

# microbit\_k230 fall detection

---

## microbit\_k230 fall detection

K230 and microbit communication

1. Experimental Prerequisites
2. Experimental wiring
3. Main code explanation
4. Experimental phenomenon

## K230 and microbit communication

---

### 1. Experimental Prerequisites

This tutorial uses a micro:bit. The corresponding example program path is [14.export\microbit-K230\10.Microbit\_k230\_falldown\_detect].

To begin the experiment, you must run the [14.export\CanmvIDE-K230\10.falldown\_detect.py] program on the K230. It is recommended that you download the program for offline operation.

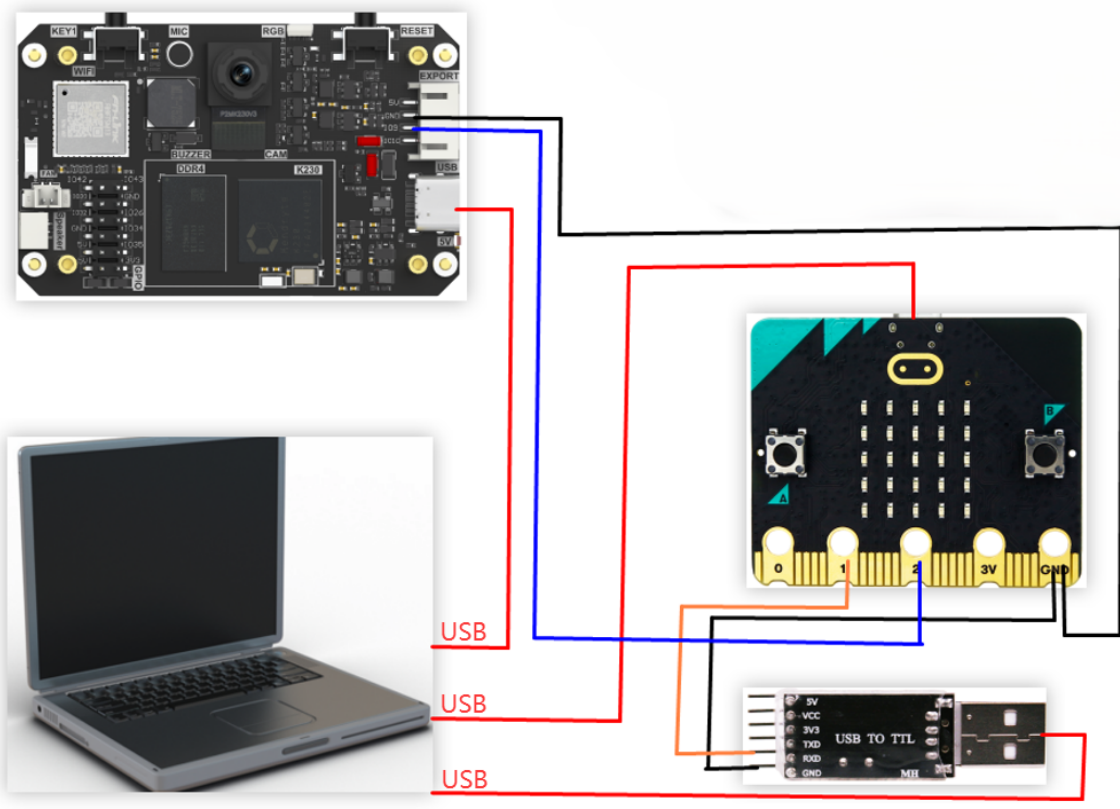
Items needed:

Windows computer, microbit, USB to TTL module, K230 vision module (including TF card with image burned), type-C data cable, connecting cable (Dupont cable), alligator clip, import K230AI library: <https://github.com/YahboomTechnology/K230-Module.git>

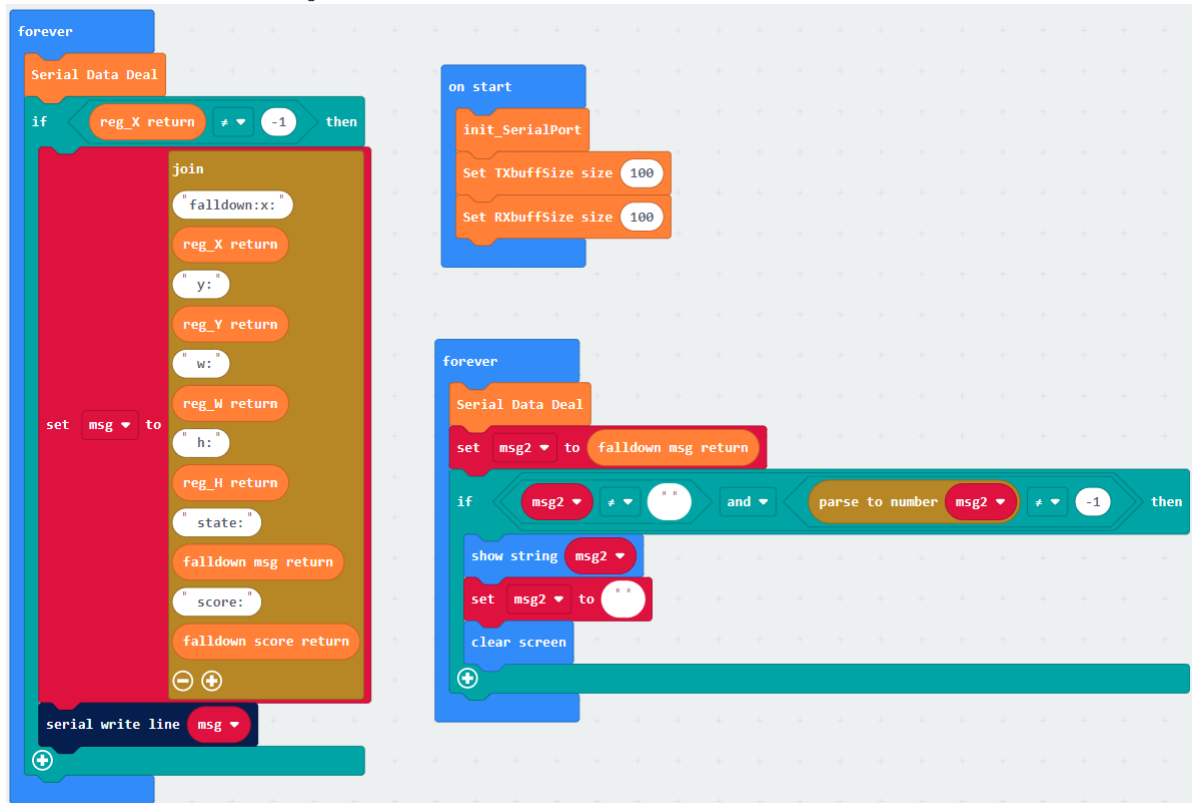
### 2. Experimental wiring

k230 vision module	Microbit
GND	GND
TXD(IO9)	P2

USB to TTL module	Microbit
RXD	P1
GND	GND



### 3. Main code explanation



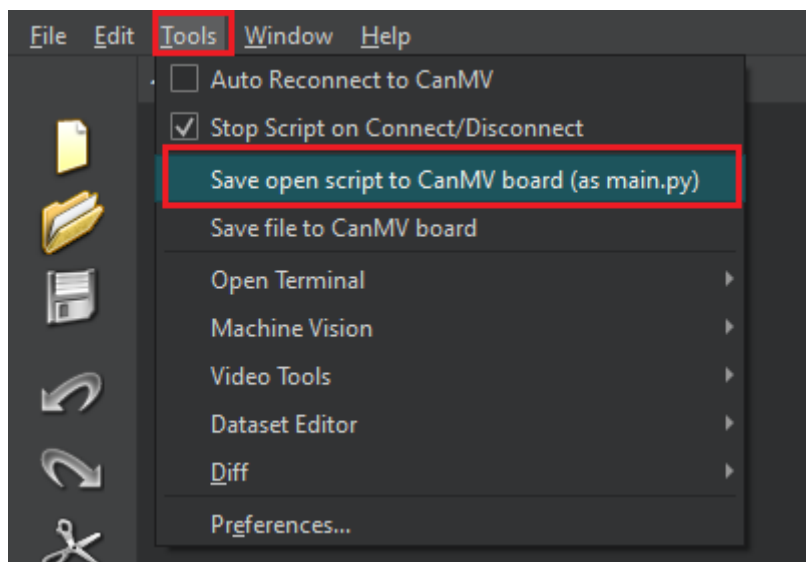
From the code, we can simply configure the serial port and call the relevant serial port and K230 building blocks to obtain data.

- x: The horizontal coordinate of the top left corner of the recognized box
- y: The vertical coordinate of the top left corner of the recognized box
- w: The width of the recognized box
- h: The height of the recognized box
- state: The recognized state. "Fall" indicates a fall, and "NoFall" indicates a non-fall.
- score: The score for the fall state.

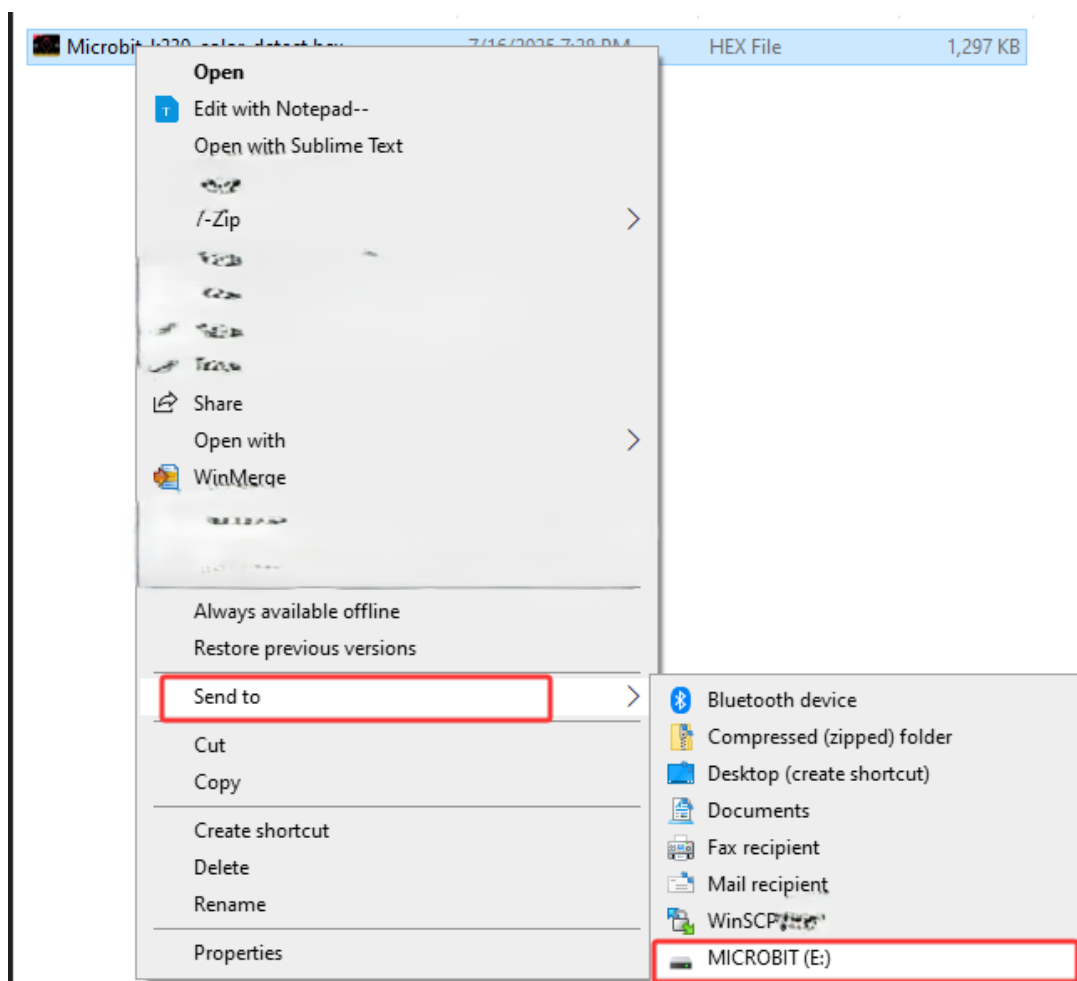
If you want to open the source code of this tutorial, please drag the microbit source code corresponding to this tutorial into the makecode online programming webpage of the browser. The online programming website is: <https://makecode.microbit.org/#>

### 4. Experimental phenomenon

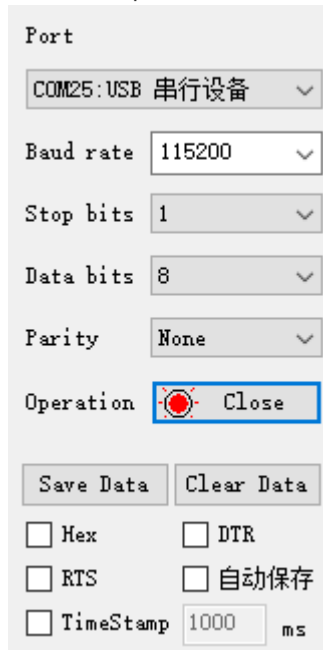
1. After connecting the cables, the k230 visual module runs offline  
After K230 is connected to Canmv IDE, open the corresponding program, click [Save open script to CanMV board (as main.py)] on the toolbar, and then restart K230.



2. Find the hex program of this tutorial, right-click the hex program, and upload the hex program of this tutorial to the microbit



3. The serial port assistant is set to the interface shown in the figure



Port


COM25:USB 串行设备

Baud rate 115200

Stop bits 1

Data bits 8

Parity None

Operation  Close

Save Data Clear Data

☐ Hex ☐ DTR

☐ RTS ☐ 自动保存

☐ TimeStamp 1000 ms

4. When the K230 camera recognizes a human body, the serial port assistant will print out the information transmitted from K230 to microbit.

- x: is the horizontal coordinate of the upper left corner of the identified box
- y: is the vertical coordinate of the upper left corner of the identified box
- w: is the width of the identified box
- h: is the length of the identified box
- state: is the state of identification, the detected fall state is "Fall", the detected non-fall state is "NoFall"
- score: is the score of the fall state

As shown in the figure below

```
falldown:x:96 y:10 w:338 h:333 state:NoFall score:38
falldown:x:107 y:10 w:311 h:333 state:NoFall score:36
falldown:x:100 y:2 w:323 h:406 state:NoFall score:32
falldown:x:93 y:0 w:323 h:354 state:NoFall score:33
falldown:x:100 y:6 w:323 h:406 state:NoFall score:36
falldown:x:93 y:2 w:323 h:334 state:NoFall score:35
falldown:x:98 y:4 w:322 h:406 state:NoFall score:35
falldown:x:107 y:10 w:311 h:333 state:NoFall score:34
falldown:x:107 y:9 w:311 h:333 state:NoFall score:36
falldown:x:99 y:6 w:322 h:406 state:NoFall score:37
falldown:x:89 y:6 w:322 h:406 state:NoFall score:37
```

