

# Micro:bit measures temperature which is sent to MatDeck and displayed in Virtument

This example illustrates communications between MatDeck and micro:bit using a com port. The obtained results are displayed in Virtument.

The document here contains the micro:bit Python program. The user can flash .HEX files onto a micro:bit directly from the document. They will need to highlight the whole Python block they would like to flash and click Deploy. If the micro:bit Python block has already been deployed to the micro:bit, you will not need to deploy it again to run it. The micro:bit should be connected to the PC. The receiver code is also in this MatDeck document

The micro:bit's processor contains a temperature sensor which can be used in your programs. It's a useful approximation of the temperature around the micro:bit.

- Flush the following code to your micro:bit (select all lines and click Deploy button from programming tab)

```
1 #py
2 from microbit import *
3
4 while True:
5     x = temperature()
6     print(x)
7     display.show(Image.YES)
8     sleep(250)
9     display.show(Image.NO)
10    sleep(250)
11
12 ###
```

The temperature read from the micro:bit unit is displayed using virtual instruments in Virtument, by the Temperature.vr file related to this document. The Virtument.vr file needs to be opened, go to ribbon tab **Toolbox** and select **Virtument**. Once Virtument is open, use the **Load** button to open the file.



After opening Virtument, evaluate the rest of the code below.

MatDeck can communicate and receive data from the micro:bit unit via com port. The micro Python code given above will cause the micro:bit to send temperature data via a com port. The data can then be displayed using Virtument.

The required parameters for com port communications are:

- COM3
- Baud rate = 115200
- Data = 8 bits
- Parity = none
- Stop = 1 bit

```
13 handle := com_open("COM3,115200,N,8,1")
14 channel := channel_create("micro1", "w")
15 t := timer_create(250)
16 Temp := 0
17 counter := 10
18 on_event(t,microbit_read())
```

Here, we send the temperature received from micro:bit to PC com port then to Virtument.

```
19 microbit_read()
20 {
21     value := com_read(handle, 100)
22
23     if(size(value) == 4)
24     {
25         valuestr := vec2str(value)
26         Temp = to_number(mid(valuestr, 0,2))
27         channel_write(channel, Temp)
28     }
29     counter -= 1
30     if(counter == 0)
31     {
32         com_close(handle)
33         timer_delete(t)
34     }
35 }
```