

python code to communicate from circuitpython board to tablet

```
# SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
# SPDX-License-Identifier: MIT

# Basic example for using the BLE Connect UART
# To use, start this program, and start the Adafruit Bluefruit LE Connect app.
# Connect, and then select UART. Any text received FROM the connected device
# will be displayed. Periodically, text is sent TO the connected device.

# Programme de commande lecture et Ecriture par le Bluetooth
# Modifier le 8/03/2023 par P Mathieu

import board
import busio
import digitalio
import time
from adafruit_ble import BLERadio
from adafruit_ble.advertising.standard import ProvideServicesAdvertisement
from adafruit_ble.services.nordic import UARTService
from adafruit_esp32spi import adafruit_esp32spi
from adafruit_airlift.esp32 import ESP32

led = digitalio.DigitalInOut(board.D2)
led.direction = digitalio.Direction.OUTPUT

from adafruit_airlift.esp32 import ESP32
esp32 = ESP32() # DEFAULT

adapter = esp32.start_bluetooth()

ble = BLERadio(adapter)
uart = UARTService()
advertisement = ProvideServicesAdvertisement(uart)

def commande_moteur():
    pass

def commande_servo():
    pass

def lecture_capteurs():
    return "45,45.3,70" # valeur provisoir pour test

while True:
    ble.start_advertising(advertisement)
    print("Attente de connexion")
    while not ble.connected:
        pass
    print("La Connexion est active")
```

```

while ble.connected:
    # Returns b" if nothing was read.
    donnees = lecture_capteurs()
    print("lecture des capteurs : ",donnees)
    uart.write(donnees)
    time.sleep(3)
    uart.write('23,10,99')
    time.sleep(3)

```

python code to communicate from tablet to circuitpython board

```

# SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
# SPDX-License-Identifier: MIT

```

```

# Basic example for using the BLE Connect UART
# To use, start this program, and start the Adafruit Bluefruit LE Connect app.
# Connect, and then select UART. Any text received FROM the connected device
# will be displayed. Periodically, text is sent TO the connected device.

```

```

import board
import digitalio
from adafruit_ble import BLERadio
from adafruit_ble.advertising.standard import ProvideServicesAdvertisement
from adafruit_ble.services.nordic import UARTService

```

```

led = digitalio.DigitalInOut(board.D6)
led.direction = digitalio.Direction.OUTPUT

```

```

from adafruit_airlift.esp32 import ESP32
esp32 = ESP32() # DEFAULT

```

```

adapter = esp32.start_bluetooth()

```

```

ble = BLERadio(adapter)
uart = UARTService()
advertisement = ProvideServicesAdvertisement(uart)

```

```

while True:
    ble.start_advertising(advertisement)
    print("Attente de connexion")
    while not ble.connected:
        pass
    print("La Connexion est active")
    while ble.connected:
        # Returns b" if nothing was read.
        one_byte = uart.read(1)
        if one_byte==b'\x02':
            led.value =True
        elif one_byte==b'\x01':
            led.value =False

```