

MOUSE PLATFORM

NFC GUIDE

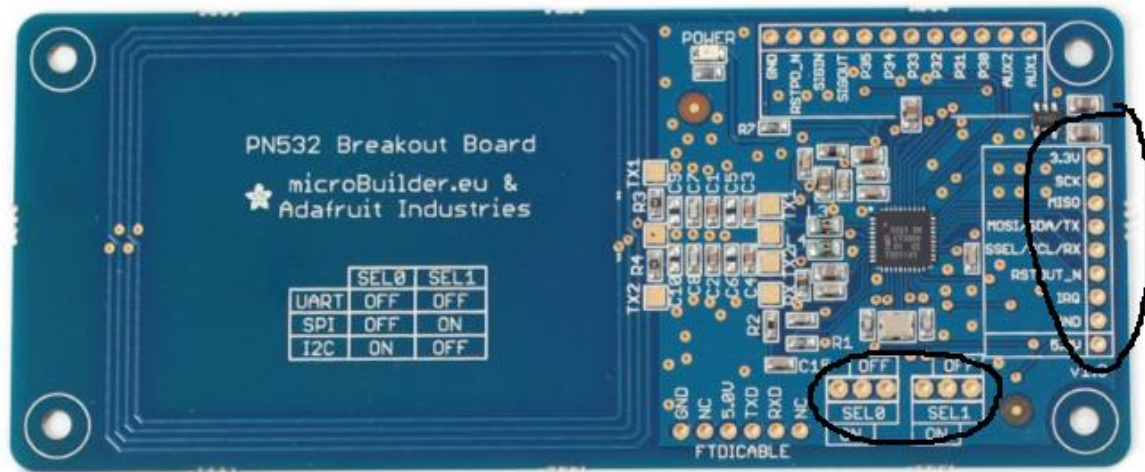
Noah Rozario

Wiring:

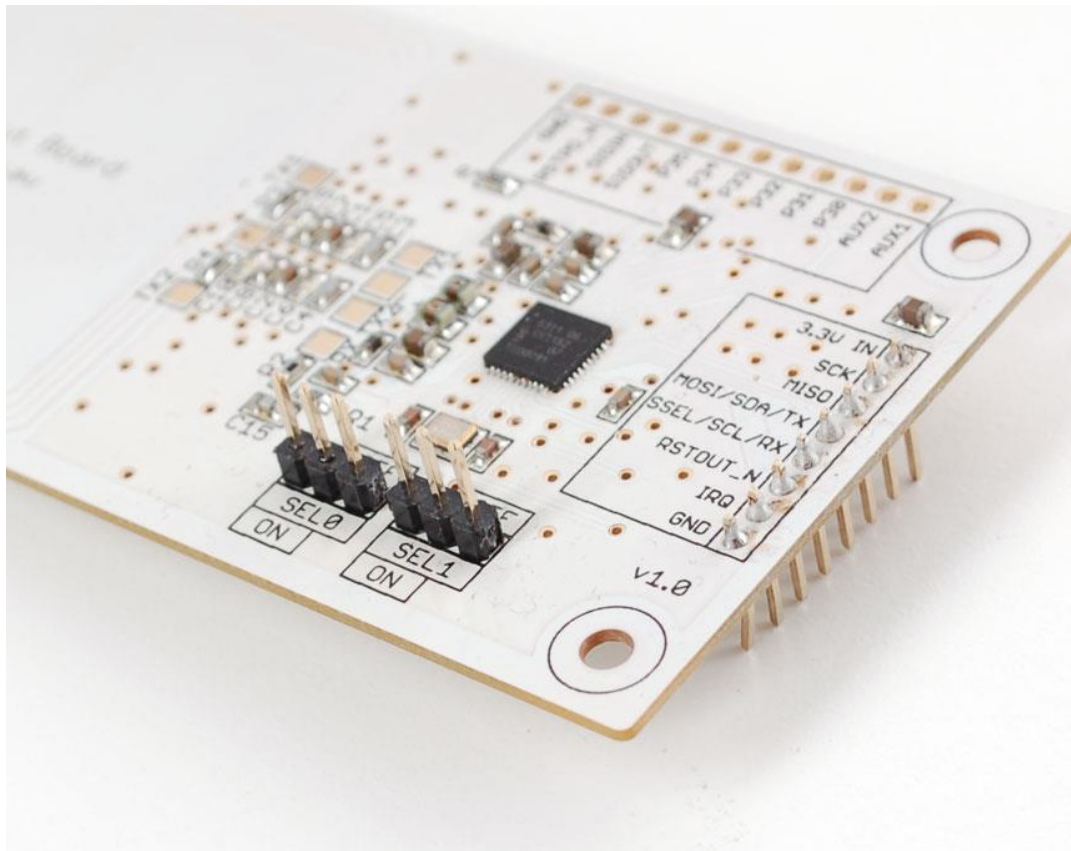
The official soldering and wiring guide is given in the Adafruit PN532 guide:

<https://learn.adafruit.com/adafruit-pn532-rfid-nfc/breakout-wiring>

Header pins must only be soldered onto the vias circled below:



The orientation of the pins is shown in the diagram below:



For a I²C connection, the jumpers provided must be placed in the ON position for SEL0 and OFF for SEL1.

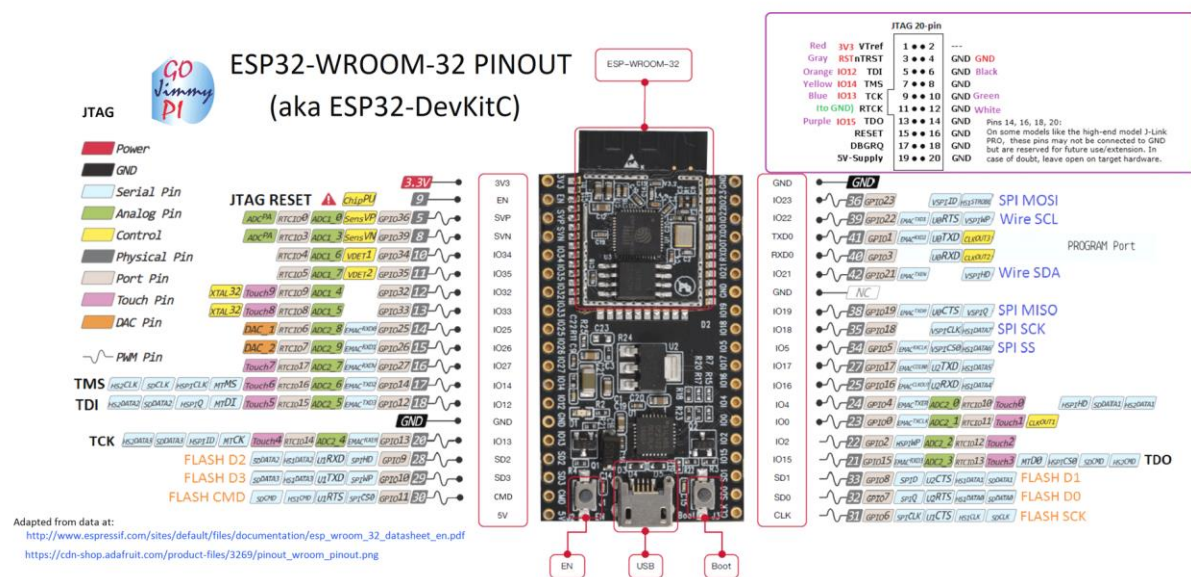
To connect to the ESP32, the following connections must be made:

ESP32	Adafruit PN532
3V3	3.3V
GPIO21	MOSI/SDA/TX
GPIO22	SSEL/SCL/RX
GPIO23	RSTOUT_N
GPIO19	IRQ
GND	GND

Note: RSTOUT_N and IRQ can be connected to any GPIO pin on the ESP32. Please make sure to change the pin values defined in the mouse_p2pInit and mouse_p2pTgt files. In my case, I used GPIO23 and 19 as shown below:

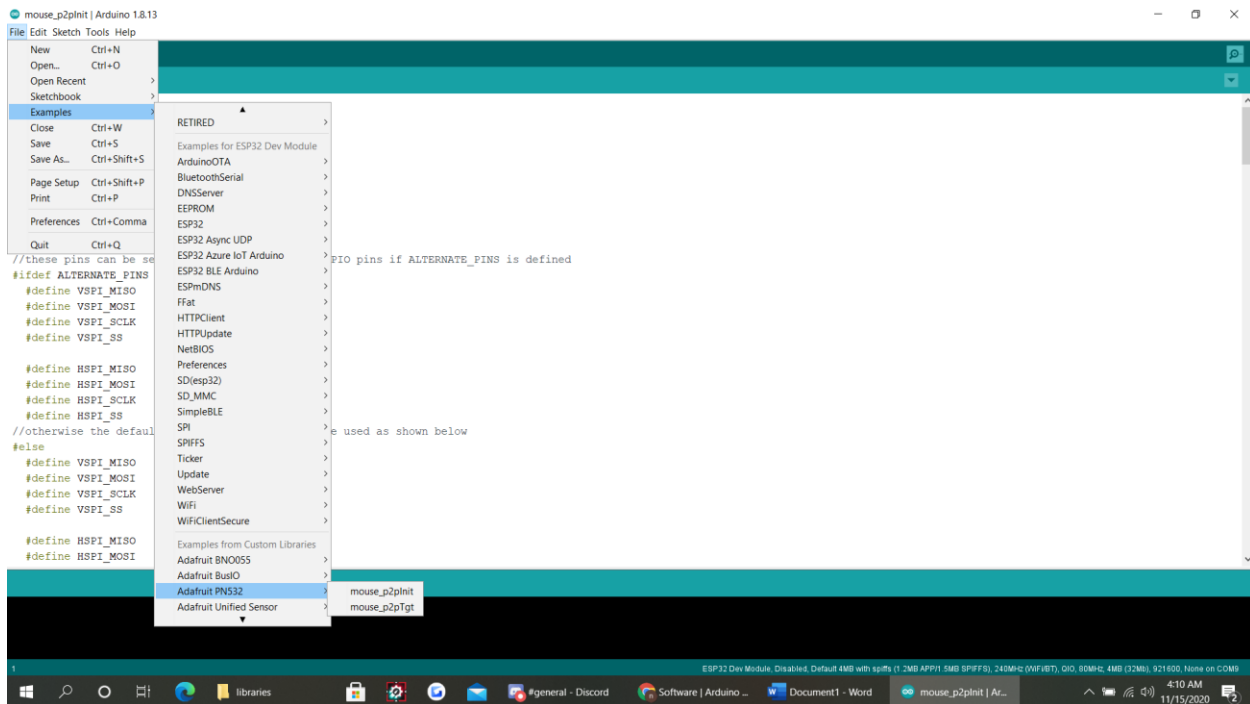
```
// If using the breakout or shield with I2C, define just the pins connected
// to the IRQ and reset lines. Can be any GPIO pin
#define PN532_IRQ    (19)
#define PN532_RESET (23)
```

For reference I have also included a pinout diagram of the ESP32 DevkitC:

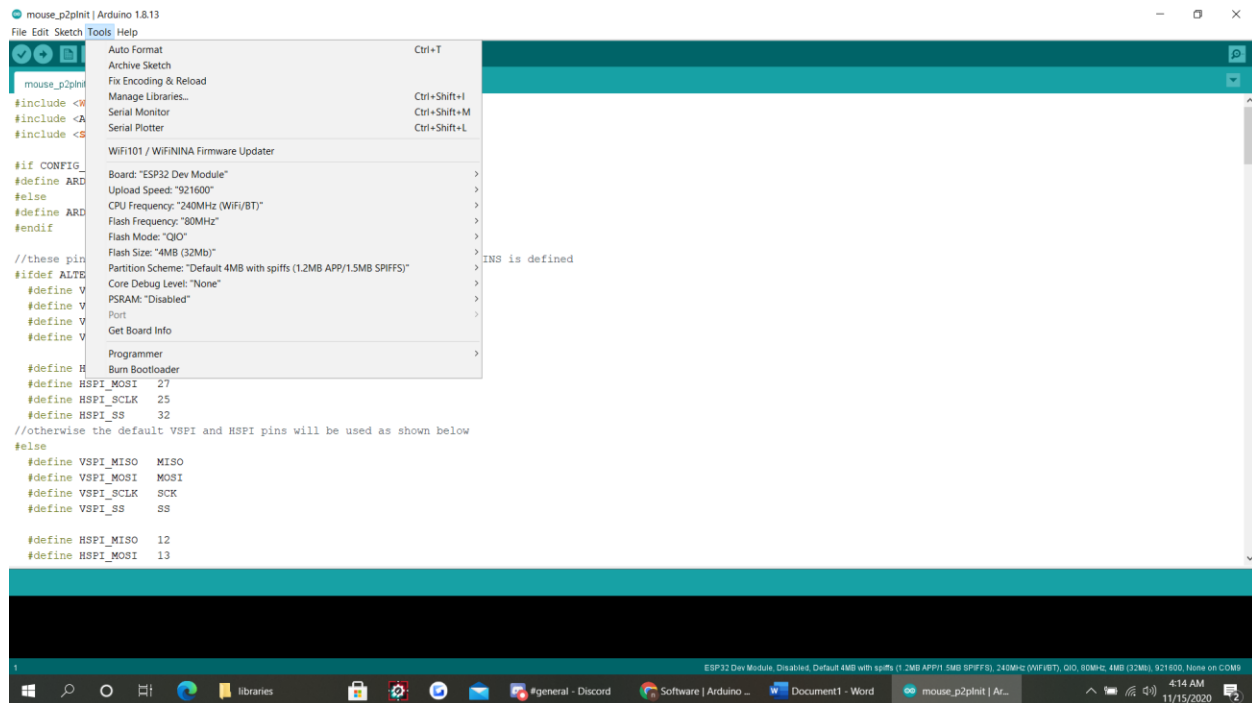


Library Installation Guide (Windows):

1. Install Arduino IDE using the following link: <https://www.arduino.cc/en/software>
2. Follow the link attached to setup the ESP32 for the Arduino IDE:
<https://randomnerdtutorials.com/installing-the-esp32-board-in-arduino-ide-windows-instructions/>
3. Go to the directory in which the Arduino IDE is installed and from there, go to libraries.
4. Paste the folder Adafruit_PN532 given by me into this directory.
5. Open the Arduino IDE.
6. Go to File->Examples->Adafruit PN532 as shown.



7. From here there are two examples, mouse_p2pInit which is the file utilizing FreeRTOS and would be uploaded onto the platform, and mouse_p2pTgt which would be ideally uploaded to the ESP32 on the MOUSE.
8. Go to Tools and make sure the following settings are in place.



The serial Port will be automatically set when you have connected the ESP32 to your PC via USB.

API References:

Important functions will be referenced here:

1. `bool Adafruit_PN532::P2PInitiator_Init()`

This function sets up the PN532 board as an initiator in a P2P connection with another board.

Parameters: None

Output:

- 1 - Initiated P2P connection successfully.
- 0 - Failed to initiate P2P connection.

2. `bool Adafruit_PN532::P2PTarget_Init()`

This function sets up the PN532 board as a target in a P2P connection with the initiator.

Parameters: None

Output:

- 1 – Connected to initiator successfully.
- 0 – Failed to connect to initiator.

3. `bool Adafruit_PN532::P2PInitiator_TxRx(uint8_t *t_buffer, uint8_t t_length, uint8_t *r_buffer, uint8_t *r_length)`

This function allows the initiator to send and receive data.

Parameters:

- `uint8_t *t_buffer`: A pointer to a buffer that contains the data to send to the target.
- `uint8_t t_length`: Size of the buffer in bytes. Can use `t_length = sizeof(t_buffer)`.
- `uint8_t *r_buffer`: A pointer to a buffer that will contain the data sent by the target.
- `uint8_t *r_length`: A pointer to the size of `r_buffer` in bytes.

Output:

- 1 – Data sent and received successfully
- 0 – Data sending and receipt failed.

4. `bool Adafruit_PN532::P2PTarget_TxRx(uint8_t *t_buffer, uint8_t t_length, uint8_t *r_buffer, uint8_t *r_length)`

This function allows the target to send and receive data sent by the initiator.

Parameters:

- `uint8_t *t_buffer`: A pointer to a buffer that contains the data to send to the initiator.
- `uint8_t t_length`: Size of the buffer in bytes. Can use `t_length = sizeof(t_buffer)`.
- `uint8_t *r_buffer`: A pointer to a buffer that will contain the data sent by the initiator.
- `uint8_t *r_length`: A pointer to the size of `r_buffer` in bytes.

Output:

- 1 – Data sent and received successfully
- 0 – Data sending and receipt failed.