# XBee Radio Modules


# Coding XBee Module Applications Using MicroPython

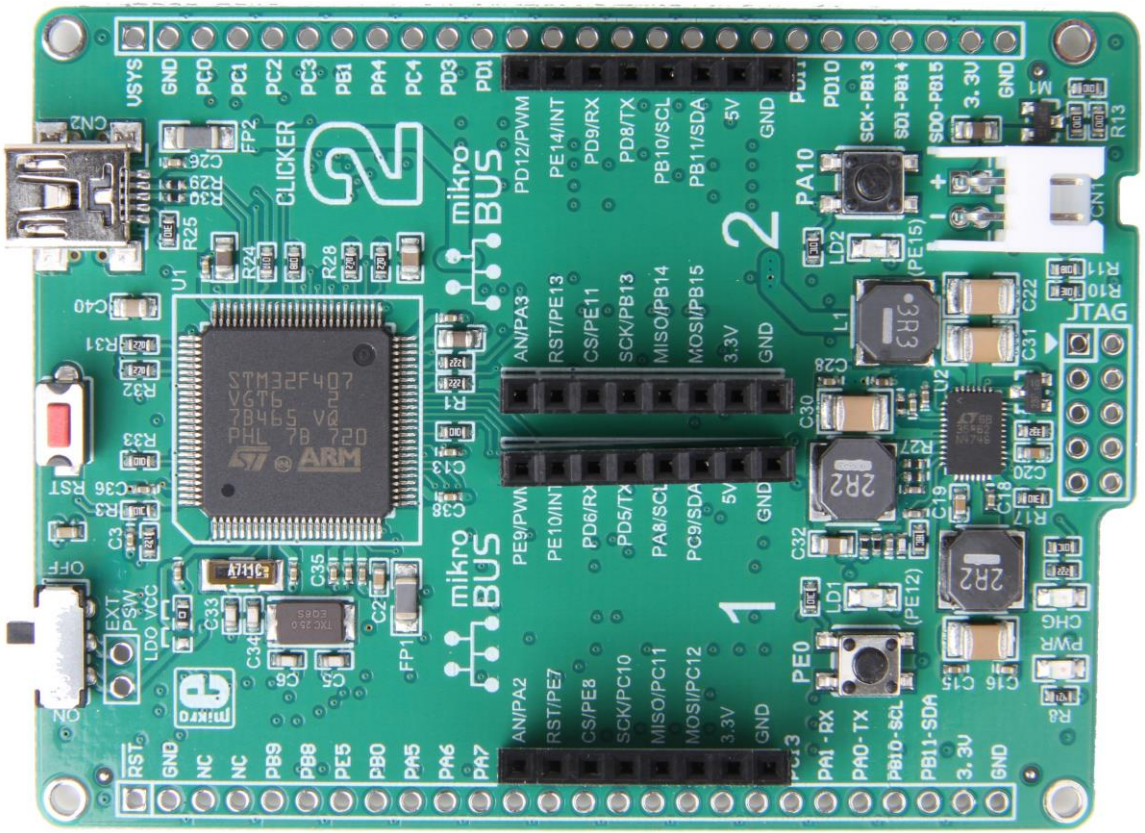
January 28, 2020

**Fred Eady**

Presented by:

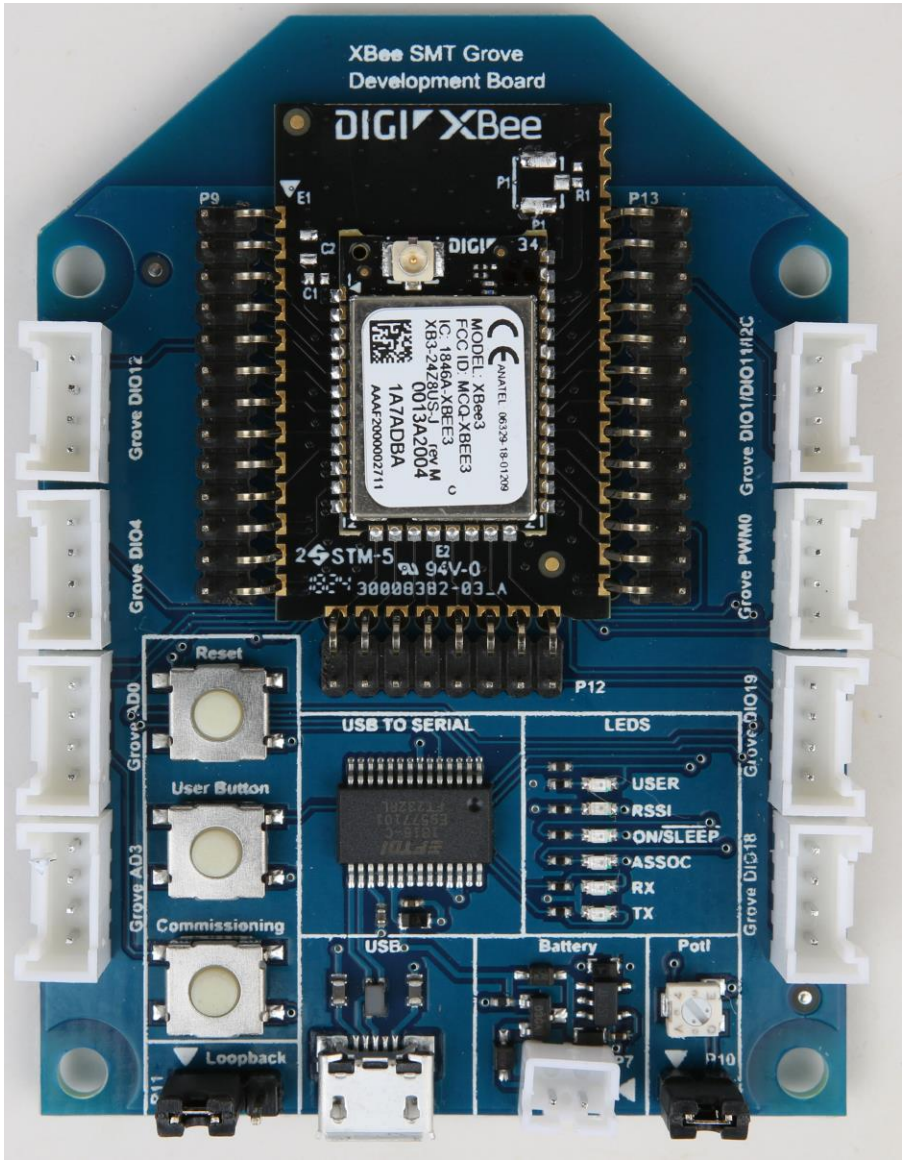
# AGENDA

- PyCharm XBee 3 Application
- PyCharm XBee/ARM/MicroPython Application
- Day 2 Summary




# XBee Radio Modules

## PyCharm XBee 3 Application

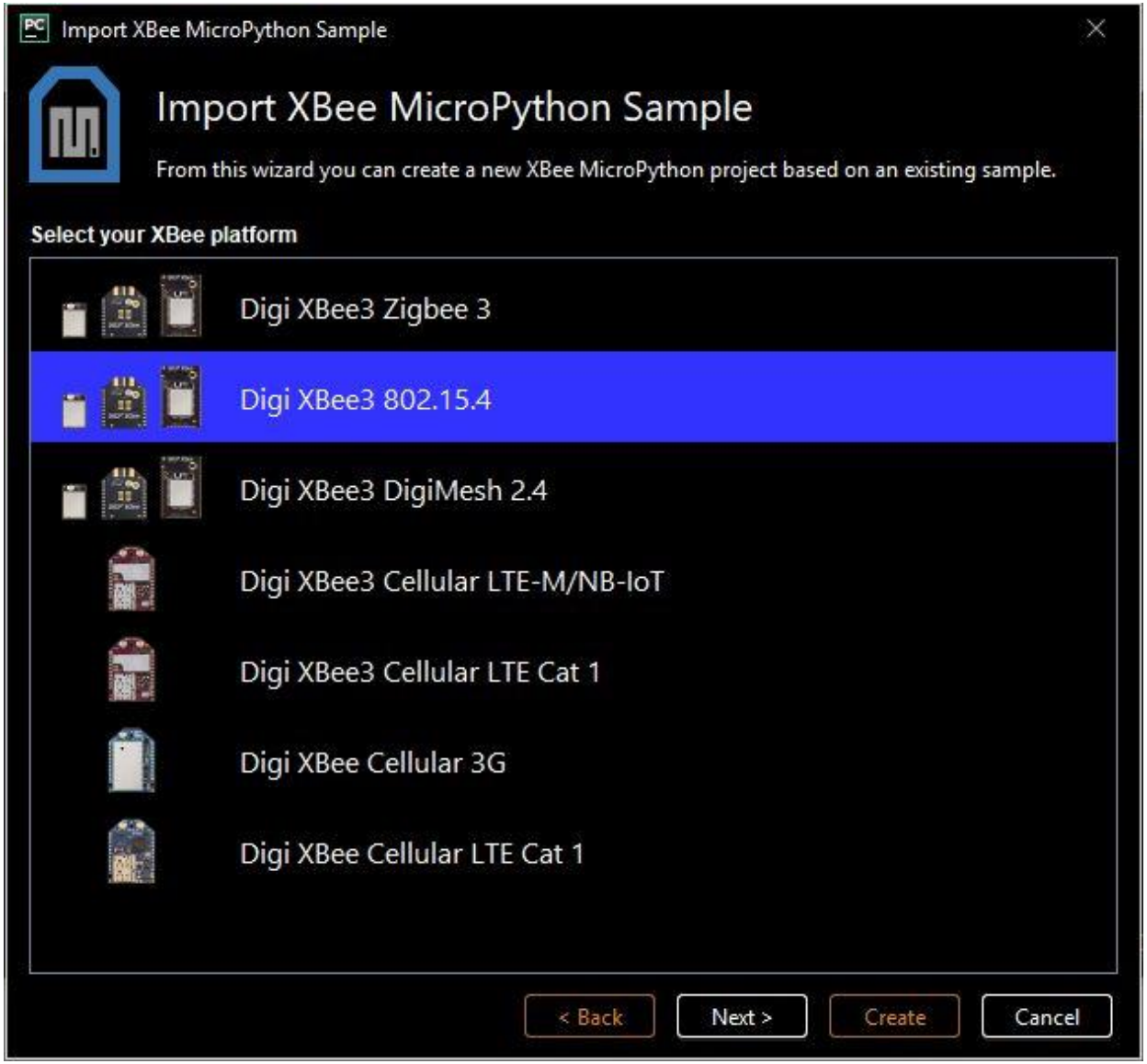


# XBee Radio Modules

## PyCharm XBee 3 Application

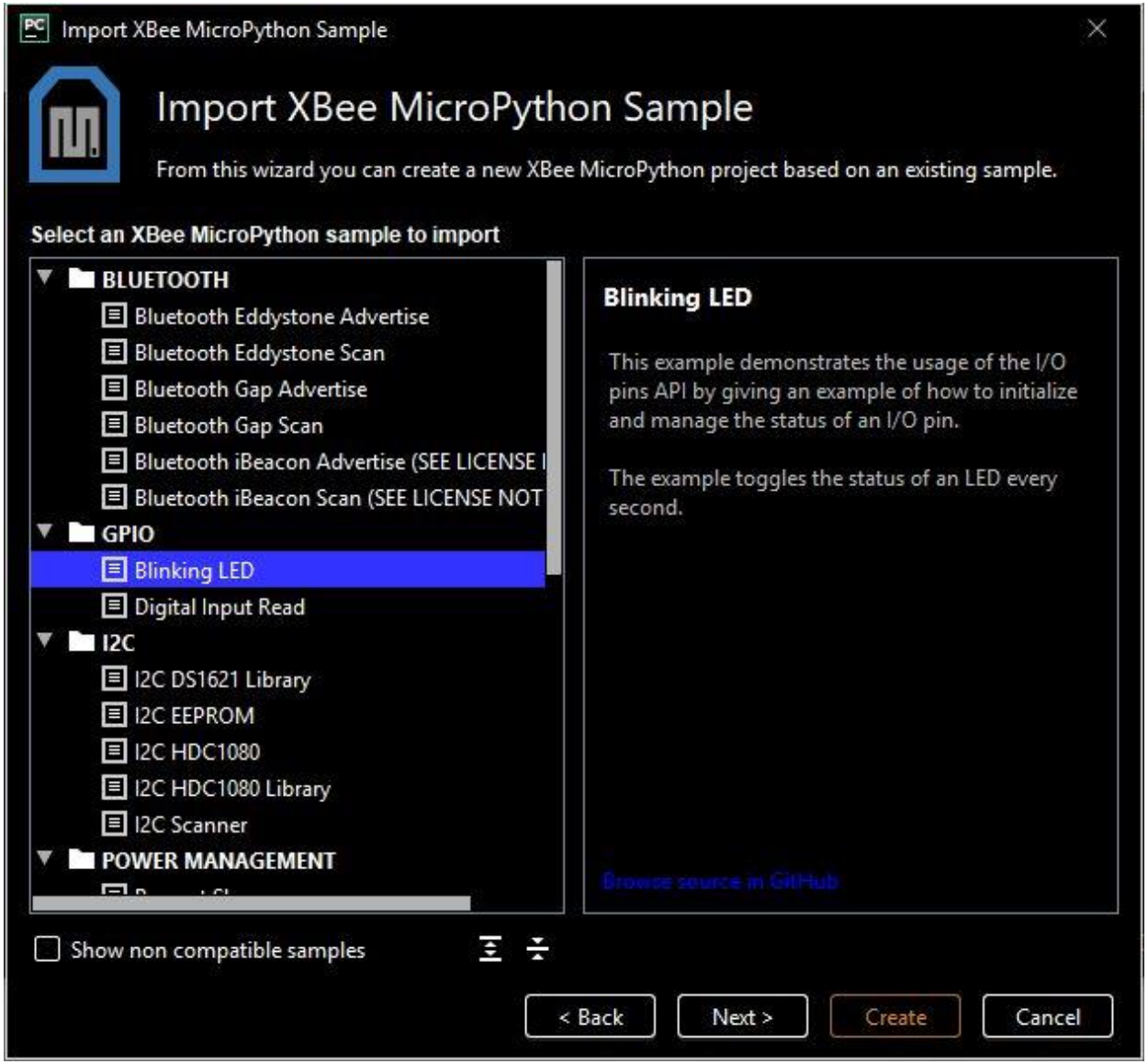
# XBee Radio Modules

## PyCharm XBee 3 Application



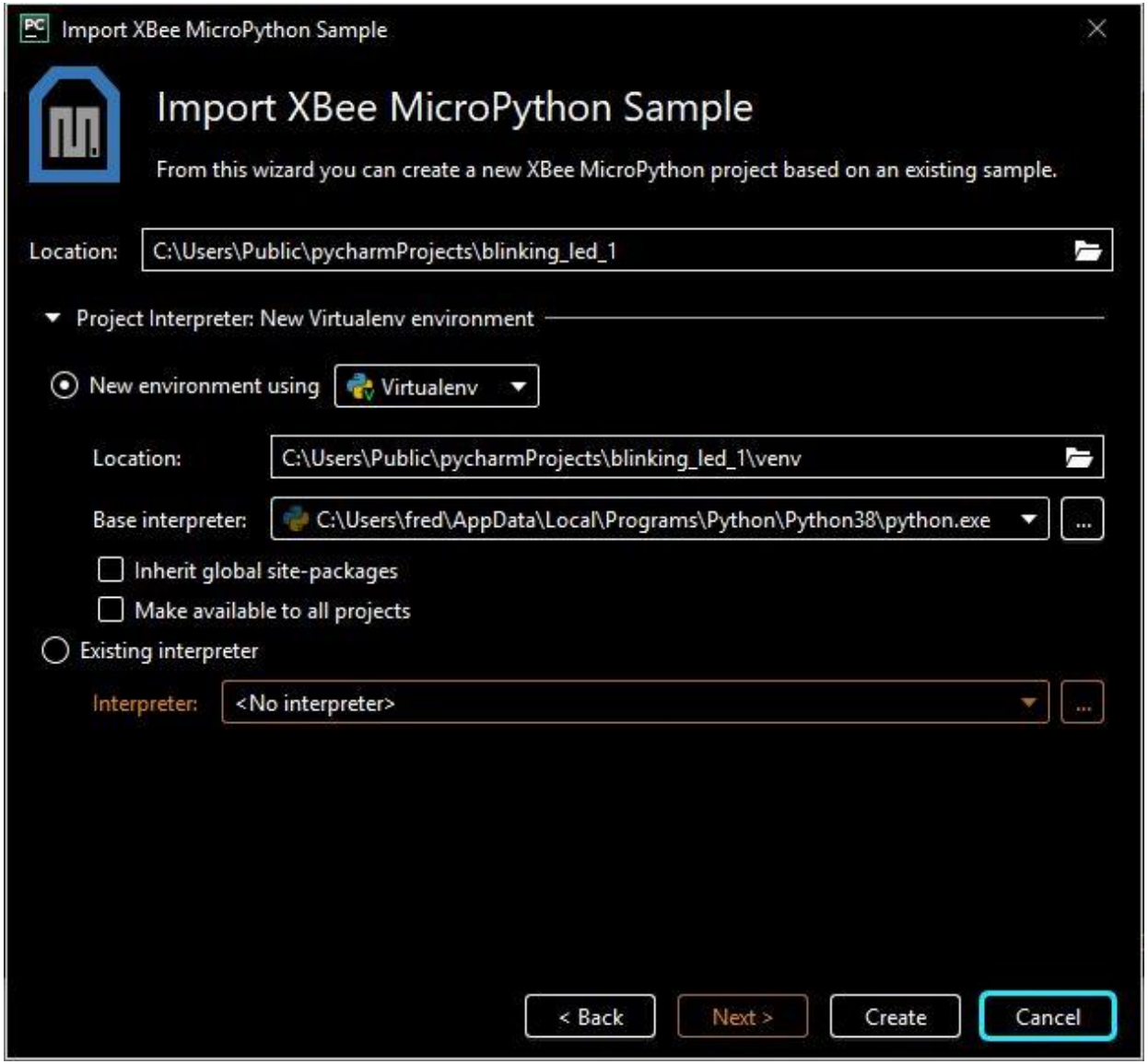
# XBee Radio Modules

## PyCharm XBee 3 Application



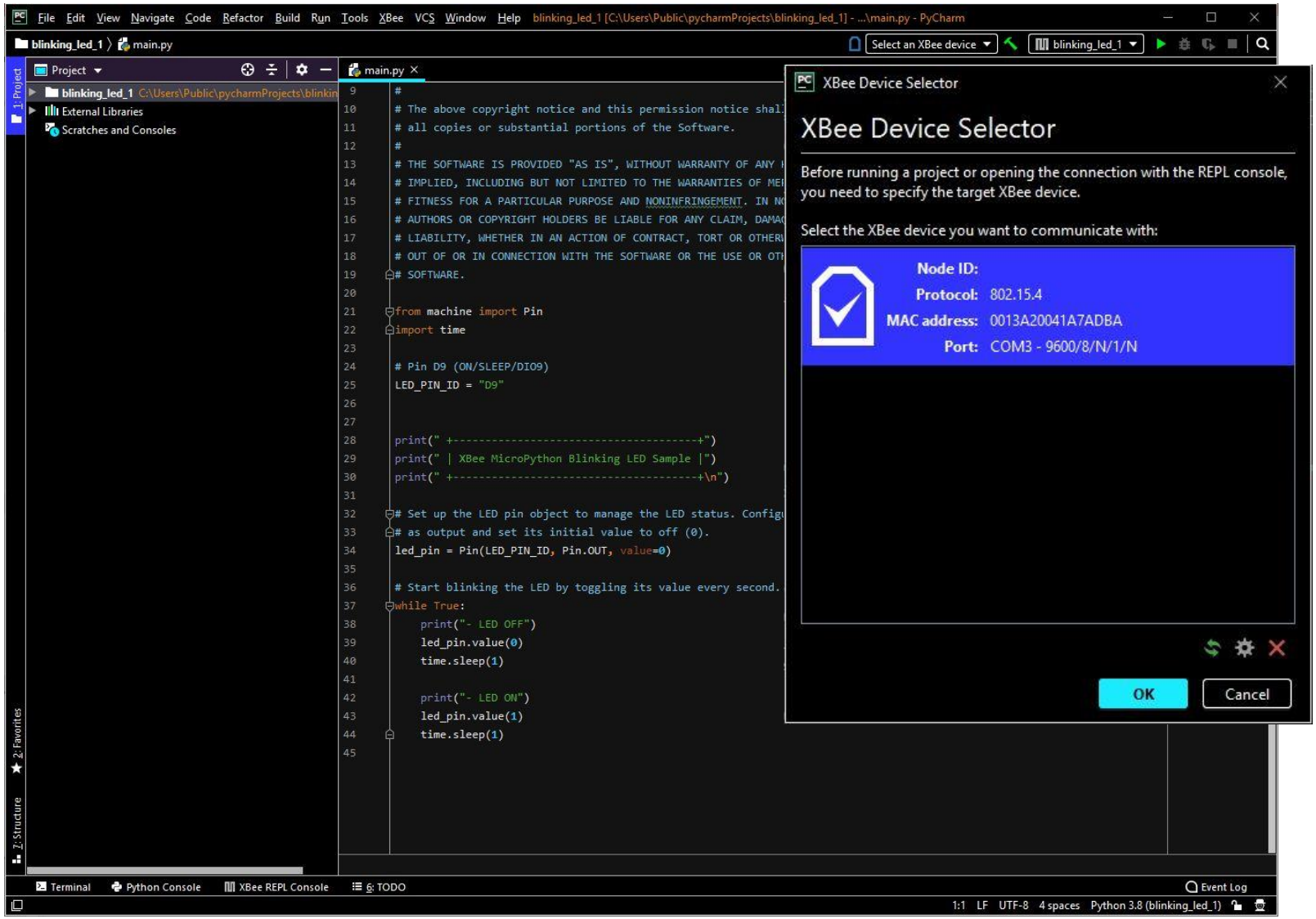
# XBee Radio Modules

## PyCharm XBee 3 Application



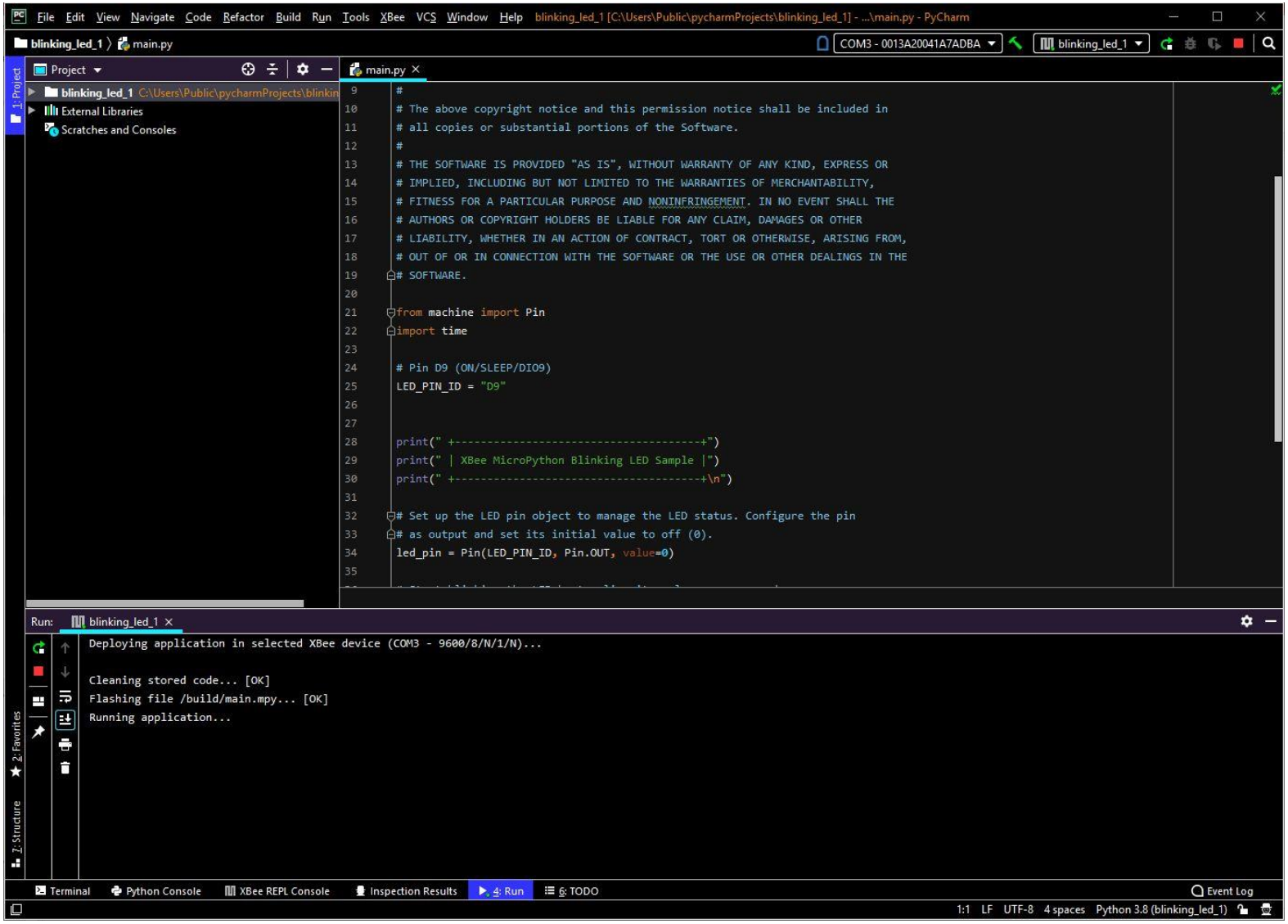
# XBee Radio Modules

## PyCharm XBee 3 Application



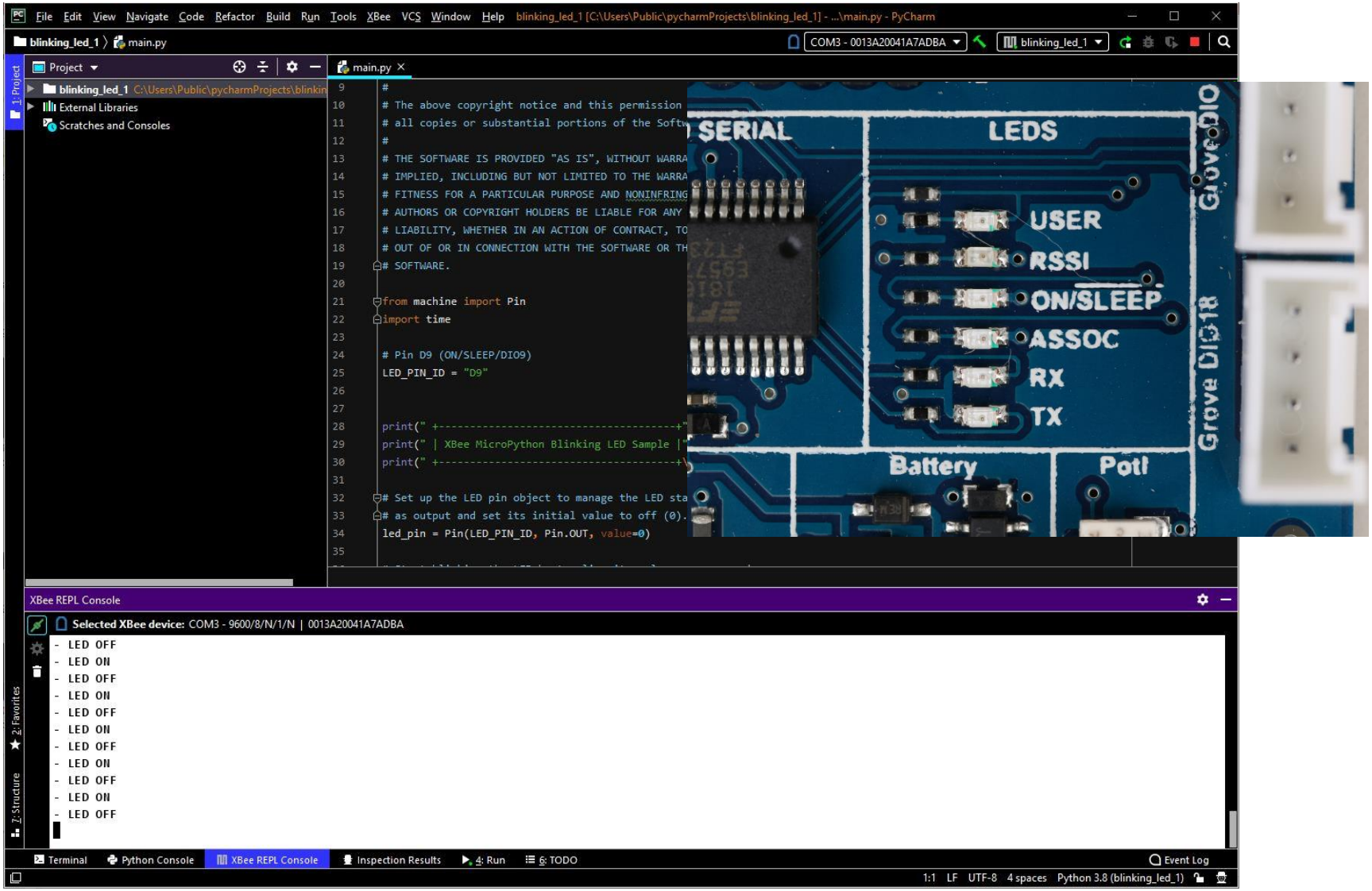
# XBee Radio Modules

## PyCharm XBee 3 Application



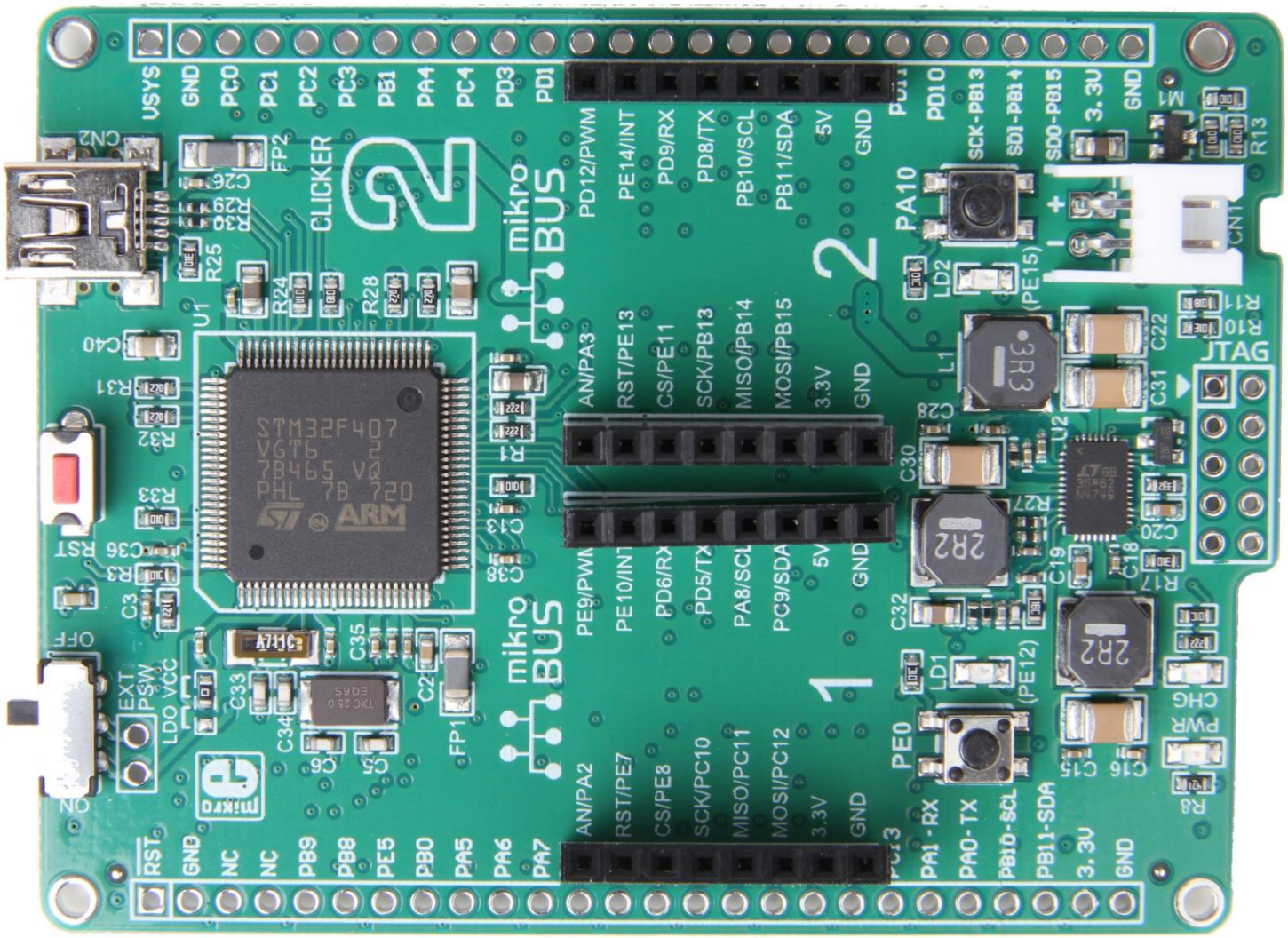
# XBee Radio Modules

## PyCharm XBee 3 Application



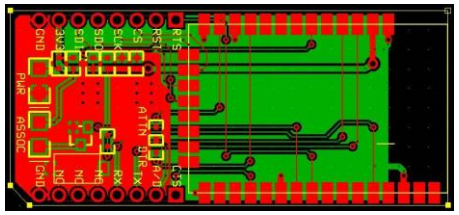
# XBee Radio Modules

## PyCharm XBee/ARM/MicroPython Application



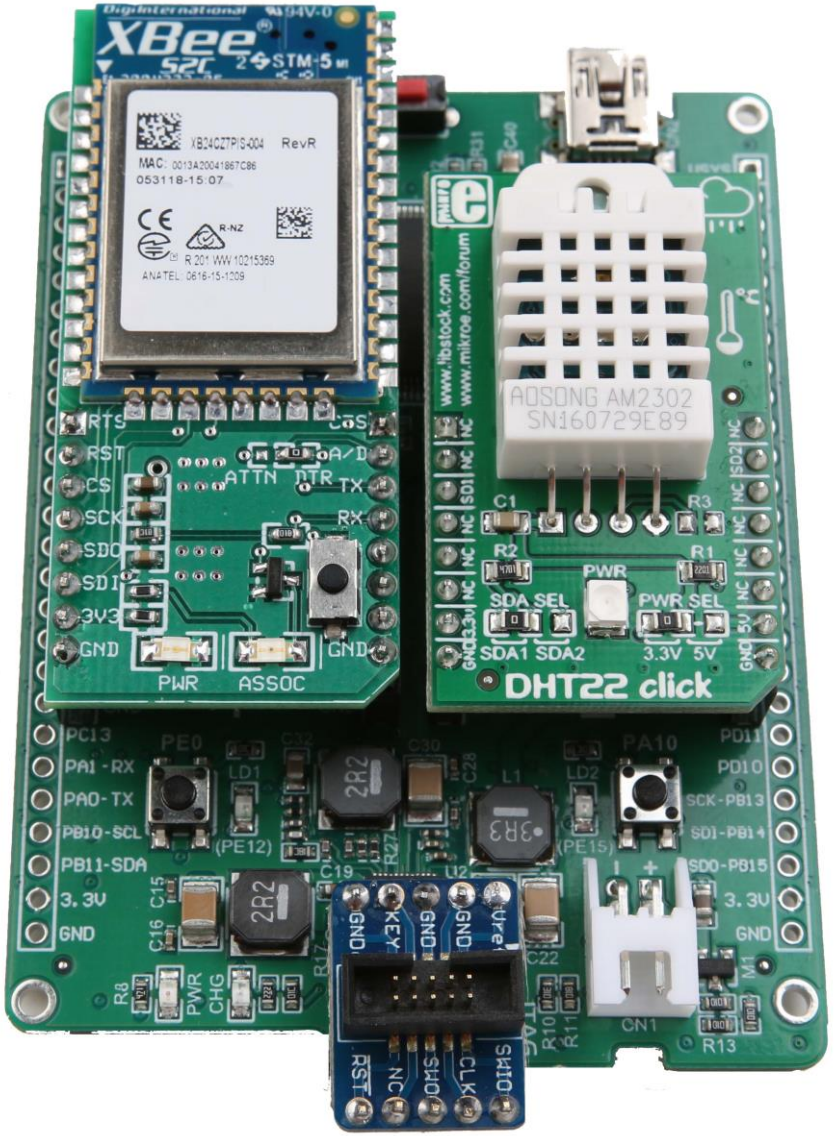
# XBee Radio Modules

## PyCharm XBee/ARM/MicroPython Application



# XBee Radio Modules

## PyCharm XBee/ARM/MicroPython Application




Presented by:

# XBee Radio Modules

## PyCharm XBee/ARM/MicroPython Application

Open — mpconfigboard.h — ~/micropython/ports/stm32/boards/STM32F407 — Save

```
#define MICROPY_HW_BOARD_NAME       "CLICK2"
#define MICROPY_HW_MCU_NAME         "STM32F407"

#define MICROPY_HW_HAS_SWITCH       (0)
#define MICROPY_HW_HAS_FLASH        (1)
#define MICROPY_HW_ENABLE_RNG       (1)
#define MICROPY_HW_ENABLE_RTC       (1)
#define MICROPY_HW_ENABLE_DAC       (1)
#define MICROPY_HW_ENABLE_USB       (1)

// HSE is 25MHz
#define MICROPY_HW_CLK_PLLM (25)
#define MICROPY_HW_CLK_PLLN (336)
#define MICROPY_HW_CLK_PLLP (RCC_PLLP_DIV2)
#define MICROPY_HW_CLK_PLLQ (7)

// UART config
#if 0
// A9 is used for USB VBUS detect, and A10 is used for USB_FS_ID.
// UART1 is also on PB6/7 but PB6 is tied to the Audio SCL line.
// Without board modifications, this makes UART1 unusable on this board.
#define MICROPY_HW_UART1_TX     (pin_A9)
#define MICROPY_HW_UART1_RX     (pin_A10)
#endif
#define MICROPY_HW_UART2_TX     (pin_D5)
#define MICROPY_HW_UART2_RX     (pin_D6)
//#define MICROPY_HW_UART2_RTS    (pin_A1)
//#define MICROPY_HW_UART2_CTS    (pin_A0)
#define MICROPY_HW_UART3_TX     (pin_D8)
#define MICROPY_HW_UART3_RX     (pin_D9)
//#define MICROPY_HW_UART3_RTS    (pin_D12)
//#define MICROPY_HW_UART3_CTS    (pin_D11)
#if MICROPY_HW_HAS_SWITCH == 0
// NOTE: A0 also connects to the user switch. To use UART4 you should
//       set MICROPY_HW_HAS_SWITCH to 0, and also remove SB20 (on the back
//       of the board near the USER switch).
#define MICROPY_HW_UART4_TX     (pin_A0)
```

C/ObjC Header — Tab Width: 8 — Ln 61, Col 1 — INS

# XBee Radio Modules

## PyCharm XBee/ARM/MicroPython Application

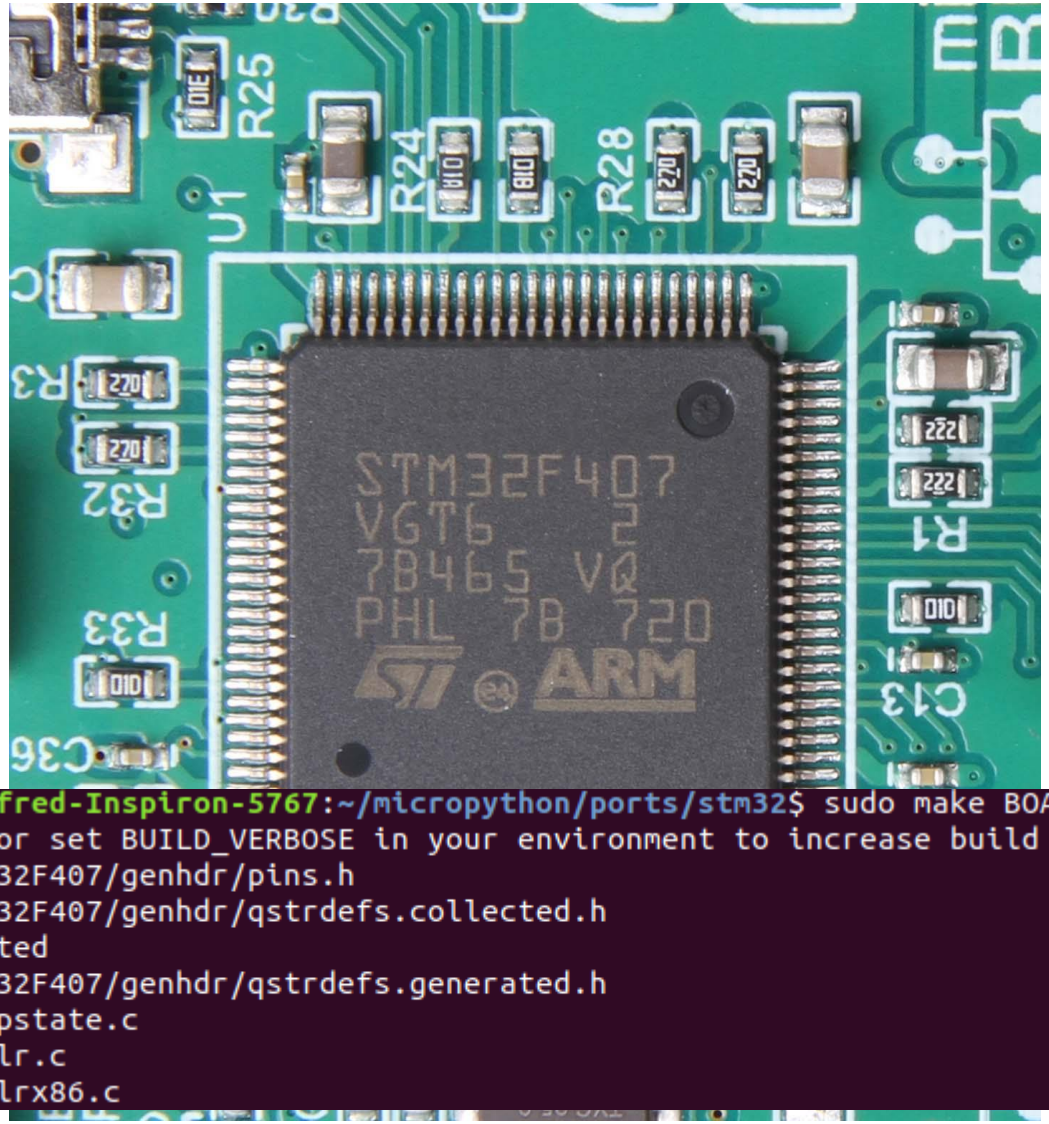
Open | stm32f4xx_hal_conf.h ~/micropython/ports/stm32/boards/STM32F407 | Save

```
/* ########################## HSE/HSI Values adaptation ##################### */
/**
  * @brief Adjust the value of External High Speed oscillator (HSE) used in your application.
  *        This value is used by the RCC HAL module to compute the system frequency
  *        (when HSE is used as system clock source, directly or through the PLL).
  */
#if !defined  (HSE_VALUE)
  #define HSE_VALUE    ((uint32_t)25000000) /*!< Value of the External oscillator in Hz */
#endif /* HSE_VALUE */

#if !defined  (HSE_STARTUP_TIMEOUT)
  #define HSE_STARTUP_TIMEOUT    ((uint32_t)100U)   /*!< Time out for HSE start up, in ms */
#endif /* HSE_STARTUP_TIMEOUT */

/**
  * @brief Internal High Speed oscillator (HSI) value.
  *        This value is used by the RCC HAL module to compute the system frequency
  *        (when HSI is used as system clock source, directly or through the PLL).
  */
#if !defined  (HSI_VALUE)
  #define HSI_VALUE    ((uint32_t)16000000) /*!< Value of the Internal oscillator in Hz*/
#endif /* HSI_VALUE */

/**
  * @brief Internal Low Speed oscillator (LSI) value.
  */
#if !defined  (LSI_VALUE)
 #define LSI_VALUE  ((uint32_t)40000)
#endif /* LSI_VALUE */                    /*!< Value of the Internal Low Speed oscillator in
Hz

                                           The real value may vary depending on the variations
                                           in voltage and temperature.  */
/**
  * @brief External Low Speed oscillator (LSE) value.
  */
#if !defined  (LSE_VALUE)
```

C/ObjC Header | Tab Width: 8 | Ln 100, Col 37 | INS

# XBee Radio Modules

## PyCharm XBee/ARM/MicroPython Application

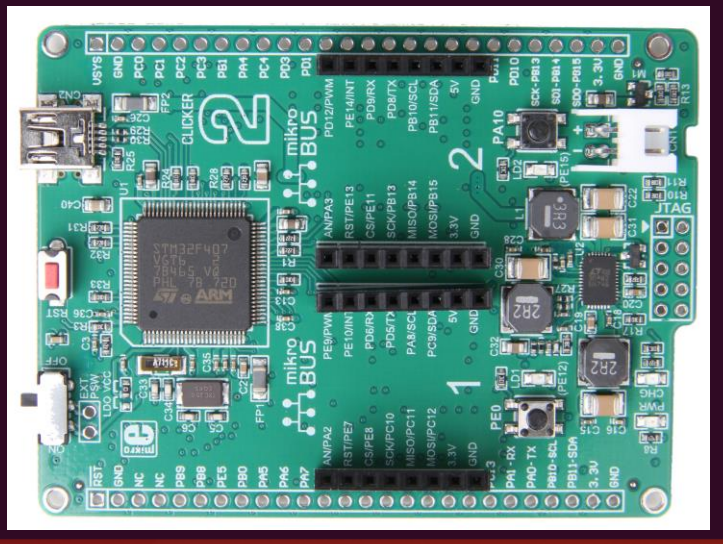



```
userfred@userfred-Inspiron-5767:~/micropython/ports/stm32$ sudo make BOARD=STM32F407
Use make V=1 or set BUILD_VERBOSE in your environment to increase build verbosity.
GEN build-STM32F407/genhdr/pins.h
GEN build-STM32F407/genhdr/qstrdefs.collected.h
QSTR not updated
GEN build-STM32F407/genhdr/qstrdefs.generated.h
CC ../../py/mpstate.c
CC ../../py/nlr.c
CC ../../py/nlrx86.c
```

# XBee Radio Modules

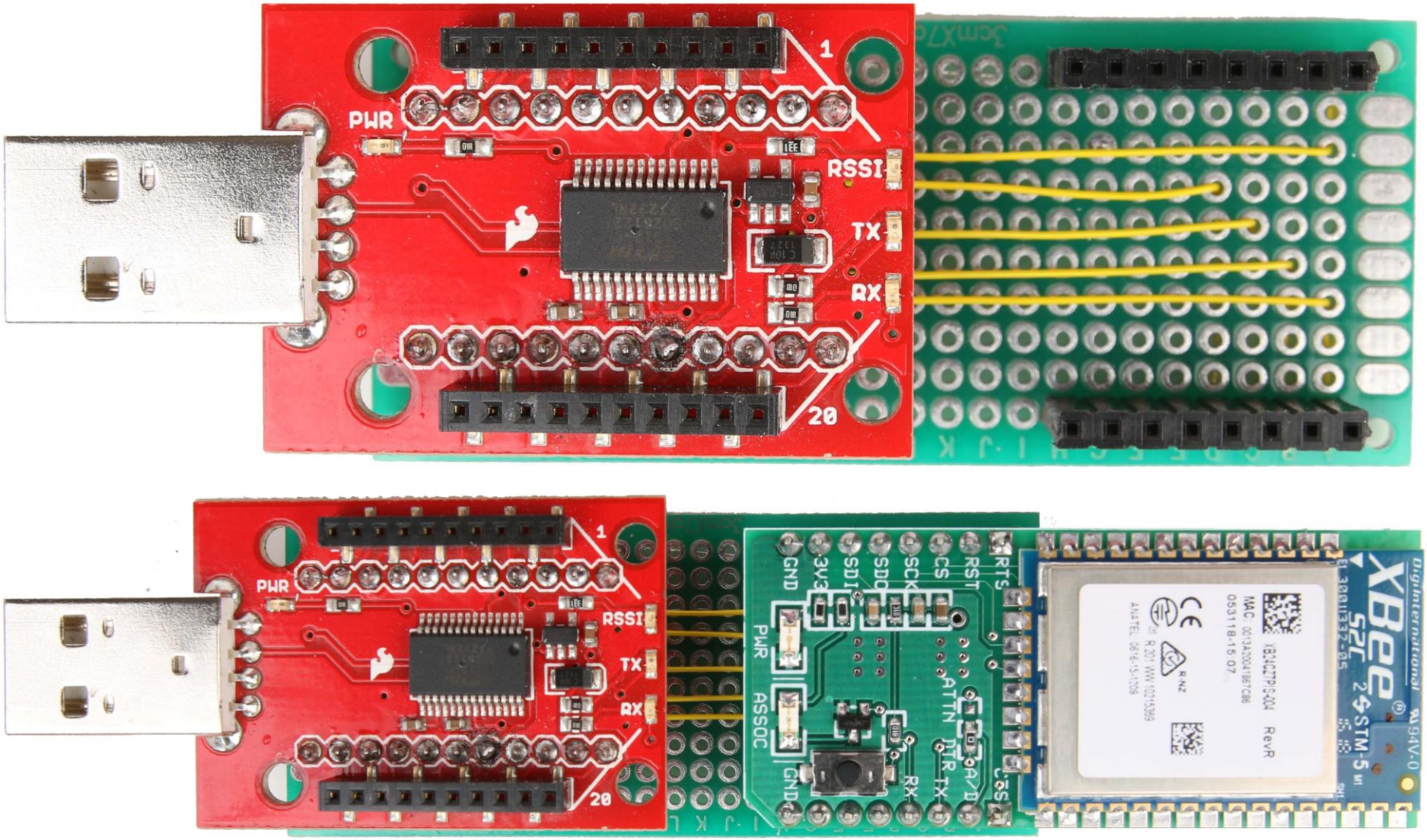
## PyCharm XBee/ARM/MicroPython Application

```
userfred@userfred-Inspiron-5767:~/micropython/ports/stm32$ cd build-STM32F407
userfred@userfred-Inspiron-5767:~/micropython/ports/stm32/build-STM32F407$ st-flash --format ihex write firmware.hex
st-flash 1.5.0
2020-01-05T17:14:18 INFO common.c: Loading device parameters....
2020-01-05T17:14:18 INFO common.c: Device connected is: F4 device, id 0x10076413
2020-01-05T17:14:18 INFO common.c: SRAM size: 0x30000 bytes (192 KiB), Flash: 0x100000 bytes (1024 KiB) in pages of 16384 bytes
2020-01-05T17:14:18 INFO common.c: Attempting to write 431628 (0x6960c) bytes to stm32 address: 134217728 (0x8000000)
Flash page at addr: 0x08060000 erasedEraseFlash - Sector:0x7 Size:0x20000
2020-01-05T17:14:28 INFO common.c: Finished erasing 8 pages of 131072 (0x20000) bytes
2020-01-05T17:14:28 INFO common.c: Starting Flash write for F2/F4/L4
2020-01-05T17:14:28 INFO flash_loader.c: Successfully loaded flash loader in sram
enabling 32-bit flash writes
size: 32768
size: 32768
size: 32768
size: 32768
size: 32768
size: 32768
size: 32768
size: 32768
size: 32768
size: 32768
size: 32768
size: 32768
size: 32768
size: 5644
2020-01-05T17:14:34 INFO common.c: Starting verification of write complete
2020-01-05T17:14:38 INFO common.c: Flash written and verified! jolly good!
userfred@userfred-Inspiron-5767:~/micropython/ports/stm32/build-STM32F407$
```



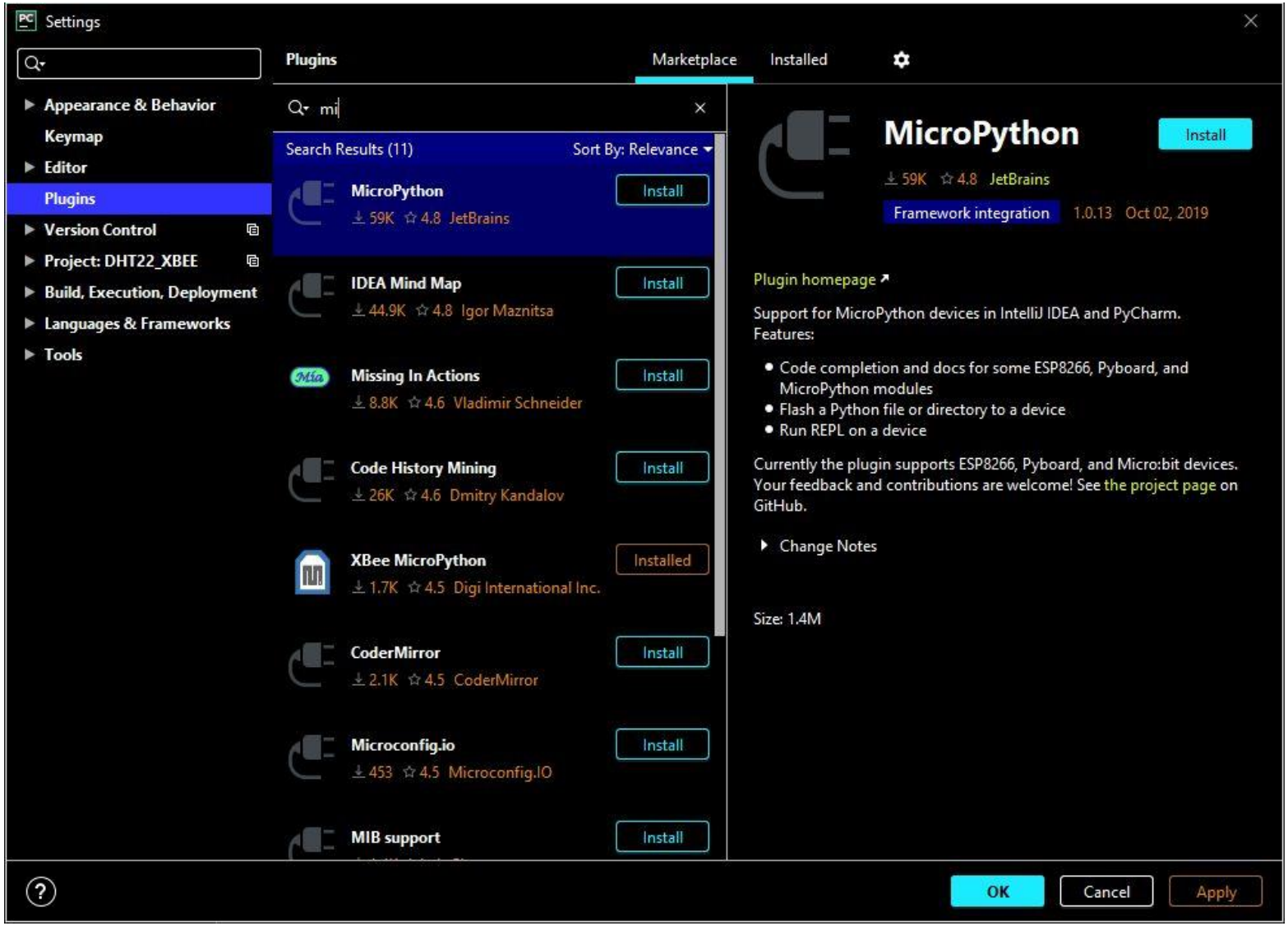
# XBee Radio Modules

## PyCharm XBee/ARM/MicroPython Application



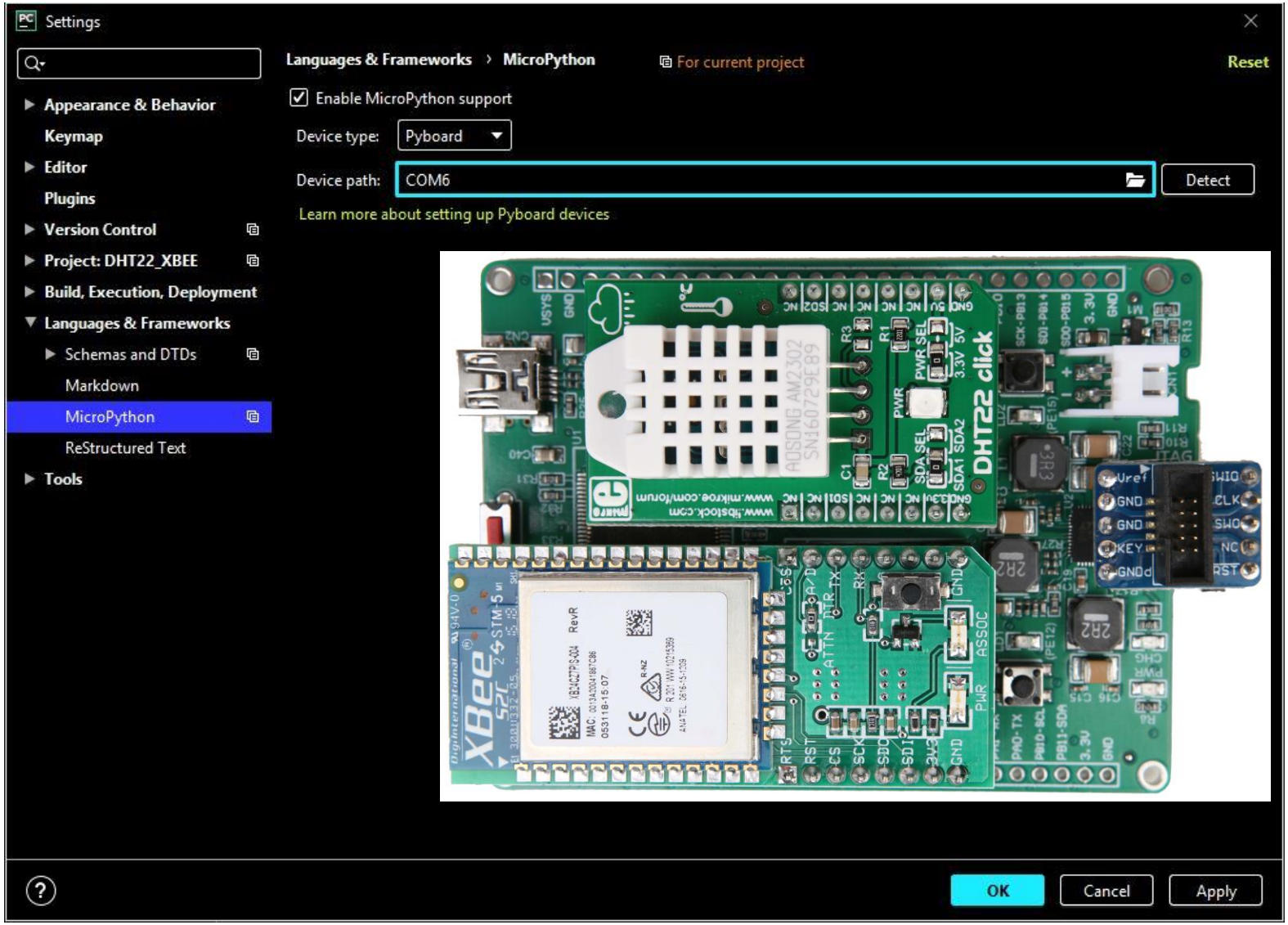
# XBee Radio Modules

## PyCharm XBee/ARM/MicroPython Application



# XBee Radio Modules

## PyCharm XBee/ARM/MicroPython Application



# XBee Radio Modules

## PyCharm XBee/ARM/MicroPython Application

Run/Debug Configurations

MicroPython

Flash main.py

Templates

Name: Flash main.py

Share through VCS

Allow parallel run

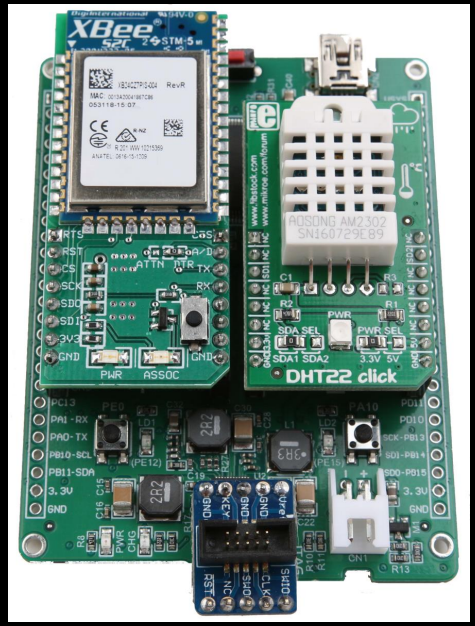
Path: C:\Users\Public\pycharmProjects\DHT22_XBEE\main.py

Before launch: Activate tool window

There are no tasks to run before launch

Show this page

Activate tool window



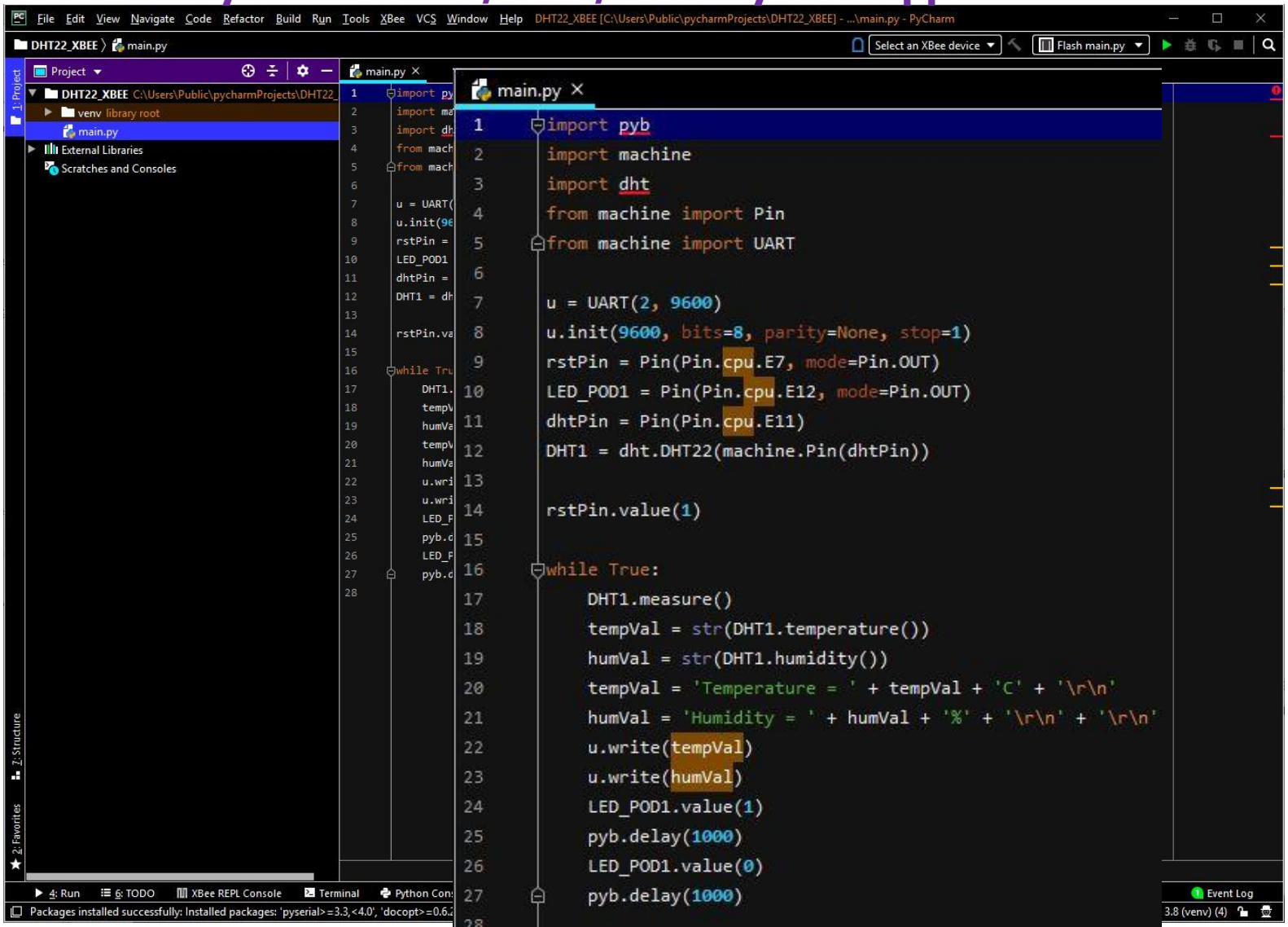

OK

Cancel

Apply

# XBee Radio Modules

## PyCharm XBee/ARM/MicroPython Application

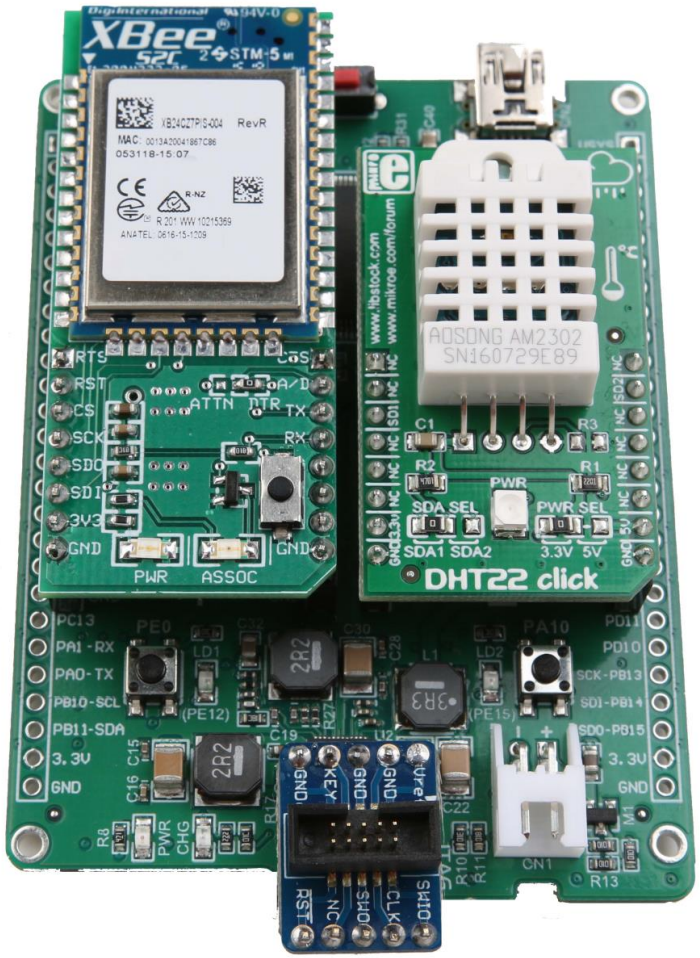


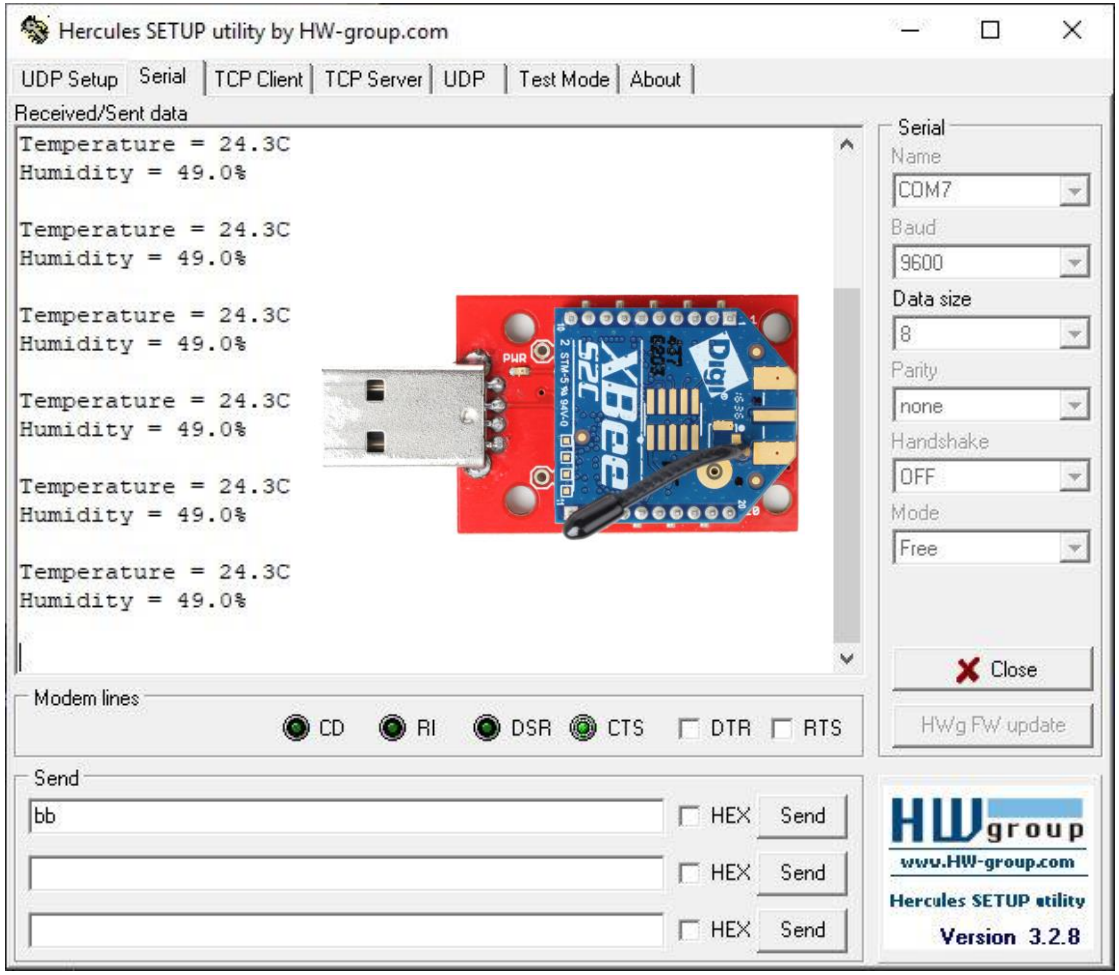



Presented by:

# XBee Radio Modules

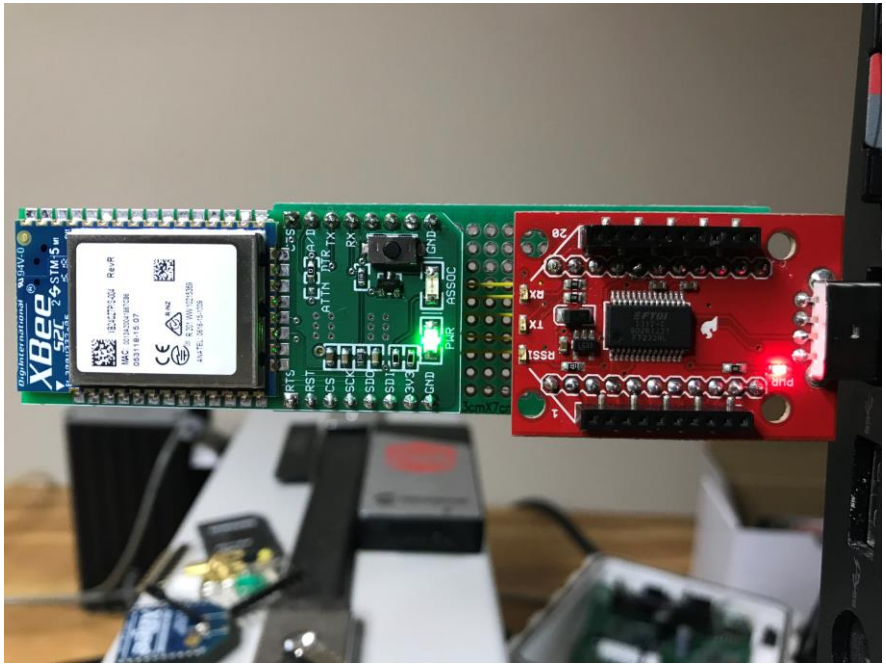
## PyCharm XBee/ARM/MicroPython Application

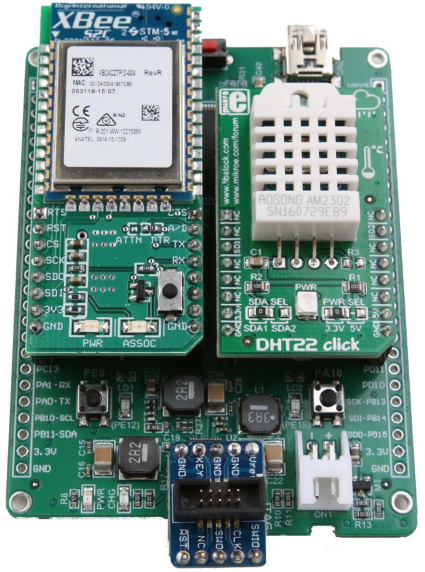


# XBee Radio Modules

## Day 2 Summary







```
main.py ×
1  import pyb
2  import machine
3  import dht
4  from machine import Pin
5  from machine import UART
6
7  u = UART(2, 9600)
8  u.init(9600, bits=8, parity=None, stop=1)
9  rstPin = Pin(Pin.cpu.E7, mode=Pin.OUT)
10 LED_POD1 = Pin(Pin.cpu.E12, mode=Pin.OUT)
11 dhtPin = Pin(Pin.cpu.E11)
12 DHT1 = dht.DHT22(machine.Pin(dhtPin))
13
14 rstPin.value(1)
15
16 while True:
17     DHT1.measure()
18     tempVal = str(DHT1.temperature())
19     humVal = str(DHT1.humidity())
20     tempVal = 'Temperature = ' + tempVal + 'C' + '\r\n'
21     humVal = 'Humidity = ' + humVal + '%' + '\r\n' + '\r\n'
22     u.write(tempVal)
23     u.write(humVal)
24     LED_POD1.value(1)
25     pyb.delay(1000)
26     LED_POD1.value(0)
27     pyb.delay(1000)
28
```