

# Using RFID Module with Raspberry-Pi

## About RFID:

Radio-frequency identification (RFID) uses electromagnetic fields to automatically identify and track tags attached to objects. The tags contain electronically stored information. Passive tags collect energy from a nearby RFID reader's interrogating radio waves. Active tags have a local power source such as a battery and may operate at hundreds of meters from the RFID reader. Unlike a barcode, the tag need not be within the line of sight of the reader, so it may be embedded in the tracked object. RFID is one method for Automatic Identification and Data Capture (AIDC).



**Fig: RFID reader**



**Fig: Card with RFID ta**

## Pin Connection:

Connect the pins of the Sensor to raspberry-pi is simple as the module given to us has only 3 pins as follows:

Pin of Sensor → Pin of Raspberry-Pi

Vcc → Pin 2

GND → Pin 6

TX → Pin 10 (This is also GPIO 15, but used as UART- RX port to receive the serial data)

A universal asynchronous receiver/transmitter(UART, is a computer hardware device for asynchronous serial communication in which the data format and transmission speeds are configurable. The electric signaling levels and methods (such as differential signaling, etc.) are handled by a driver circuit external to the UART.

## Code:

```
import serial
#import serial module

def read_rfid ():
    ser = serial.Serial ("/dev/ttyS0")           #Open named port
    ser.baudrate=9600;
    data = ser.read(12)  #Read 12 characters from serial port to data
    ser.close ()
    return data                                #Return data

id = read_rfid ()                            #Function call
print (id);
```

## Alternative Code:

```
import time
import serial

ser = serial.Serial(port='/dev/ttyS0',baudrate =
9600,parity=serial.PARITY_NONE,stopbits=serial.STOPBITS_ONE,bytesize=serial.EIGHTBITS,time
out=1)
print("connected to: " + ser.portstr)
counter=0

while 1:
    if (ser.inWaiting()>0):
        x=ser.readline()
        print x
```

---

**Note:** This has been tried in **Raspberry Pi 3 Model B**, for the **Raspberry-Pi 2** the serial port is at **"/dev/ttyAMA0"**, so change the parameter appropriately as follows:  
**ser = serial.Serial ("/dev/ttyAMA0")**

**Also for Raspberry-Pi 3 there are work arounds (Not sure about Raspberry-Pi 2):**

**In /boot/cmdline.txt change the line to** `dwc_otg.lpm_enable=0 console=tty1 root=/dev/mmcblk0p7 rootfstype=ext4 elevator=deadline fsck.repair=yes rootwait`

**Then**

`sudo raspi-config`

Expand filesystem and enable serial on advanced page, **exit and reboot.**

If still doesn't work then In `/boot/config.txt`, add the line `core_freq=250`. (Sometimes, it may not work, in such case do consider rolling back from this step)

### Sample Output:

```
pi@raspberrypi: ~  
pi@raspberrypi:~ $ sudo python rfid2.py  
connected to: /dev/ttyS0  
0900970C8517  
0900970C8517  
0900970C8517  
0900970C8517  
0900970C8517  
0900970C8517
```

### References:

<https://frillip.com/raspberry-pi-3-uart-baud-rate-workaround/>

[https://en.wikipedia.org/wiki/Universal\\_asynchronous\\_receiver/transmitter](https://en.wikipedia.org/wiki/Universal_asynchronous_receiver/transmitter)

[https://en.wikipedia.org/wiki/Radio-frequency\\_identification](https://en.wikipedia.org/wiki/Radio-frequency_identification)

### Consider Reading:

<https://www.abelectronics.co.uk/kb/article/1035/raspberry-pi-3-serial-port-usage>

<http://www.instructables.com/id/A-Universal-RFID-Key/step2/Whats-stored-on-the-card/>