

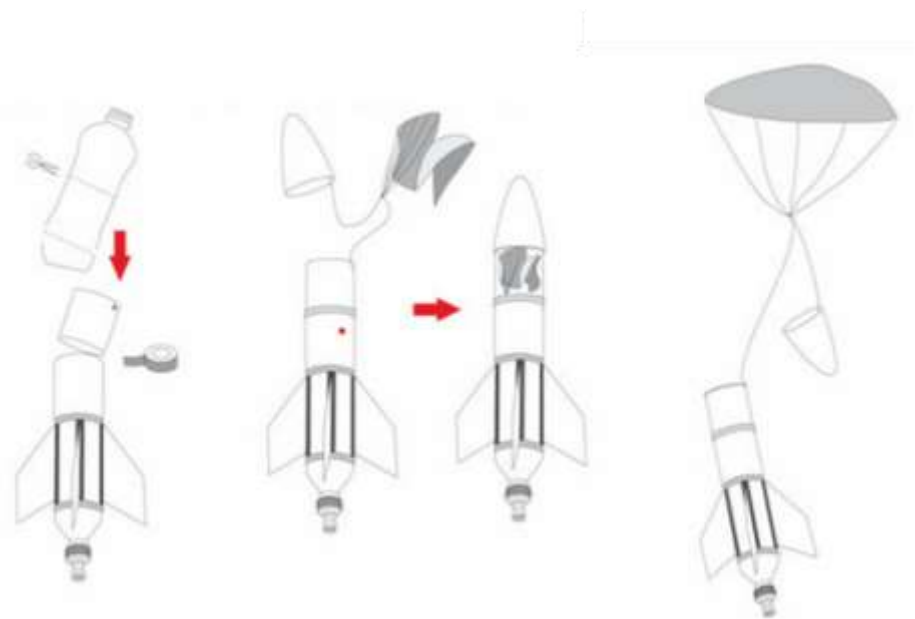
The Education University of Hong Kong

2021-2022 Quality Education Fund Thematic Network – Tertiary Institutes

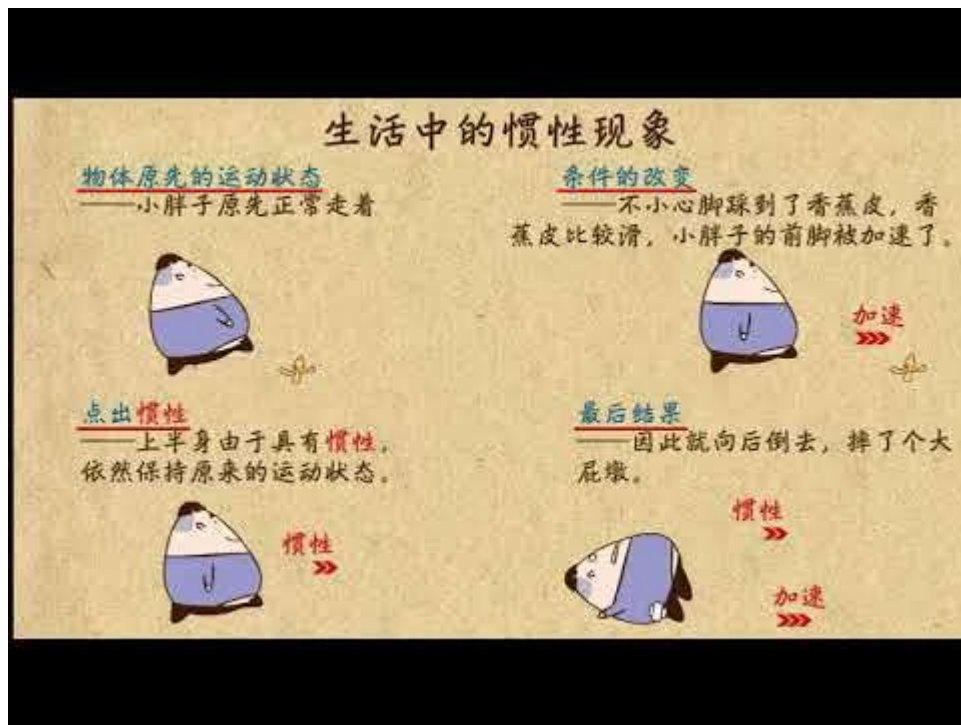
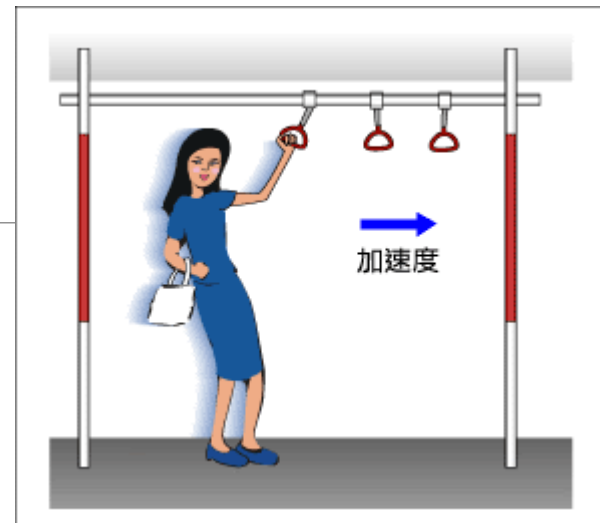
STEM Project Team

SCHOOL: P. H. C. WING KWONG COLLEGE (S2)
TOPIC: WATER ROCKET WITH PARACHUTE 1

(1) 降落傘



慣性 (Inertia)



Design no.1, 500ml*2



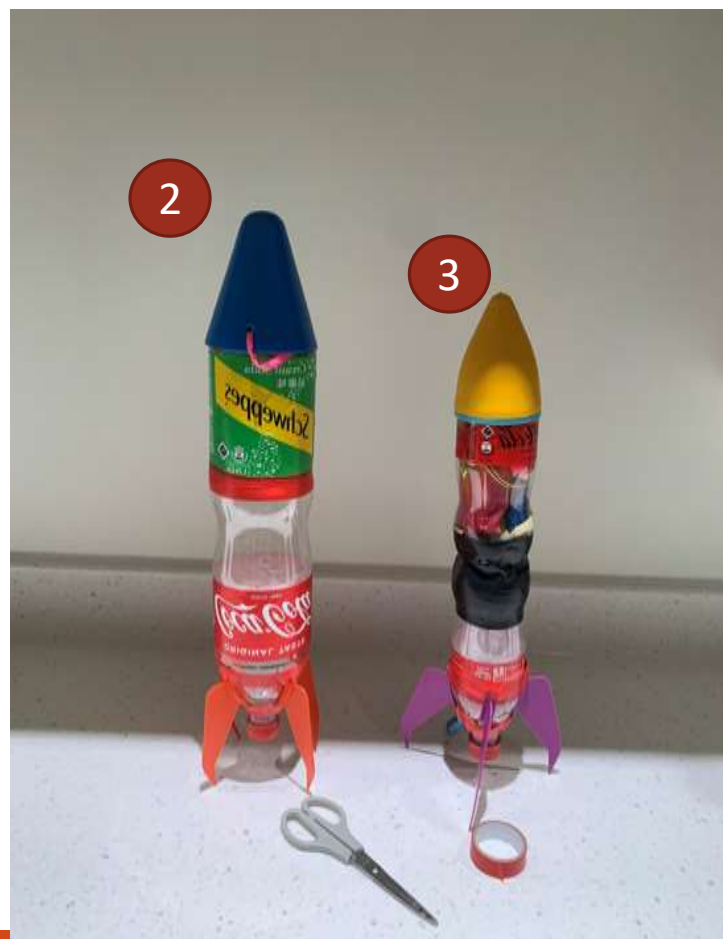
Design no.2, 1.25L*2



Design no.3, 500ml*2



Test



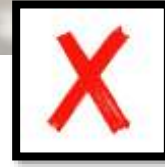
Success Video(1)



Success Video(2)



1



3



500ml



1.25L



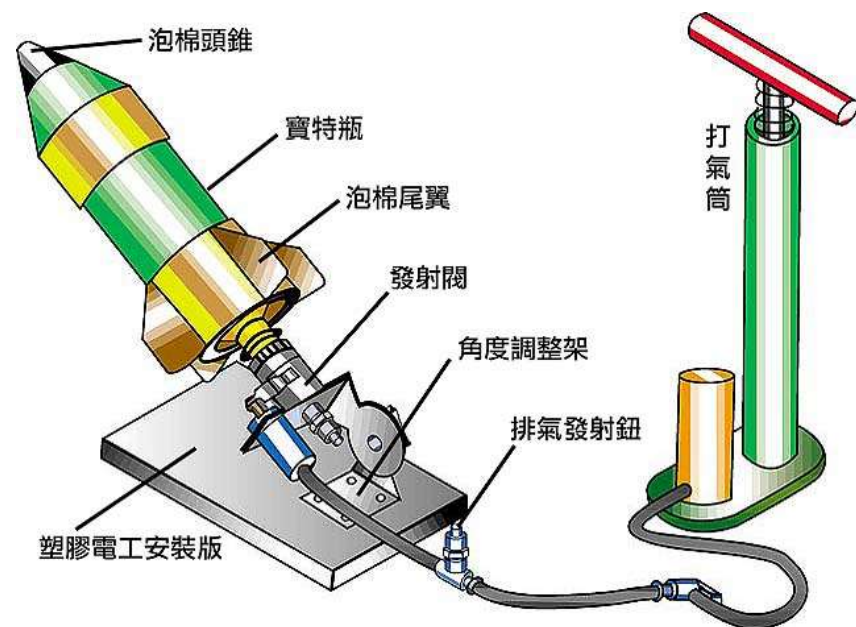
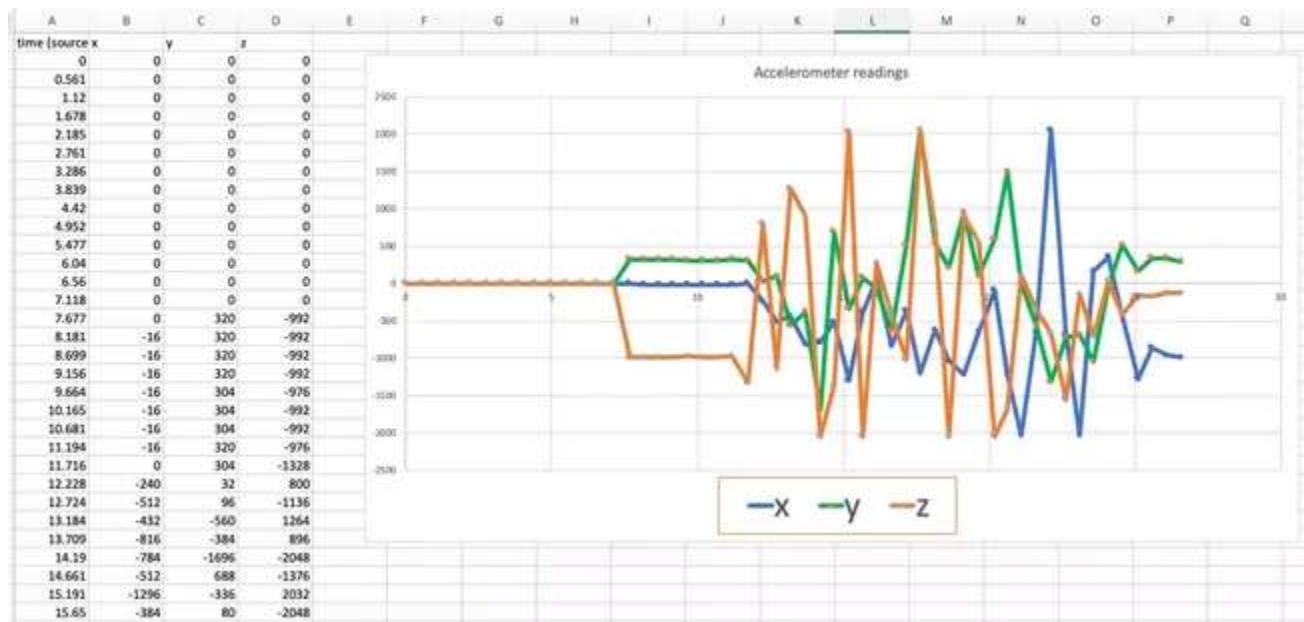
4



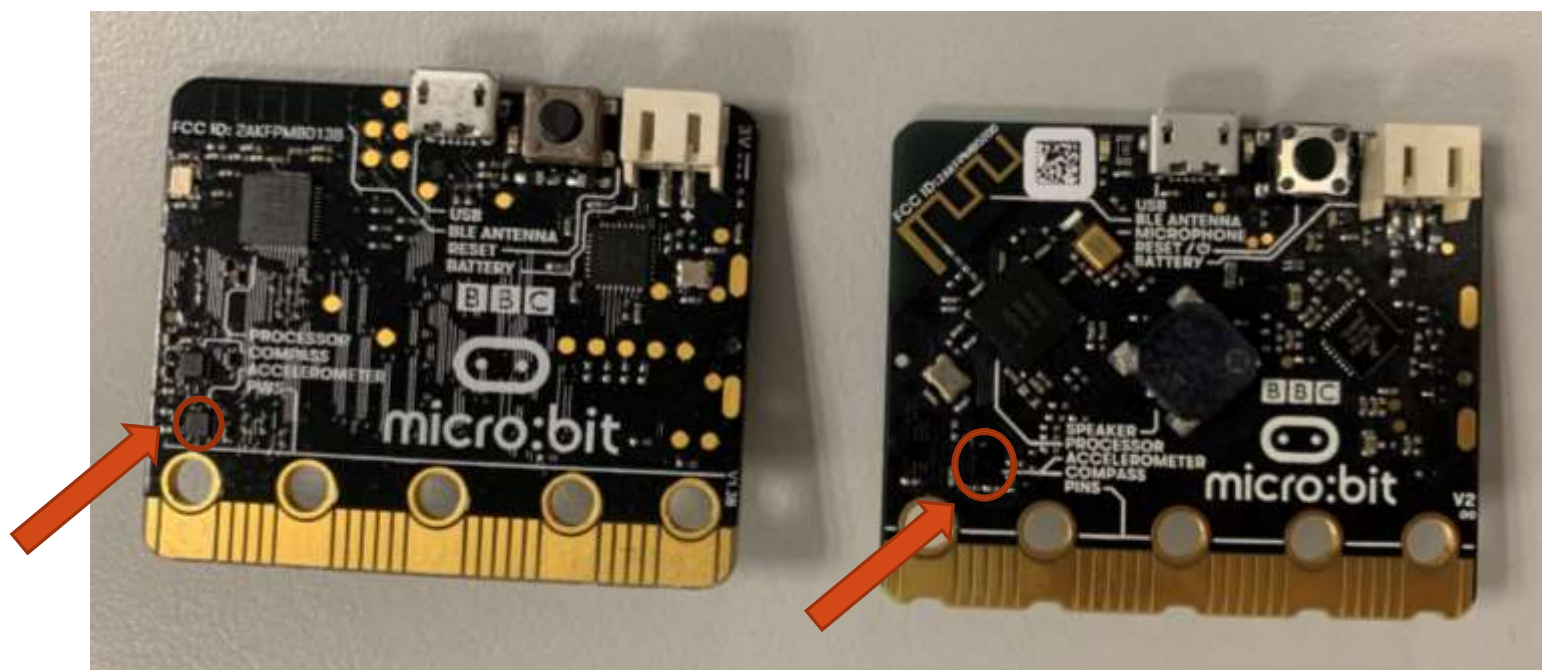
(2) Data logger

(1) Accelerometer (加速計)

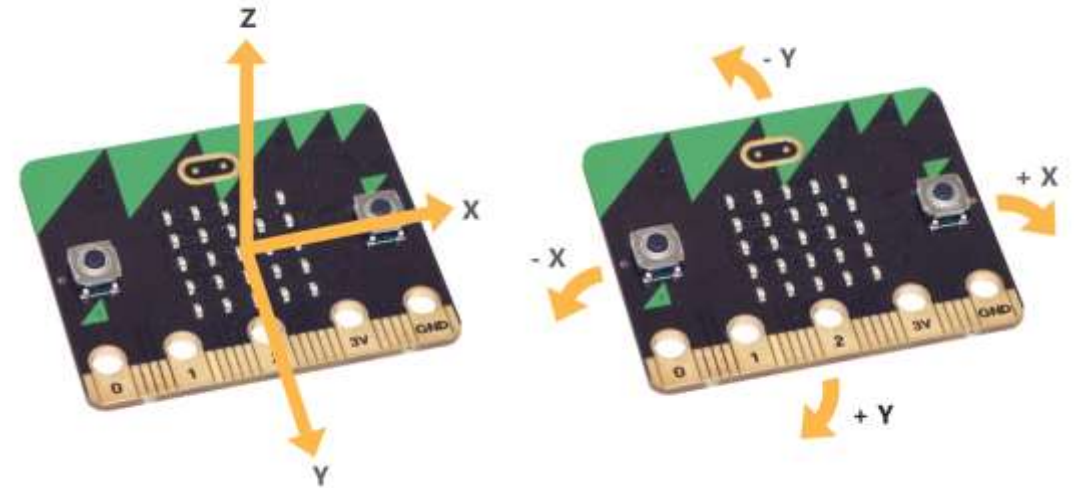
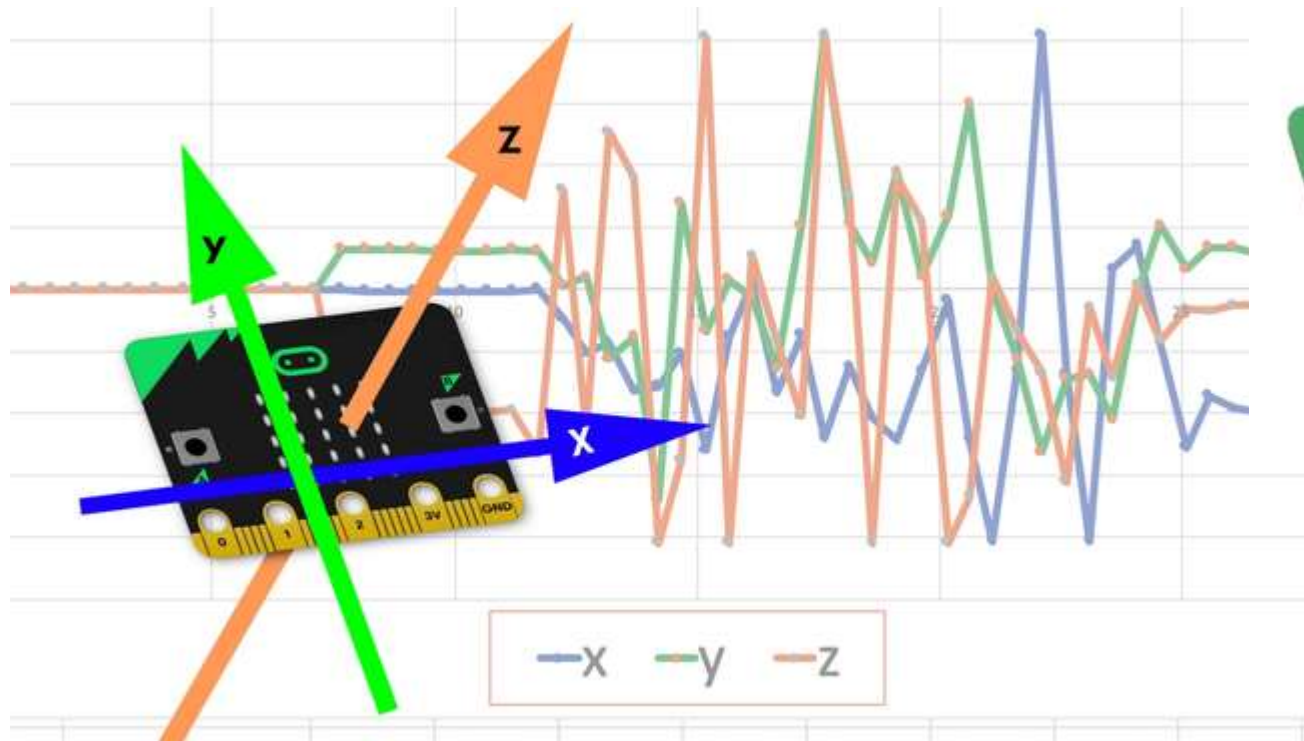
(2) Radio (Transmitter/Receiver)



Micro:Bit - Accelerometer (加速計)



3 Dimensions (x, y and z axes)



X - tilting from left to right.
Y - tilting forwards and backwards.
Z - moving up and down.

Test01, LED Display

Microsoft | micro:bit

Blocks Python

acc

Search

- Basic
- Input
- Music
- LED
- more
- Radio
- Loops
- Logic
- Math
- Advanced

forever

show number acceleration (mg) x

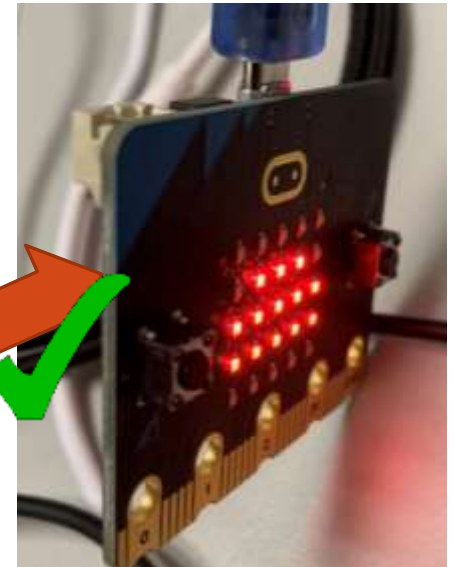
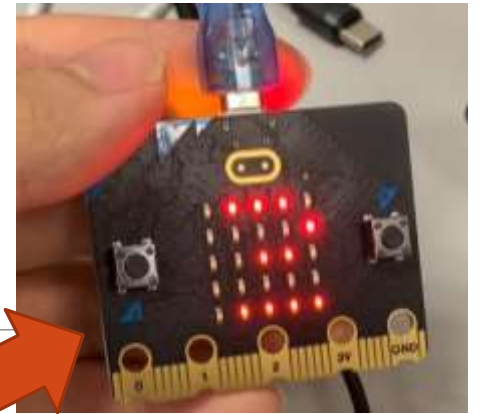
forever

plot bar graph of acceleration (mg) x

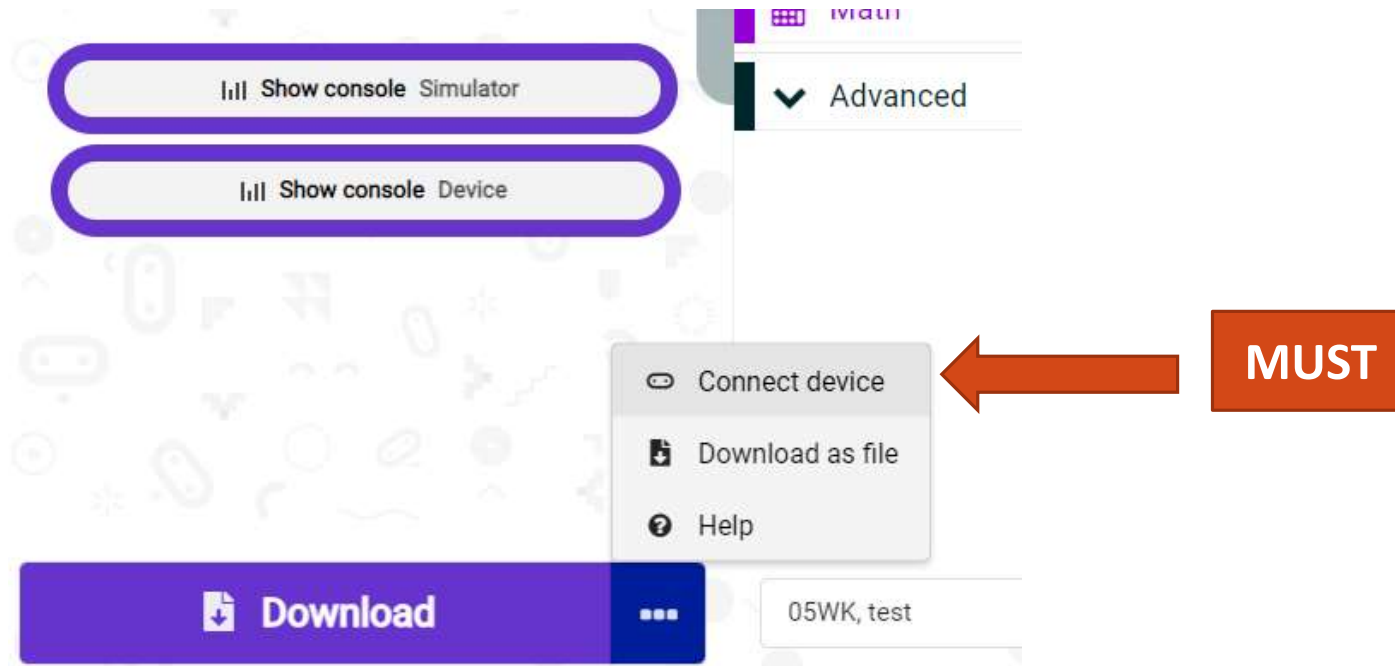
up to 1024

Show console Simulator

Show console Device



Test01, Connect device



Test01, Device Console

Microsoft | micro:bit

Blocks Python

Device

ax:0

Show console: Simulator

Show console: Device

664
1028
724
832
680
692
696
540

Value: ± 2048 ($\pm 1024/1\text{mg}$)
 ~ 0.2 fps (per/5 sec)

Test02, Sensor's code

The screenshot displays the Microsoft MakeCode micro:bit editor interface. On the left, two micro:bit boards are shown. The top board displays real-time sensor data: ax: 476, ay: -738, and az: -525. The bottom board displays ax: 0. The central panel shows a search bar and a category list including Basic, Input, Music, Led, Radio, Loops, Logic, Variables, Math, and Advanced. The right panel contains the code blocks: an 'on start' block with a 'radio set group' block set to 99, and a 'forever' loop containing three 'radio send value' blocks for 'x', 'y', and 'z' axes, each set to 'acceleration (mg)'. An orange arrow points to the 'radio set group' block.

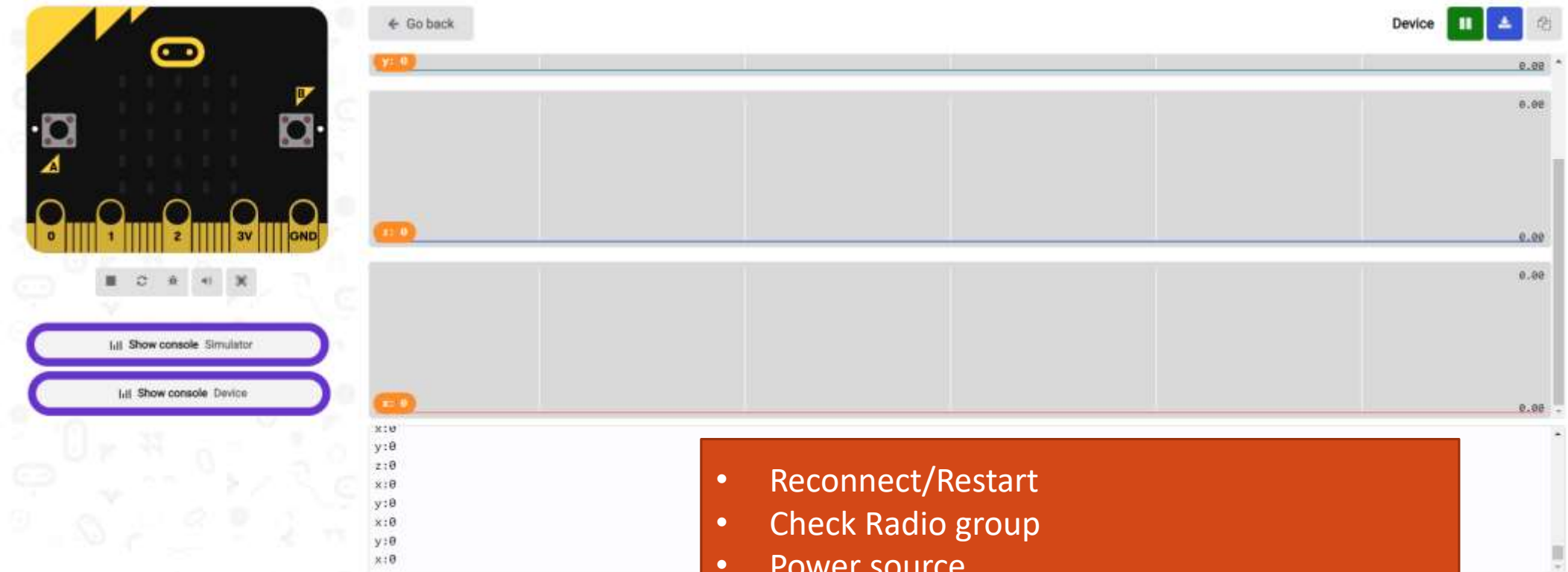
Test02, Receiver's code

The screenshot displays the Scratch IDE interface for a radio receiver project. On the left, a virtual breadboard is shown with a radio module. The central workspace contains the following code blocks:




- on start** block containing a **radio set group** block with the value **99**.
- forever** loop block containing three **serial write value** blocks:
 - serial write value** "x" = x
 - serial write value** "y" = y
 - serial write value** "z" = z
- on radio received** block with **name** and **value** inputs, containing an **if-else-if** structure:
 - if** name = "x" **then** **set x** to value
 - else if** name = "y" **then** **set y** to value
 - else if** name = "z" **then** **set z** to value

An orange arrow points to the **radio set group** block in the **on start** block.

Test02, Device Console (Error)



← Go back

Device   

y: 0 0.00

z: 0 0.00

x: 0 0.00

x: 0 0.00

y: 0 0.00

x: 0 0.00

y: 0 0.00

x: 0 0.00

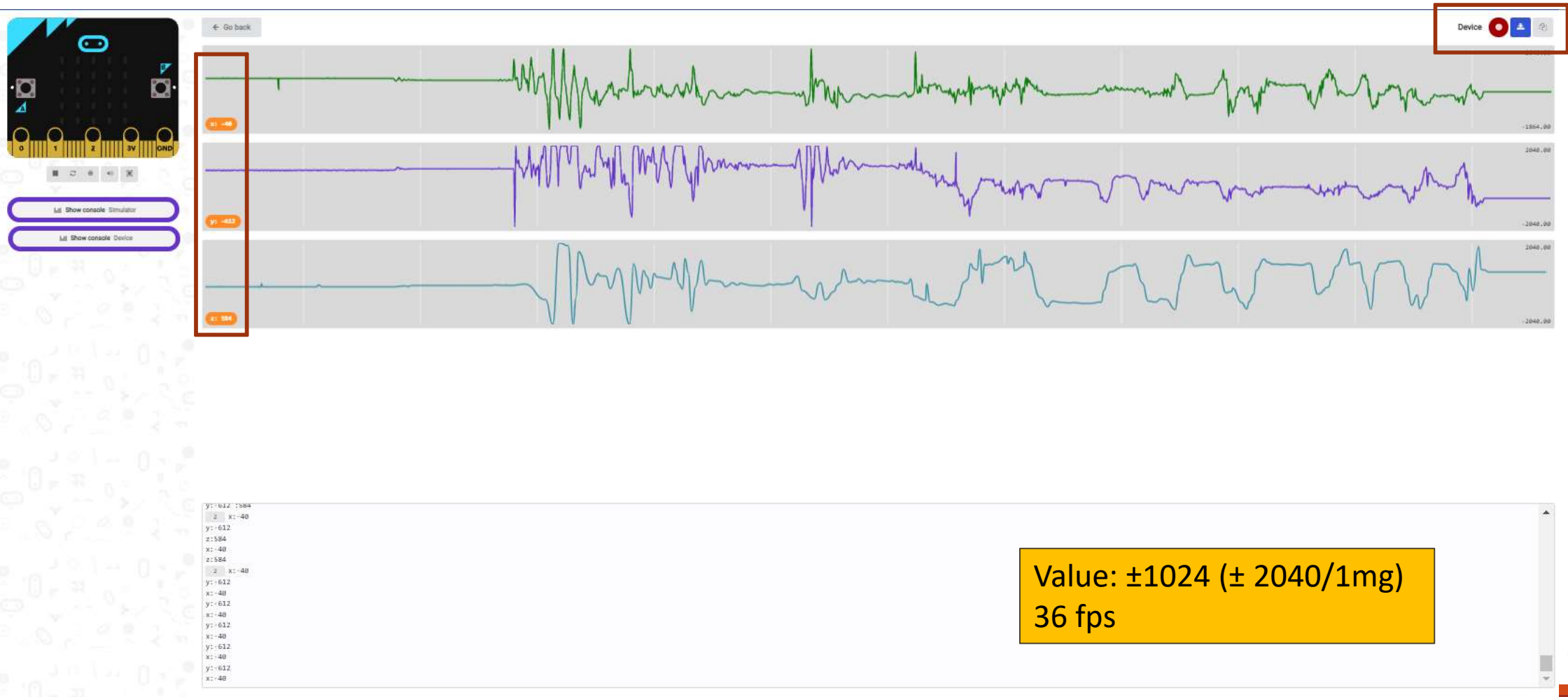
Show console Simulator

Show console Device

x:0
y:0
z:0
x:0
y:0
x:0
y:0
x:0

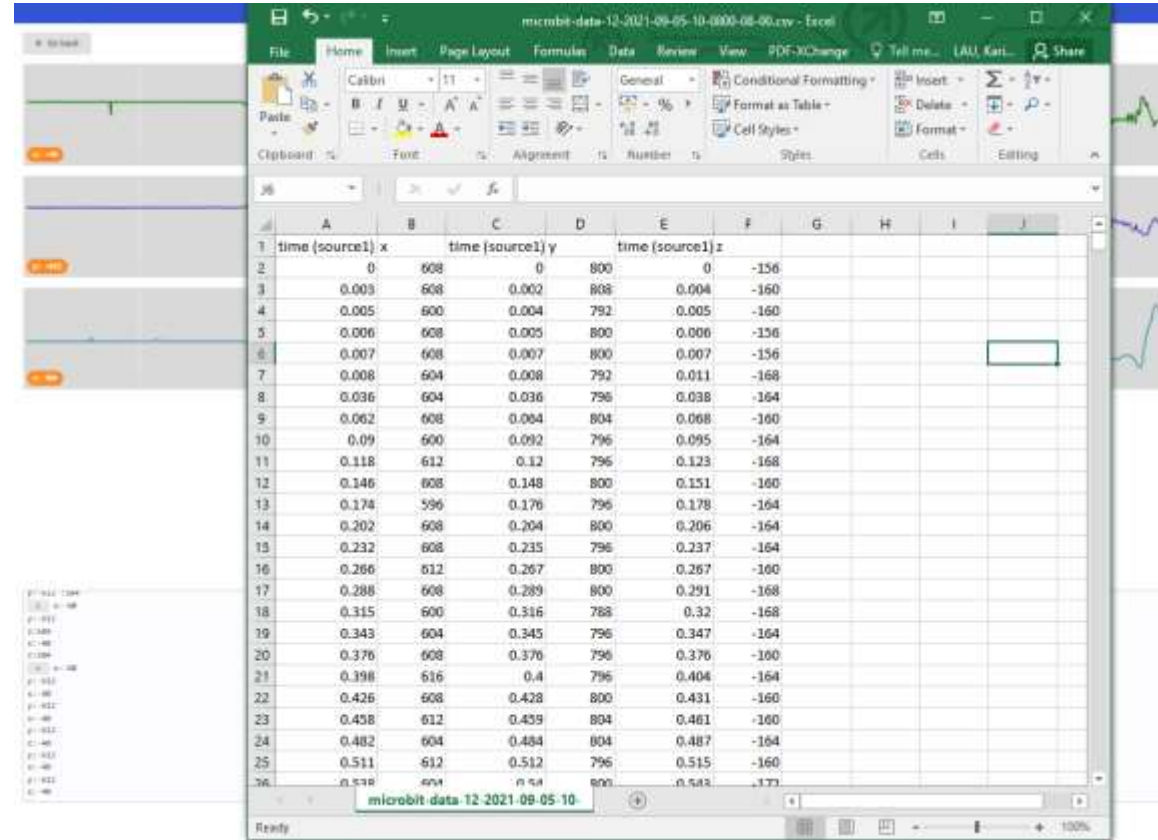
- Reconnect/Restart
- Check Radio group
- Power source
- Check Sensor and Receiver connection cable

Test02, Device Console



Value: $\pm 1024 (\pm 2040/1\text{mg})$
36 fps

Test02, Export Data(excel)

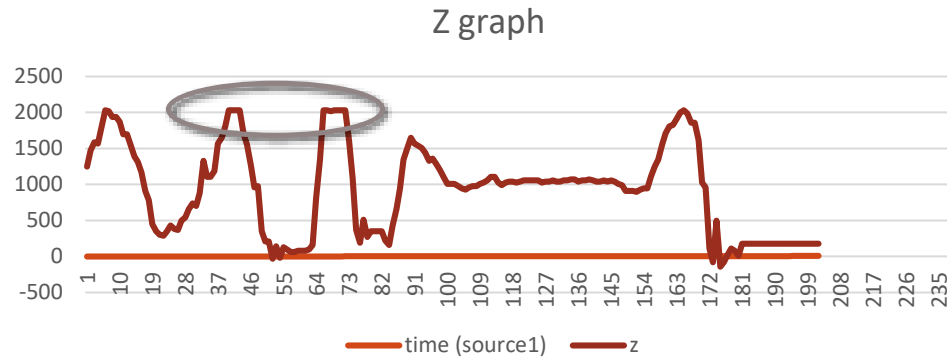


The screenshot shows a Microsoft Excel spreadsheet with the following data:

time (source1)	x	time (source1)	y	time (source1)	z
0	608	0	800	0	-156
0.003	608	0.002	808	0.004	-160
0.005	600	0.004	792	0.005	-160
0.006	608	0.005	800	0.006	-156
0.007	608	0.007	800	0.007	-156
0.008	604	0.008	792	0.011	-168
0.036	604	0.036	796	0.038	-164
0.062	608	0.064	804	0.068	-160
0.09	600	0.092	796	0.095	-164
0.118	612	0.12	796	0.123	-168
0.146	608	0.148	800	0.151	-160
0.174	596	0.176	796	0.178	-164
0.202	608	0.204	800	0.206	-164
0.232	608	0.235	796	0.237	-164
0.266	612	0.267	800	0.267	-160
0.288	608	0.289	800	0.291	-168
0.315	600	0.316	788	0.32	-168
0.343	604	0.345	796	0.347	-164
0.376	608	0.376	796	0.376	-160
0.398	616	0.4	796	0.404	-164
0.426	608	0.428	800	0.431	-160
0.458	612	0.459	804	0.461	-160
0.482	604	0.484	804	0.487	-164
0.511	612	0.512	796	0.515	-160
0.538	604	0.54	800	0.542	-172

Limitation of Accelerometer

± 2040 ~ error



Error :
Accelerometer

$$v = u + at \quad (1)$$

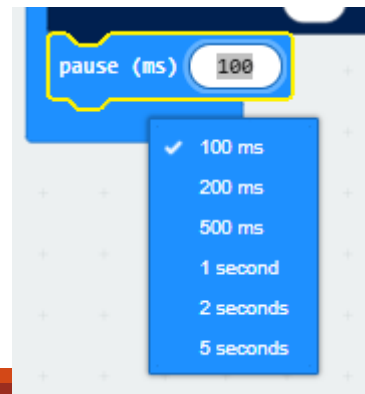
$$s = \frac{1}{2}(u + v)t \quad (2)$$

$$s = ut + \frac{1}{2}at^2 \quad (3)$$

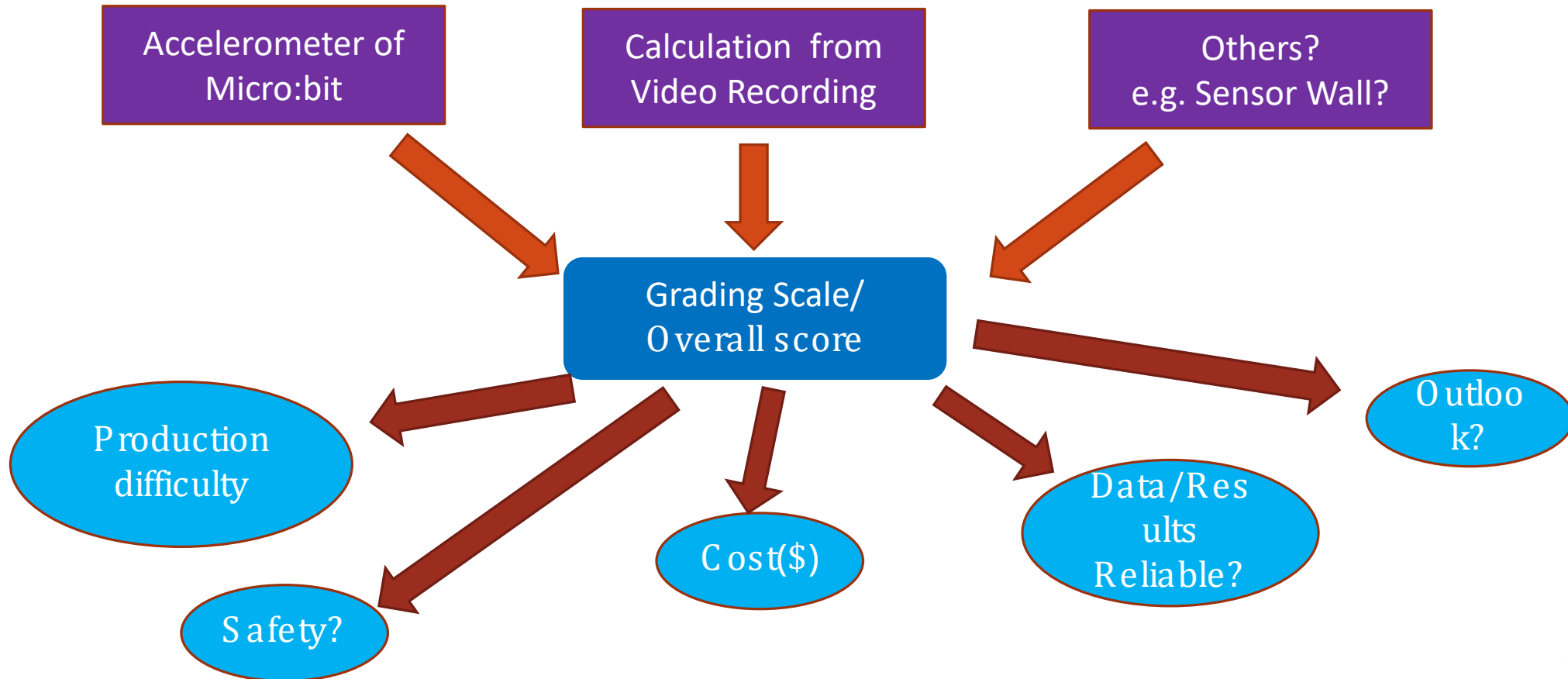
$$v^2 = u^2 + 2as \quad (4)$$

$$s = vt - \frac{1}{2}at^2 \quad (5)$$

Time



Engineering Design Comparison



Notes

- Is 36 fps enough?

>>Enough for accelerometer(between 10Hz to 100Hz)

>>For reference about human eye, animation are mainly 24fps

- Transmitter/Sensor & Receiver?

>> Show leds?

- Range of micro:bit Transmit Power?

>>Offical 70m (<https://makecode.microbit.org/reference/radio/set-transmit-power>)

>>Normally, at least 15m

