPIO/>]

Use the terminal command:

```
$sudo apt-get install python3-rpi.gpio
```

Python_RPi

- `spidev` [<https://pypi.org/project/spidev/>]
Use the terminal command:

```
$sudo apt-get install python3-spidev
```
 - `pySerial` [<https://pypi.org/project/pyserial/>]
Use the terminal command:

```
$sudo apt-get install python3-serial
```
- ii) Make sure to enable SPI on RPi by the following steps.
- i. Use the terminal command:

```
$ sudo raspi-config
```
 - ii. Under the Software Configuration Tool, select:
Interfacing Options → SPI → Yes → Ok.
- iii) Make sure to enable UART (disable serial login shell, enable serial interface) on RPi by the following steps.
- i. Use the terminal command:

```
$ sudo raspi-config
```
 - ii. Under the Software Configuration Tool, select:
Interfacing Options → Serial → No → Yes → Ok.

Notes

Source code modification is required if you want to change the pin numbers or the data channel type (default to SPI). For your convenience, we have marked these code sections with the title "modify this section to fit your need".

Run the Example

To run the examples, use Python commands:

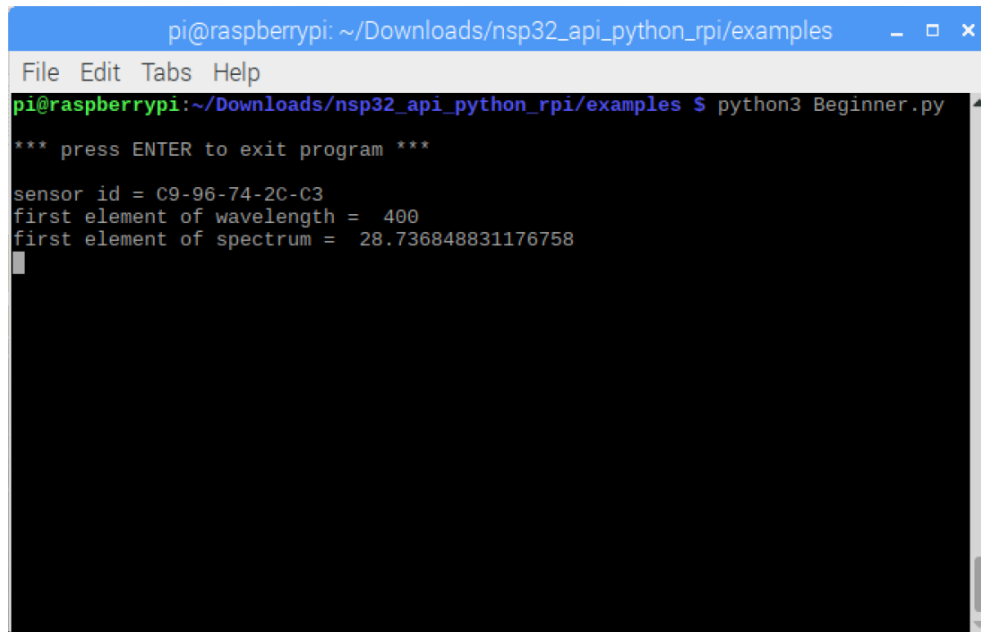
```
$ python3 Beginner.py
```

```
$ python3 ConsoleDemo.py
```

```
$ python3 SpectrumMeter.py
```

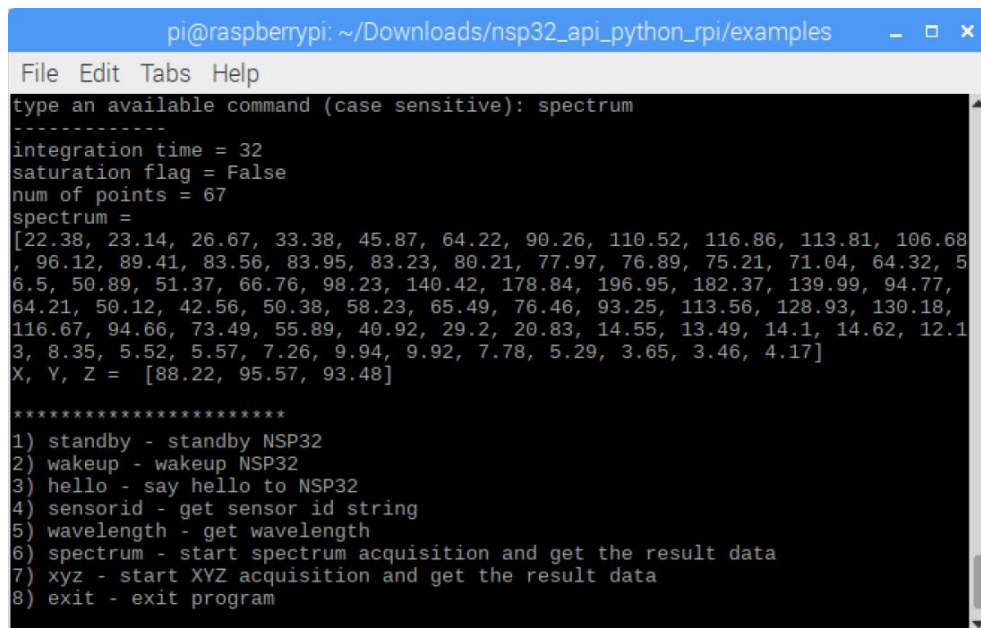
Python_RPi

1) Example Snapshot of “Beginner.py”



```
pi@raspberrypi: ~/Downloads/nsp32_api_python_rpi/examples
File Edit Tabs Help
pi@raspberrypi:~/Downloads/nsp32_api_python_rpi/examples $ python3 Beginner.py
*** press ENTER to exit program ***
sensor id = C9-96-74-2C-C3
first element of wavelength = 400
first element of spectrum = 28.736848831176758
```

2) Example Snapshot of “ConsoleDemo.py”



```
pi@raspberrypi: ~/Downloads/nsp32_api_python_rpi/examples
File Edit Tabs Help
type an available command (case sensitive): spectrum
-----
integration time = 32
saturation flag = False
num of points = 67
spectrum =
[22.38, 23.14, 26.67, 33.38, 45.87, 64.22, 90.26, 110.52, 116.86, 113.81, 106.68
, 96.12, 89.41, 83.56, 83.95, 83.23, 80.21, 77.97, 76.89, 75.21, 71.04, 64.32, 5
6.5, 50.89, 51.37, 66.76, 98.23, 140.42, 178.84, 196.95, 182.37, 139.99, 94.77,
64.21, 50.12, 42.56, 50.38, 58.23, 65.49, 76.46, 93.25, 113.56, 128.93, 130.18,
116.67, 94.66, 73.49, 55.89, 40.92, 29.2, 20.83, 14.55, 13.49, 14.1, 14.62, 12.1
3, 8.35, 5.52, 5.57, 7.26, 9.94, 9.92, 7.78, 5.29, 3.65, 3.46, 4.17]
X, Y, Z = [88.22, 95.57, 93.48]

*****
1) standby - standby NSP32
2) wakeup - wakeup NSP32
3) hello - say hello to NSP32
4) sensorid - get sensor id string
5) wavelength - get wavelength
6) spectrum - start spectrum acquisition and get the result data
7) xyz - start XYZ acquisition and get the result data
8) exit - exit program
```

Python_RPi

3) Example Snapshot of "SpectrumMeter.py"

