Co-funded by the Erasmus+ Programme of the European Union

Padova

Istituto di Istruzione Superiore Pietro Scalcerle

M O T I O N

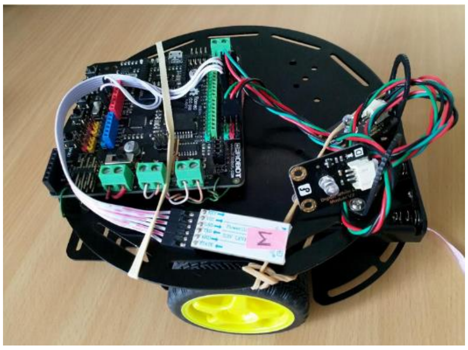
10.12. – 13.12.2019

Topic	Age	Country	Date
2-wheel robot's movement	>14	France	Padova Dec 2019

Objective:

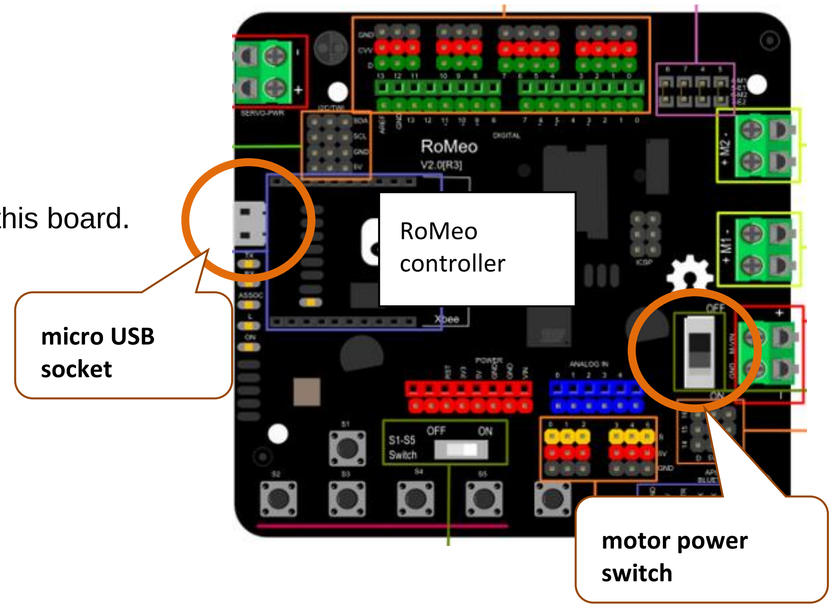
During this activity students will modify a C code to move a 2-wheel robot.

This robot can simulate the behavior of an autonomous vacuum cleaner like the one shown in the picture beneath.



The robot's brain is the RoMeo controller.

You will use the Arduino IDE to program this board.



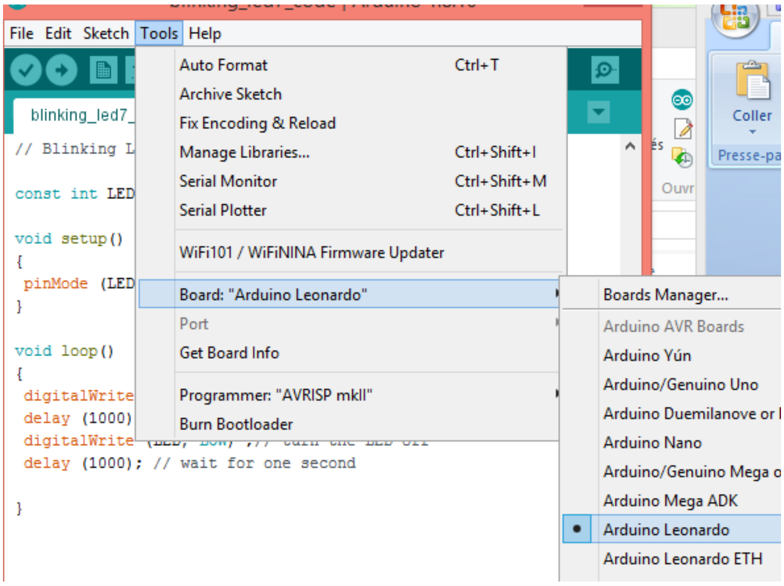
Please **Turn OFF the Motor Power Switch** when Romeo is connected to the computer through USB cable. Or the external power supply(>12V) will destroy your Romeo.

STEP 1 Blinking Led code - first step with Arduino IDE

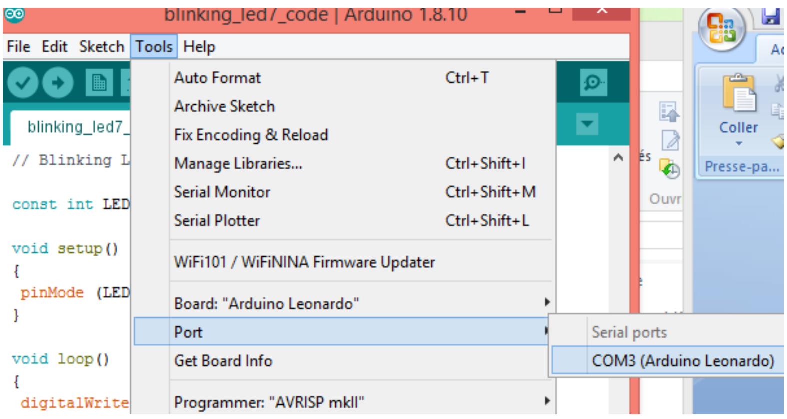
Using the USB wire, **connect** your RoMeo board to the computer. The micro USB socket is on the RoMeo's left side (see the picture on the first page).

Open the code "blinking_led3_code":

Click **Tools**, then Board and select **Arduino Leonardo** in the list.



To choose the serial port, click **Tools>Port** and **select the COM** where Arduino Leonardo is connected to.



Click on the **arrow Upload** to upload the sketch (code) to Romeo:

If your sketch has uploaded successfully, **you will now see this LED happily flashing on and off slowly on your board**.

If so, **congratulations**, you have just successfully uploaded and ran your first sketch.

Have a look at the "blinking_led3_code" and find the command that sets the blink's time.

Then, **change the time** to **100** milliseconds and upload your new sketch.

Find, in the code, the statement that switches the LED on.

You will use a similar statement to rotate motors.

blinking_led3_code

```
// Blinking Led code

const int LED=3; // LED connected to digital PIN 3

void setup()
{
 pinMode (LED, OUTPUT) ; // LED is an OUTPUT
}

void loop()
{
 digitalWrite (LED, HIGH) ;// turn the LED on
 delay (1000); // wait for one second
 digitalWrite (LED, LOW) ;// turn the LED off
 delay (1000); // wait for one second
}
```

STEP 2 Testing robot's movement - first step in coding robots

Insert the batteries in your robot.
Connect your robot to the PC. **Return** your robot (so it will not move on the table)..
Open the code "**Romeo_motor_forward** ". Click **Tools>Port** and **select the right COM.**
Upload the code.

Remove the USB wire, put the robot on the floor and slide the power switch to **ON**.
After some seconds your robot will move forward and stop.

STEP 3 Moving the robot about 50 cm forward.

Slide the power switch to OFF.

Question *How much duration (delay statement) do you need to go forward about 50cm?*

Hint:
If the wheel makes two revolutions, how far the robot will go?

Explain your answer.

Now, you have to modify the previous code (the part shown in the picture) so the robot moves straight and its wheels make about two rotations.

```
// move forward during 1 second:
analogWrite (E1,120);  //speed 120
digitalWrite(M1,LOW);
analogWrite (E2,120);  //speed 120
digitalWrite(M2,LOW);
delay(1000); //wait for 1000 ms, so
```

Change the parameters in **analogWrite** **(E2, 120)** and in **delay** **(1000)** to have a straight movement and two revolutions of the wheels. The marks on the wheels can help you.

Save your new code as **"Romeo_motor_forward_1"**.

Upload your code, remove the USB wire, put the power switch to ON and TEST the robot's movement.

STEP 4 Moving the robot about 25 cm backward.

Slide the power switch to OFF.

To reverse the direction of rotation, you have to change the polarity of the supply voltage.

For example, if you want the M1 motor to move backward, you have to change the statement : **digitalWrite** (M1, LOW) to **digitalWrite** (M1, **HIGH**).

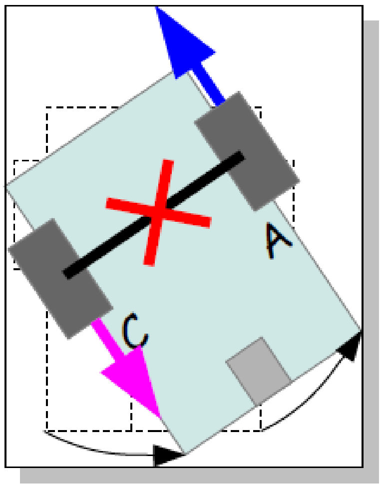
Add some statements to the previous code so your robot drives forward 50cm, then stops for 3 seconds, moves backward about **25cm** and stops.

Save your code as "Romeo_forward_backward".

STEP 5 Turn the robot 180°

Slide the power switch to OFF.

There are several methods to turn the robot. In this activity you have to program the one shown in the picture.



Question:

How much duration does the wheel need for the robot to turn 180°?

Explain your answer.

Add to the previous code some statements to turn your robot.

Keep experimenting until it is almost perfect!
Save your code as "Romeo_move_turn".

Extra tasks:

Task 1 Movement and blinking Leds

Connect a green Led to the Pin 3 and a red Led to the Pin 2.

Modify your code "Romeo_move_turn" to have the green Led switched on when the robot moves forward, the red led switched on when the robot moves backward and the both Leds switched on when the robot turns.

Task 2 Moving around a square

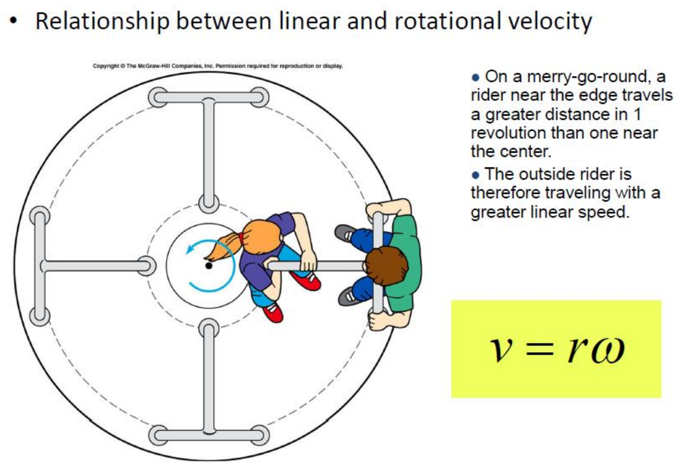
Make your robot drive around a square.

Task 3 Calculating speeds

Calculate the robot's velocity: **v (m/s)**

Calculate the robots rotational speed: **N** (rotation/minute) and **ω** (radian/second).

Check if the calculated velocities satisfy the relation shown in the picture:



Task 4 Edge detection

Connect an edge detection sensor to your robot and write a code that will make the robot move on a table (<u>not black surface</u>). The sensor value is **LOW** or **0** when an object is detected (<10cm) and **HIGH** or **1** when an edge is detected.

See the "edge sensor hint code", if you need some hints.

A black surface must be 4-5 cm from the sensor (if less, the sensor reads 1).

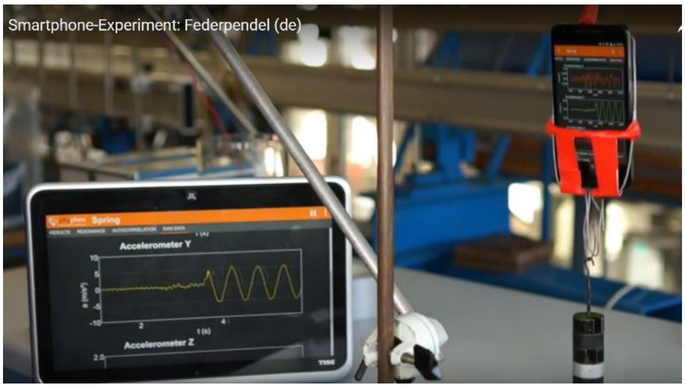
https://wiki.dfrobot.com/DFRobot_Infrared_sensor_breakout__SKU_SEN0042_

Topic	Age	Country	Date
Data transfer (phyphox)	>14	Germany	Oct. 2019

Remote control and data transfer

The picture on the right shows the swinging mobile phone with the tablet showing the elongation over time. That´s a great advantage since you can read the data immediately during the experiment without looking at the smartphone.

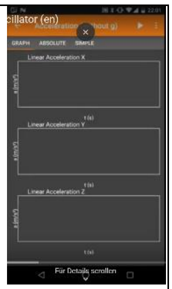
Instead of a tablet one can also use a laptop or a second smartphone.

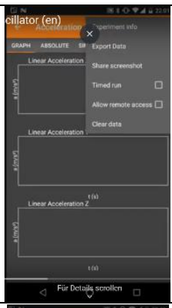
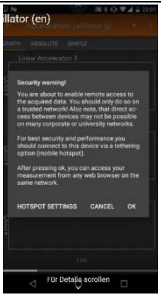
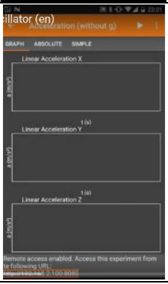


There are two possibilities to connect your smartphone with your tablet

1. WIFI
   Both devices are logged onto the same WIFI!

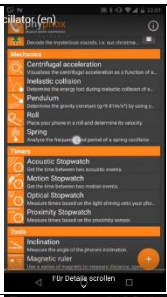
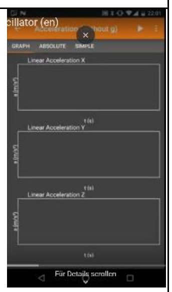
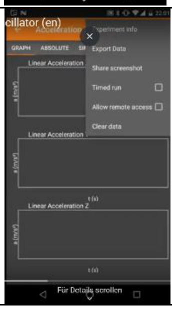
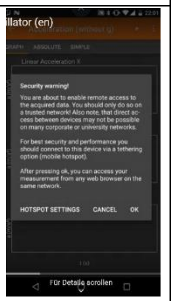
   (Unfortunately schools and hotels use different suberservers, sometimes it is the same.)

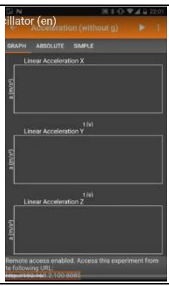
1.) Open the Phyphox App und click to the application “Spring”.		2.) Open the menu bar by clicking on the three dots on the top right on the screen.	

3.) Activate “Allow remote access”. 	4.) Click OK. 
5.) A URL - address will be shown at the bottom of the screen. 	6.) Enter this address to the web-Browser on your tablet. If there are more address numbers, try them all. 

2) <u>Mobile Hotspot</u>

Activate a hotspot on your smartphone.

1. Deactivate your WIFI on your smartphone. Open a Mobile Hotspot on your smartphone.	2. If you have an adroid smartphone deactivate *mobile data* (this option is not available on iPhones).
3. Connect your tablet with your smartphone: To do so you have to click the WIFI symbol on your tablet and find the name of your smartphone.	4. Now both devices use the same network.
5. Open the Phyphox App und click to the application “Spring”. 	6. Open the menu bar by clicking on the three dots on the top right on the screen. 
7. Activate “Allow remote access” 	8. Click **OK.** 

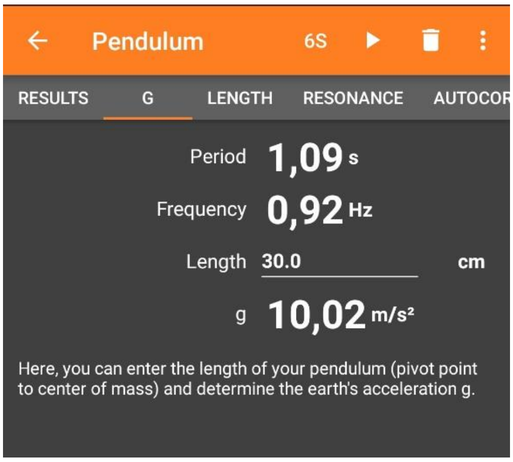
9. The URL - address will be shown at the bottom of the screen. 	10. Enter this address to the web-Browser on your tablet. If there are more address numbers, try them all. 

Topic	Age	Country	Date
(Simple) Pendulum with Phyphox	>14	Germany	Oct 2019

- toilet roll
- some string
- a construction to fix the pendulum (-> use information in the app)

1) You can e.g. determine the gravity constant by using your smartphone as pendulum. You have to enter the length L of the pendulum. With the including gyroscope it's possible to measure the period T. ($g = \frac{4\pi^2 \cdot L}{T^2}$)
2) You can plot the amplitude against the detected frequency.
3) You can discover the formula $f = \frac{1}{2\pi} \cdot \sqrt{\frac{g}{l}}$

<u>Important</u>: You can use the possibility of a timed run (enter *start delay* and *duration*) or use a second smartphone/ tablet/ laptop/ PC for starting and stopping the measurement.



Topic	Age	Country	Date
Spring pendulum with Phyphox	>14	Germany	Oct. 2019

Experimental setup materials:

1 smartphone and a smartphone case

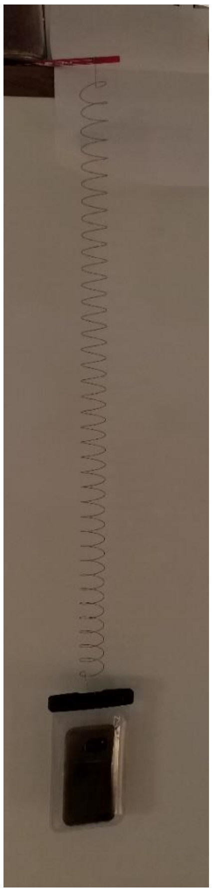
1 tablet/laptop

1 pencil/ mounting device

scotch tape

2 springs

weights



Experimental setup and procedure:

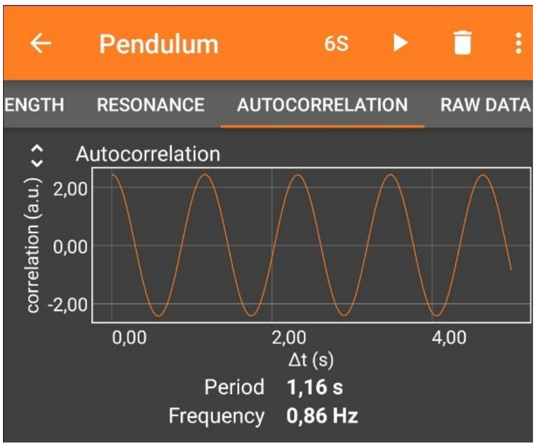
Start **Phyphox** App on your smartphone and setup the experiment as shown in the picture. You can either use the mounting device or a pencil with some tape. Be aware, the smartphone position in the case must be **vertical**.

You can use the possibility of a timed run (enter *start delay* and *duration*) or use another smartphone/ tablet/ laptop/ PC for starting and stopping the measurement.

Evaluation of the experiment:

By quality:

1) Describe the t-y-diagram. Discuss/explain the correlation between oscillation and the period T by using the t-y-diagram. Sometimes it is useful to click the Pause button.
2) Modify the elongation of the oscillation (not varying weight and spring) and observe the variation of period T.
3) Modify the mass of your smartphone, you could tape weights to the case. Observe the variation of period T.
4) Modify the spring constant D (be aware, use the same mass and elongation as in the earlier experiments) and observe the period T.



By quantity: (influence on T)

5) Modify the mass and enter the measured data of the period in an m-T-diagram.
6) Modify the spring constant and enter the measured data of the period in a D-T-diagram.
7) Do you find a correlation between T, D and m?

Topic	Age	Country	Date
Damped spring pendulum with Phyphox	>14	Germany	Oct 2019

Damped spring pendulum

Experimental setup materials:

1 smartphone

1 tablet/laptop

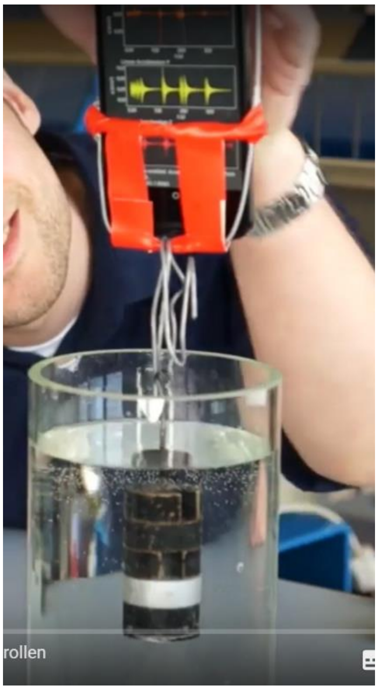
1 pencil/ mounting device

Scotch tape

1 spring

weights

1 glass of water



Experimental setup and procedure:

Start the **Phyphox App** on your smartphone and setup the experiment as shown in the picture. Fix the weights to the smartphone case with Scotch tape. Be aware, the smartphone position in the case must be **vertical**.

Elongate the smartphone and start the experiment on the **tablet/laptop screen**. In order to do so you must click the **play button** (on the top right hand side of the screen). Select the button **acceleration.** For this experiment limit the time of the measurement to 5 seconds.

Experiment evaluation:

8) Describe the t-y-diagram.

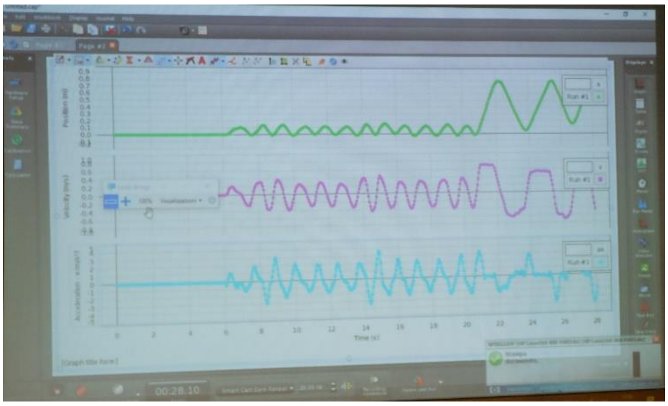
9) Explain the differences between the t-y-diagram of the harmonic oscillator and the damped oscillator!

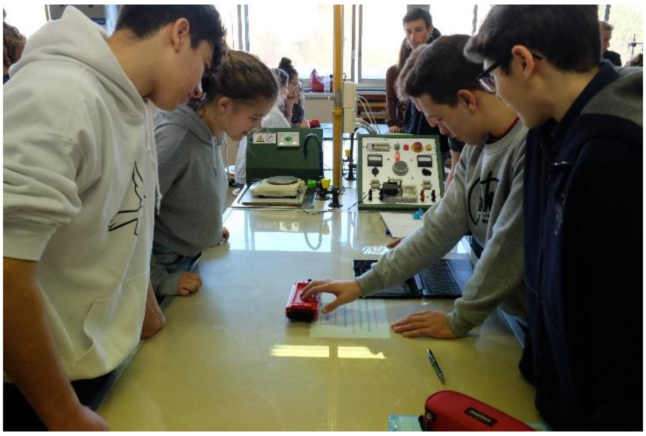
10) Guess what will happen if you exchange the water for jelly? Draw a rough t-y-diagram for this case.

Topic	Age	Country	Date
Motion (Capstone)	>14	Italy	Dec 2019

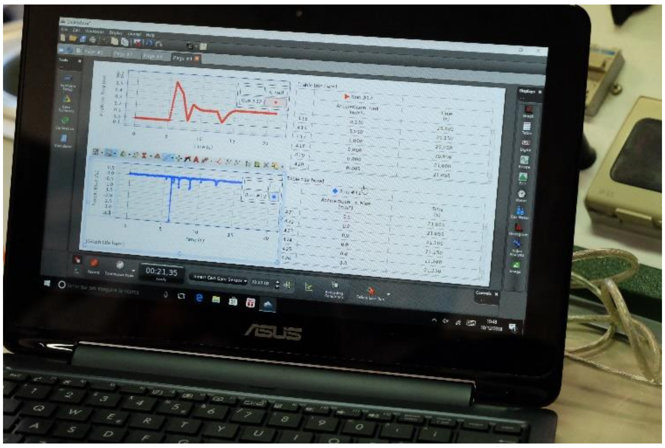
A smart cart is moved about the desk. It sends data directly to the computer.

Position, speed and acceleration are visualized by the program.





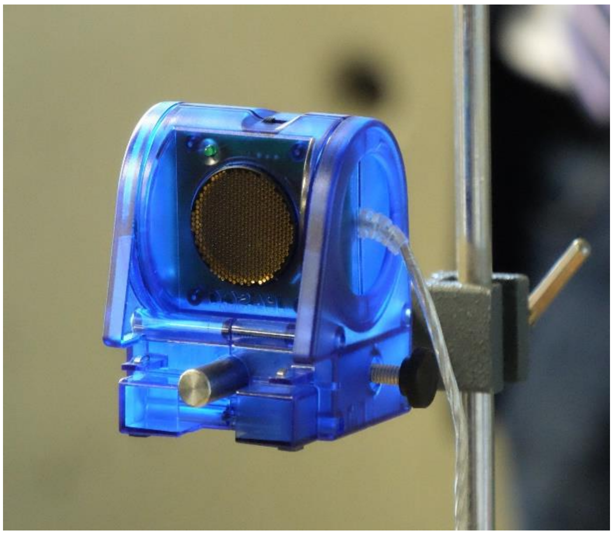
The movements of two different carts can be compared.

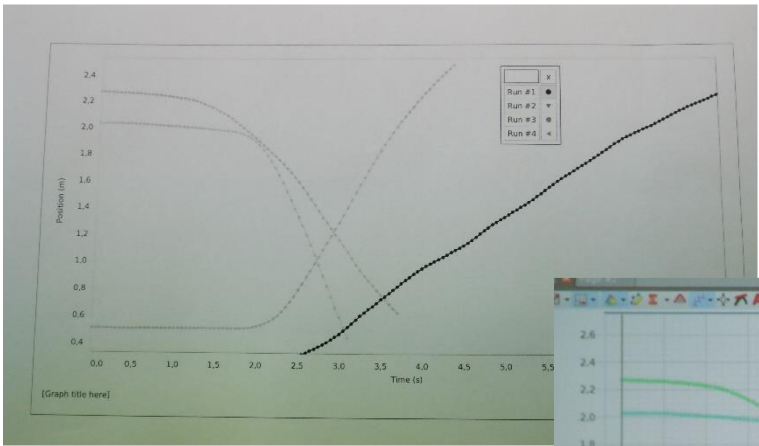


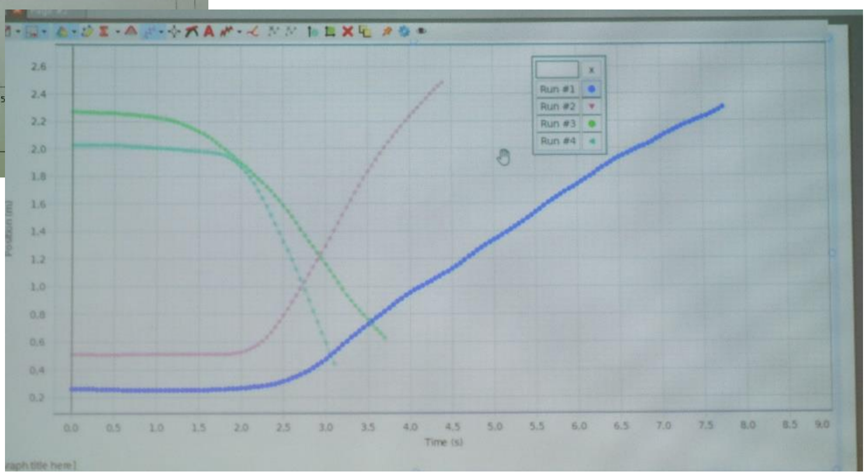
Topic	Age	Country	Date
Motion (Pasco)	>14	Italy	Dec 2019

Position-time graphs are created by linear motion of students. Different graphs can be compared and shaped to match one another.

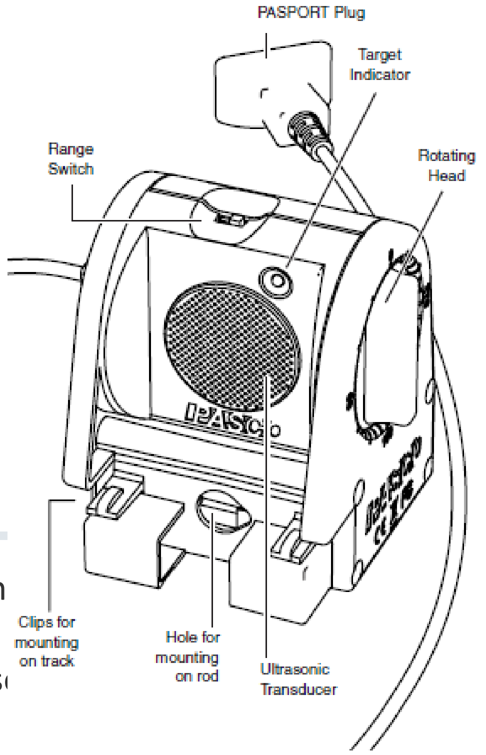
The students are asked to create a given graph.







PASPORT Motion Sensor - PS-2103A



How It Works

An electrostatic transducer in the face of the Motion 16 ultrasonic pulses with a frequency of about 49 reflect off a target and return to the face of the s flashes when the transducer detects an echo.

The sensor measures the time between the trigger rising edge and the echo rising edge. It uses this time and the speed of sound to calculate the distance to the object.

Specifications

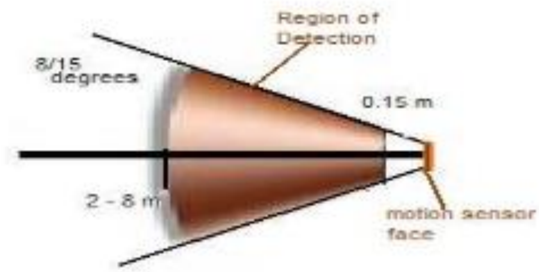
Range	• 0.15 to 8 m
Resolution	• 1.0 mm
Maximum Sample Rate	• 50 Hz
Transducer Rotation Range	• 360°

The normal range of sampling rates is between 1 Hz and 50 Hz. At the default rate, the Motion Sensor can measure distance up to 8 m. The maximum distance decreases with increasing sample rate. At very high sample rates (between 50 Hz and 250 Hz), the maximum distance is less than 2 m.

To Aim the Motion Sensor at an Object

1. Set the range switch to the short range () or long range () setting.

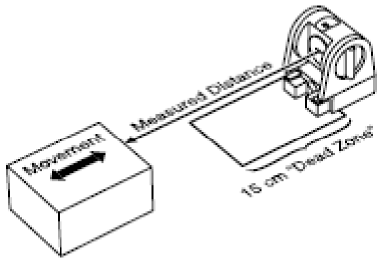
- Select  for measuring a cart on a track.
- Select  for measuring most other objects.



 15°  8° small cone angular opening

2. Arrange the Motion Sensor and object so that the Motion Sensor's transducer faces the object.

- The object should be at least 15 cm from the transducer.
- If the object will move, it should move directly toward or away from the Motion Sensor.
- Aim the motion sensor slightly up to avoid detecting the tabletop.



3. Remove objects that may interfere with the measurement. These include objects between the sensor and target object, either directly in front of the sensor or to the sides.

Experiment: Understanding Motion - Distance and Time (Motion Sensor)

Concept: linear motion

EQIPMENT NEEDED

- PASCO Capstone Software
- PASCO USB Link Interface
- PASPORT Motion Sensor
- Base and support rod
- Reflector board (optional)

PURPOSE

The purpose of this activity is to introduce the relationships between the motion of an object - YOU - and a Graph of position and time for the moving object.

NOTE: This activity is easier to do if you have a partner to run the computer while you move.

THEORY

When describing the motion of an object, knowing where it is relative to a reference point, how fast and in what direction it is moving, and how it is accelerating (changing its rate of motion) is essential. A sonar ranging device such as the Motion Sensor uses pulses of ultrasound that reflect from an object to determine the position of the object. As the object moves, the change in its position is measured many times each second. The change in position from moment to moment is expressed as a velocity (meters per second). The change in velocity from moment to moment is expressed as an acceleration (meters per second per second). The position of an object at a particular time can be plotted on a graph. You can also graph the velocity and acceleration of the object versus time. A graph is a mathematical picture of the motion of an object. For this reason, it is important to understand how to interpret a graph of position, velocity, or acceleration versus time. In this activity you will plot a graph in real-time, that is, as the motion is happening.

PROCEDURE

For this activity, you will be the object in motion. The Motion Sensor will measure your position as you move in a straight line at different speeds. The Capstone program will plot your motion on a graph of position and time.

Computer Setup

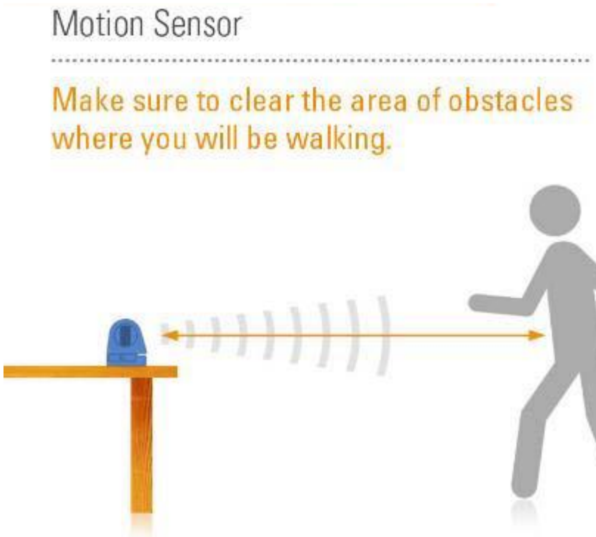
1. Plug the *USB Link* into the computer's USB port.
2. Plug the *PASPORT Motion Sensor* into the USB Link. This will automatically launch the Capstone window.

The motion detector is presented here as a "black box". There is no need to explain its working to students. It is enough that students are convinced that the motion detector does what it is expected to do — that is, that it specifies the position of an object at different times. This is goal of the first approach.

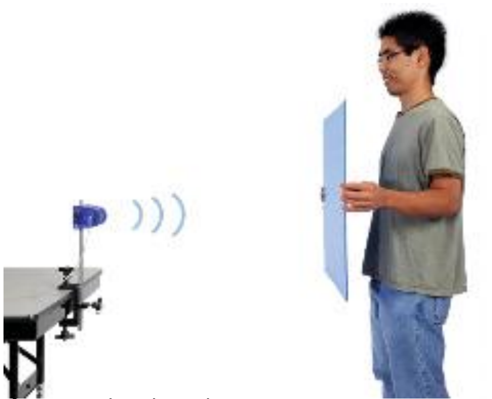
Warning: You will be moving backward, so be certain that the area behind you is free of obstacle because the distance is calculated from the first object that reflects the ultrasounds.

Equipment Setup

- Mount the Motion Sensor on a support rod so that it is aimed at your midsection when you are standing in front of the sensor. Make sure that you can move at least 2 meters away from the Motion Sensor.
- Position the computer monitor so you can see the screen while you move away from the Motion Sensor



Sc e
closest object. If you swing your arms or take large steps, the detector will at one moment see an arm, at another a leg. This motion causes an erratic position-time graph. The best position-time graphs are obtained by clasping a reflector board in front of you and moving with short, shuffling steps of about 5 cm each.



What are we going to measure?

All the points that are on a circumference that has the sensor as its center are at the same distance from the sensor itself, although it is possible to collect measurements only of those points that are located within the cone "visible" to the sensor. Furthermore, it should be noted that what is detected is the **distance** and not the **position**, ie the sensor does not say exactly where the object is but only how far it is. To verify this statement, let's do a simple test.

We place your partner in front of the sensor about 2 meters away and make a position-time graph; obviously, the result is a line parallel to the time axis, since, if your partner has been reasonably still, its distance from the sensor has not changed.

Small variations are attributable to the fact that your partner, despite his best effort, cannot be perfectly still because he is an animated being and therefore he breathes.

Don't forget to stop recording, otherwise your computer's memory will be filled unnecessarily.

Now let's try to make a similar graph, but making our lab rat move along the line drawn on the ground on which before it had been stopped, that is, to make it clear, the line that highlights the 2 meters. Obviously, the movement has to take place inside the detection cone of the sensor.

We can see that this second graph is extremely similar to the first one despite, in this case, your partner has changed his position several times; this is because he did not change his distance from the sensor since he walked along the line whose points were equidistant from the sensor itself. We can therefore conclude that what is measured is the distance of the object and not its position. But the motions that will be observed and studied will almost all be in one direction only and therefore, in this hypothesis, the two concepts of **distance and position are the same**, from this point on we will simply speak of position.

Investigation A

Observe, while recording, the motion of your partner while, after standing still for the first two seconds, he moves away from the sensor slowly and with the most regular gait as possible. Don't forget to stop recording. As your partner moves, the computer creates a graph that represents how its position from the sensor changes as time passes.

Investigation B

Now look at what happens if a person repeats experience A increasing the gait. Let's make the computer superimpose the two graphs; what do you notice?

Prediction

What's your prediction about the position-time graph when your partner moves towards the sensor with a regular and slow gait? Draw with a pencil in the apposite space below the graph you predict:

GRAPH C

Investigation C

Let's make now a verification of your prediction by carrying out the motion. As in the previous cases, a person will stand still for a couple of seconds and then leave. Bring the newly computer-generated graph to the same space where you just sketched your prediction (graph C). Was your prediction correct**?**

Prediction

How do you think the position-time graph will be, repeating experience C but with a more sustained gait? Draw your prediction below.

GRAPH D

Investigation D

We create the graph using the computer. Place it together with your prediction and don't forget to note the characteristics of the motion. Let's ask now the computer to superimpose the two graphs C and D. What do you notice?

THEORY

Remember that in the previous graphs the **position** and the **instant** are to be understood in the definition we give to them in physics: the *position* of a body is a point in space, ie we *model* the body as a point, the *instant* is a point on the axis of time and it is NOT an interval, as in everyday life! The position and the corresponding reading of the instant on a clock (eg the one in the computer) are closely connected and we call this combination "***event***".

The change of position from a point s1 to another point s_2 is called **displacement** Δs, where $\Delta s = s_2 - s_1$ (the symbol Δ represents the change in value of a quantity: final value minus initial value).

When numbers are inserted as position values, the movement in the positive direction (away from the sensor) is always a positive sign, and a movement in the opposite direction (approaching the sensor) will always be represented by a negative sign number.

The displacement is an example of a **vectorial quantity**, that is a quantity which is characterized by a *direction*(angle), with its *sense*(orientation), and by a *magnitude*. In the displacement, its *magnitude* (ie the number of meters) is the *distance* between the *initial* and *final position*, the sense of its direction, from the initial position to the final position, *is represented by a positive or negative sign*.

In previous experiences you have been invited to move with a slow and sustained gait. Analyzing the respective position-time graphs, it has been observed that different speeds are linked to different slopes of the straight sections in the position-time graph. The greater the speed, the greater the slope of the path in the graph.

Another clear information that can be deduced from the graphs is the negative slope obtained in approaching motions. Remembering what we have studied with the straight line in the Cartesian plane, let's think about what calculation we can perform with the numbers ***s*** and ***t*** to obtain direct information on how quickly the person (a body in general) is moving.

By ***average velocity*** ***in a given time interval we mean the displacement divided by the time interval in which the movement took place:***

$$v_m = \frac{\Delta s}{\Delta t} = \frac{s_2 - s_1}{t_2 - t_1}$$

Apply this formula with the data in your possession from the previous activities recorded in the file and calculate the average velocity in some time intervals of the different motions. Think about how these values vary.

You can see that the value is great when the body moves quickly; the value is small when the body moves slowly; the algebraic sign indicates the direction of motion, and so on.

Warning: often in our everyday language we mean by velocity a quantity that is not the one previously defined (*velocity vector*), but rather the *average scalar speed* (the gait).

While the average vector velocity concerns the displacement of a body, this other type of average velocity considers the total length actually traveled (for example the number of meters traveled), regardless of direction.

That is V_S = (*total path length*) / Δt.

The average scalar velocity differs from the average vector velocity also because it does NOT include the sense(orientation) and therefore lacks an algebraic sign.

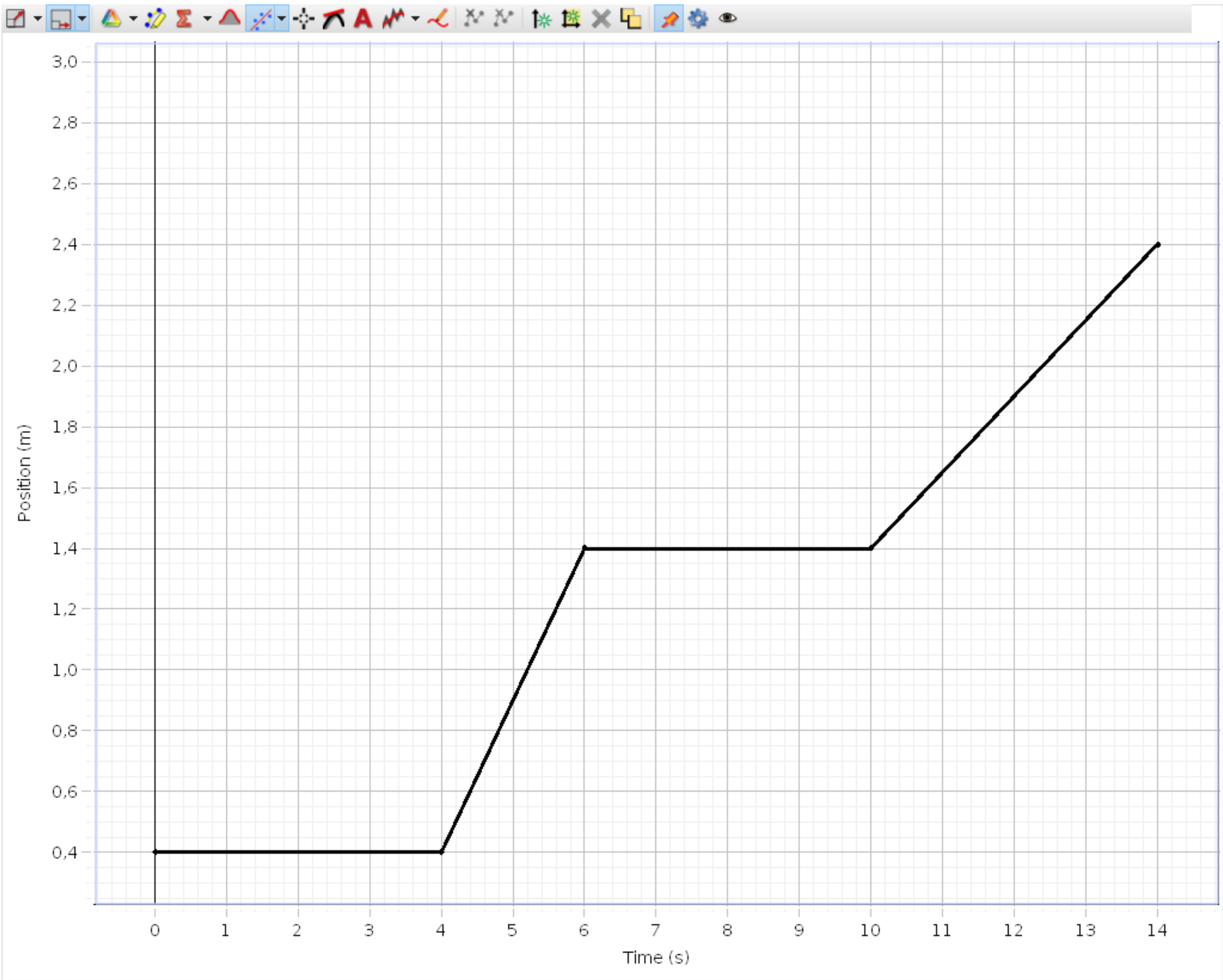
These differences lead to different results depending on the concept of speed considered. Eg in the forward and backward motion in front of the sensor, when you return to the same starting position, you have a zero vector velocity because Δs = 0 !!

In Anglo-Saxon countries the different use of the concept of speed is solved with two different words: "*speed*" for the scalar speed (the gait of our motions) and "*velocity*" for vector velocity.

Match the Graphs

A competition among groups

Let's form now small groups with your companions. Analyze, together with your group mates, the graph below:



Prediction

How should your partner move to get a graph that overlaps the one above? Do a group prediction.

In the first four seconds the person must ______________________________ between the fourth and sixth second must ...

Let's verify empirically the group predictions. Each group will now designate a representative who will:

1. Move according to the indications developed by its group.

2. Try to obtain, in any possible way, the previous graph, that will now appear on the computer screen.

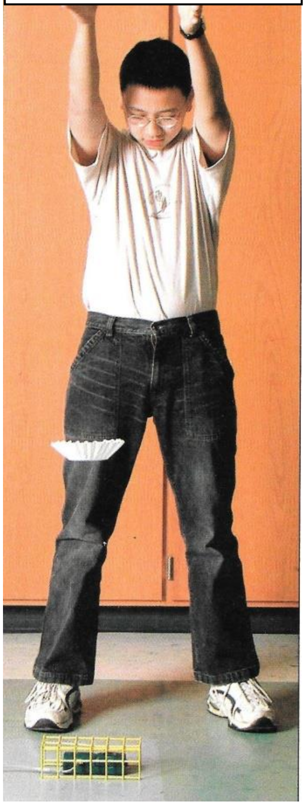
Were certain parts of the plots easier to match than other parts? Why or why not?

EXPERIMENT - TERMINAL SPEED (Uri Haber-Schaim et al, "Force, Motion, and Energy", Science Curriculum Inc. USA, 2002)

When you ride a bicycle starting from rest, you very quickly reach a speed that you want to keep constant for most of the ride. For cars, trains, airplanes, and so on, this speed is called the *cruising speed*. Does a "cruising speed" occur only through human control, or does it also happen in nature?

To find out, you will investigate how a coffee filter falls to the floor after being released. In deciding from which position to release the filter, take two considerations into account. You want the filter to fall over the longest possible distance, but you also want the detector to "see" only filter, not you as well.

A coffe filter shortly after being released. Note the position of the student's arms and the position of the motion detector.



- How should you hold the filter to achieve a reasonable compromise between these two considerations?
- After releasing the filter, why should you keep your arms up until the filter reaches the floor?

With your partner, try a run or two to see how the graph look. The motion detector only registers the location of objects at a distance greater than 15 cm.

- How does the height-versus-time graph show that the motion detector loses sight of the filter when it is closer than 15 cm?

Change the time axis so that only the part where the motion of the filter has been recorded is visible.

- In what time interval did the coffee filter fall at constant speed? How did you reach your conclusion?

The constant speed reached by a falling body is called the terminal speed.

- What was the terminal speed of the coffee filter?

THEORY

There are only two forces acting on the filter once you release it — the gravitational force and the friction with the air. When you released the filter, the force of friction exerted by the air must have been less than the gravitational force. It the force of friction had been as strong as the gravitational force, the filter would have stayed suspended in midair. It did not. The filter fell and reached a terminal speed in a very short time. This tells us that the frictional force exerted by the air increased as the speed of the filter increased until it balanced the gravitational force. Generally, the force of friction increases with speed as a solid object moves through a gas or a liquid.

Figure is a graph of height versus time for a falling coffee filter. The motion detector registered the height every tenth of a second. From the graph, find the time at which the filter reached terminal speed.

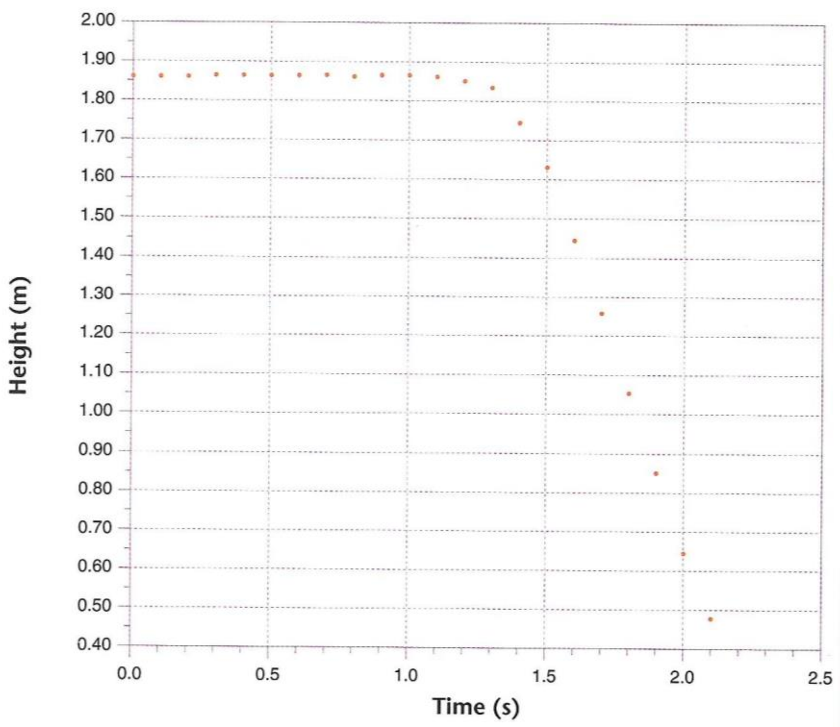


Figure is a graph of height versus time for a falling coffee filter. The motion detector registered the height every tenth of a second. From the graph, find the time at which the filter reached terminal speed.

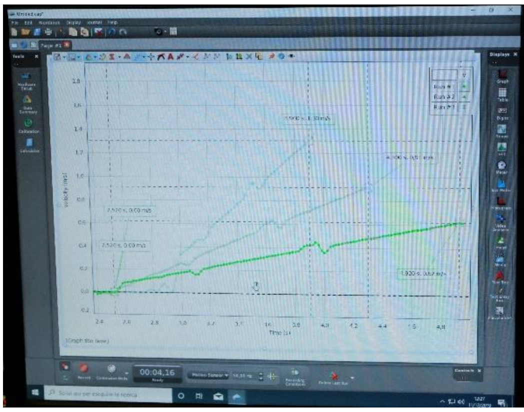
Topic	Age	Country	Date
Parabolic Motion (Pasport)	>14	Italy	Dec 2019

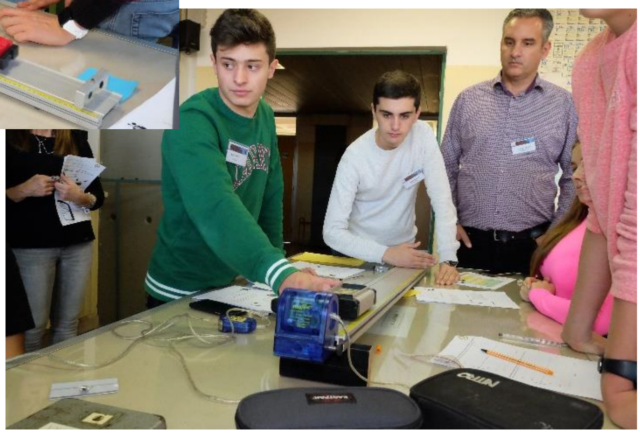
Parabolic motion

TEAM n° ____Students: ____________________________________

Aim: to determine the range and the maximum height that a body fired by a cannon reaches along its parabolic trajectory.

Equipment: Ballistic cart, smartphone, Pasco Capstone software.





Instructions:

1) Prepare equipment as in the picture:
2) Launch the ballistic cart and shoot a video of what happens, if possible, in high definition
3) Send the video to the PC to analyze the motion

Theoretical outline

Parabolic motion is a motion in 2 dimensions, the results from the composition of two simultaneous and independent one-dimensional motions, along perpendicular directions:

- uniform rectilinear motion
- uniform accelerated rectilinear motion

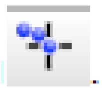
It can be described by means of the kinematic relationships between vector quantities position, velocity and acceleration.

Enter Analysis Mode

To analyze a movie clip, you will need to define the horizontal and vertical axis. The axis is movable by click-dragging the yellow circle at the origin. Grabbing the yellow circles on the axis can also rotate the coordinate system. You will also need something in the movie frame of known length. The calibration tool can be moved and scaled to define the movie scale. The setting of these two tools will give us the proper values in our analysis.

The default movie controls are for video are for play and data sync mode.

Video analysis mode is already enabled. You access it by clicking on the button. (Note that the toolbar changes to a new set of tools).

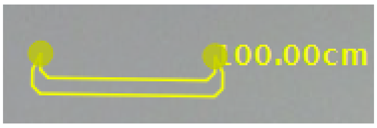


1. Using the playback controls (Experiment Control bar at the bottom of the page) advance the movie to the frame where the ball is coming out of the ballistic cart launcher.

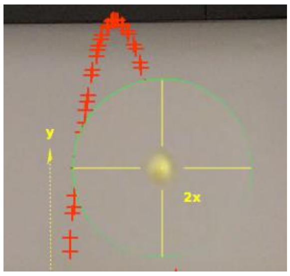
2. Click in the movie frame to make it active.

3. Move the origin of the XY axis tool to the middle of the PASCO cart (where the white marker is – it is probably partially hidden by the bracket that launches the projectile out of the cart). Once positioned, you can rotate the axis a little to align with the track.

4. Click on the Calibration Tool and move its ends to end-stops on the track. The distance between the outside of both end stops is 120 cm (this distance has been pre-set for you in properties and under "Calibration Tool" - "Real World Length").



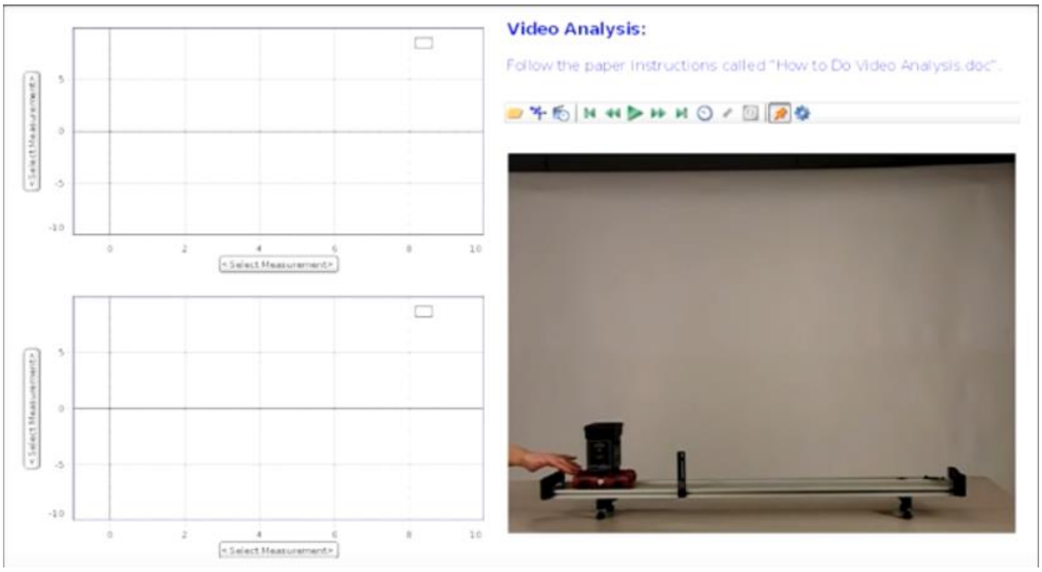
5. You can start clicking the ball to track it – but using the magnifier tool will yield more accurate data. Activate the magnifier tool (in the movie toolbar). Now you can see the projectile positions better and target a point on the ball consistently with each frame.



Graphs:

Obtain the following graphs:

a. v_x (x-component of velocity) vs. time graph. Find the average horizontal velocity of the cart
b. v_y (y-component of velocity) vs. time graph.
c. Superimpose the two graphs and print the result.



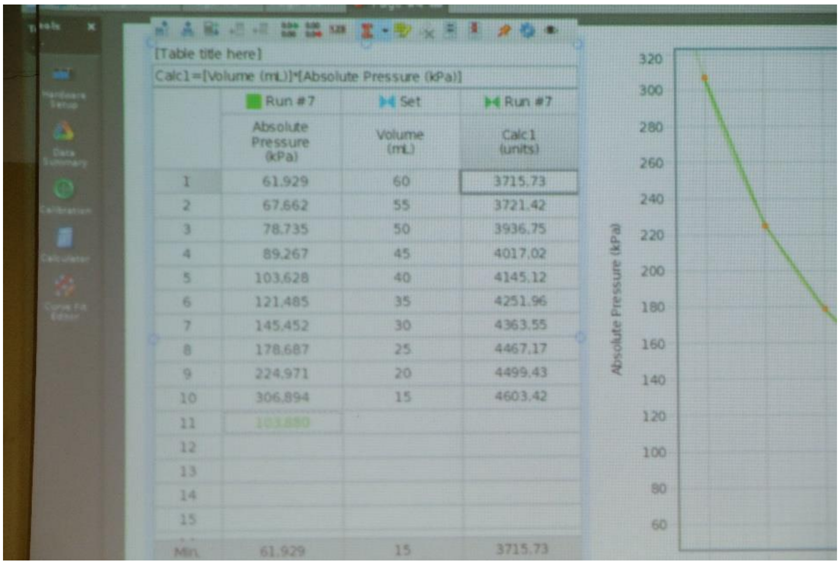
Answer the following questions:

1) How does the x-component (horizontal) of the velocity of the ball vary?
2) How does the y-component (vertical) of the velocity of the ball vary?
3) Get the software to display the graph of y-component of the acceleration vs. time and calculate its average value. Is this value close to a?
4) Can you infer more information from other graphs?

Topic Pressure (Capstone)	Age >14	Country Italy	Date Dec 2019

Rising pressure is created in an empty syringe and registered by a pressure sensor.

The rising rates of pressure are visualized by the program.



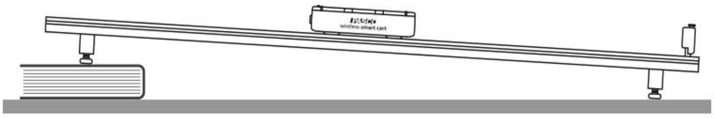
Topic	Age	Country	Date
Motion with constant acceleration (Pasco)	>14	Italy	Dec 2019

EXPERIMENT: Motion with constant acceleration

Team n° ____Students: ____

Aims: to define acceleration and discuss its sign. Relate speed vs. time graphs with acceleration vs. time graphs. Identify accelerated and decelerated motion in a speed vs. time graph.

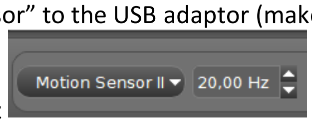
Equipment configuration: on the left configuration with the sonar sensor; on the right with the Bluetooth cart.



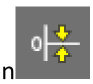
Instructions for sonar sensor, data acquisition software and cart operation

1) Launch the CAPSTONE software (icon on the PC desktop)  .

2) You will need aluminium rail, Pasco ultrasound sonar “motion sensor”  , Pasco USB adaptor  .

3) Connect the USB adaptor to the PC (via USB); connect the sonar “motion sensor” to the USB adaptor (make sure that the connector faces the right way). Set a sampling rate of at least 50Hz 

4) Set the cart at its start position, making sure that it is, at least, 10 cm away from the “motion sensor” (below such distance, position measurements are inaccurate).

5) Set the starting point (origin of position axis) using the position reset button .

6) Start data acquisition 

7) Stop the cart (by hand) before it reaches the end of the rail. Select and delete position data that are acquired after the cart has stopped.

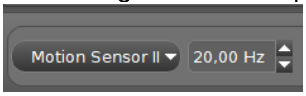
Instructions for Bluetooth cart:

1) Launch the CAPSTONE software (icon on the desktop of laptop PC) 

.

2) You will need aluminium rail, Pasco Bluetooth cart.  .

3) Turn the Bluetooth cart on, using the switch on its side, and connect the cart to the data acquisition software. To achieve this, click on the cart icon, making sure that characteristics and codes shown in the software agree with those printed on the cart. Set a sampling rate of at least 50Hz (for position measurements)

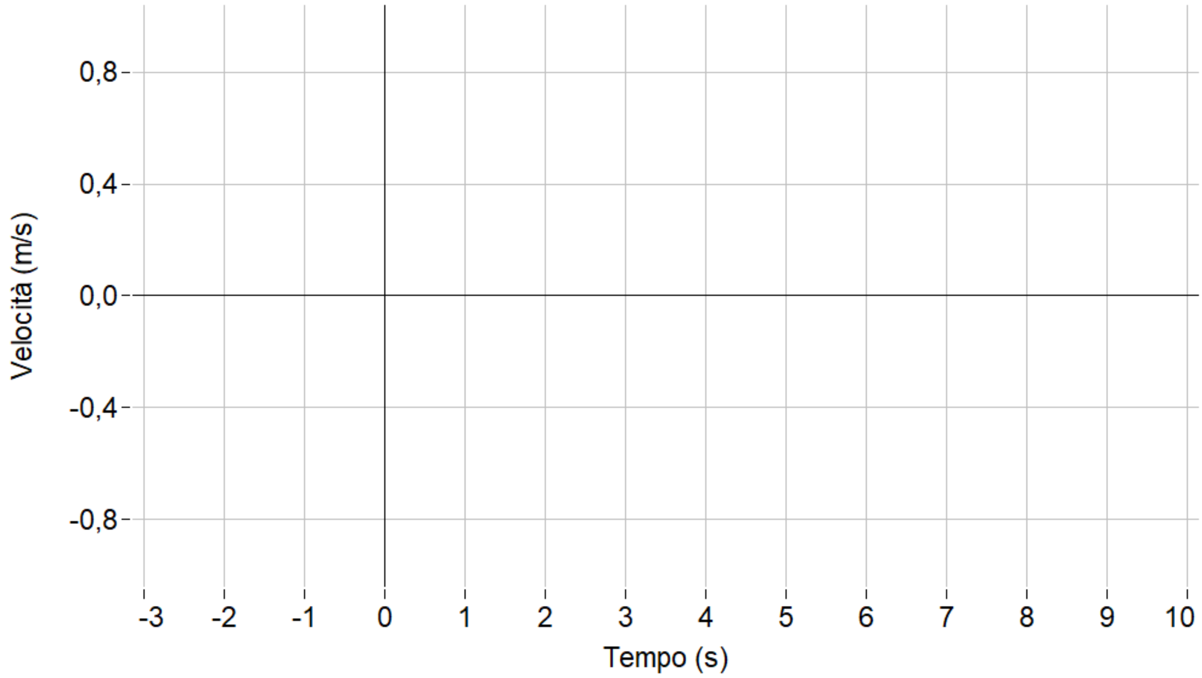


4) Set the cart on its start position (top of the rail)

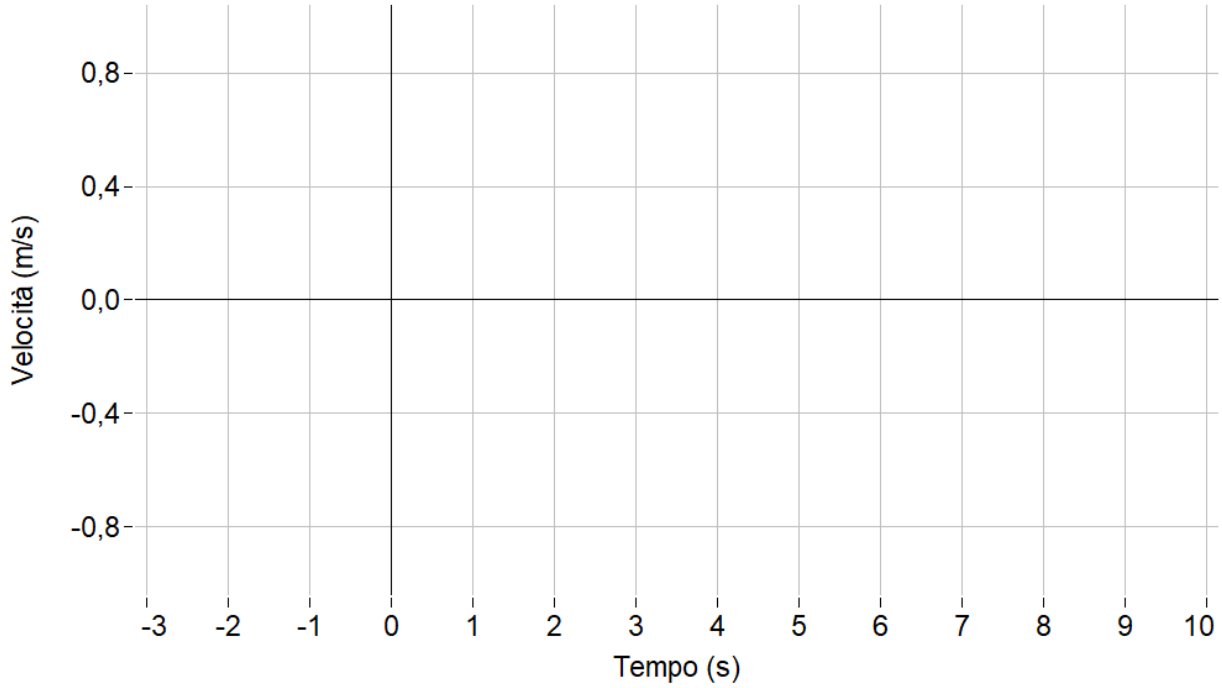
5) Check the direction: the cart should move towards the positive direction of the x-axis printed on it.

6) Stop the cart (by hand) before it reaches the end of the rail. Select and delete position data that are acquired after the cart has stopped.

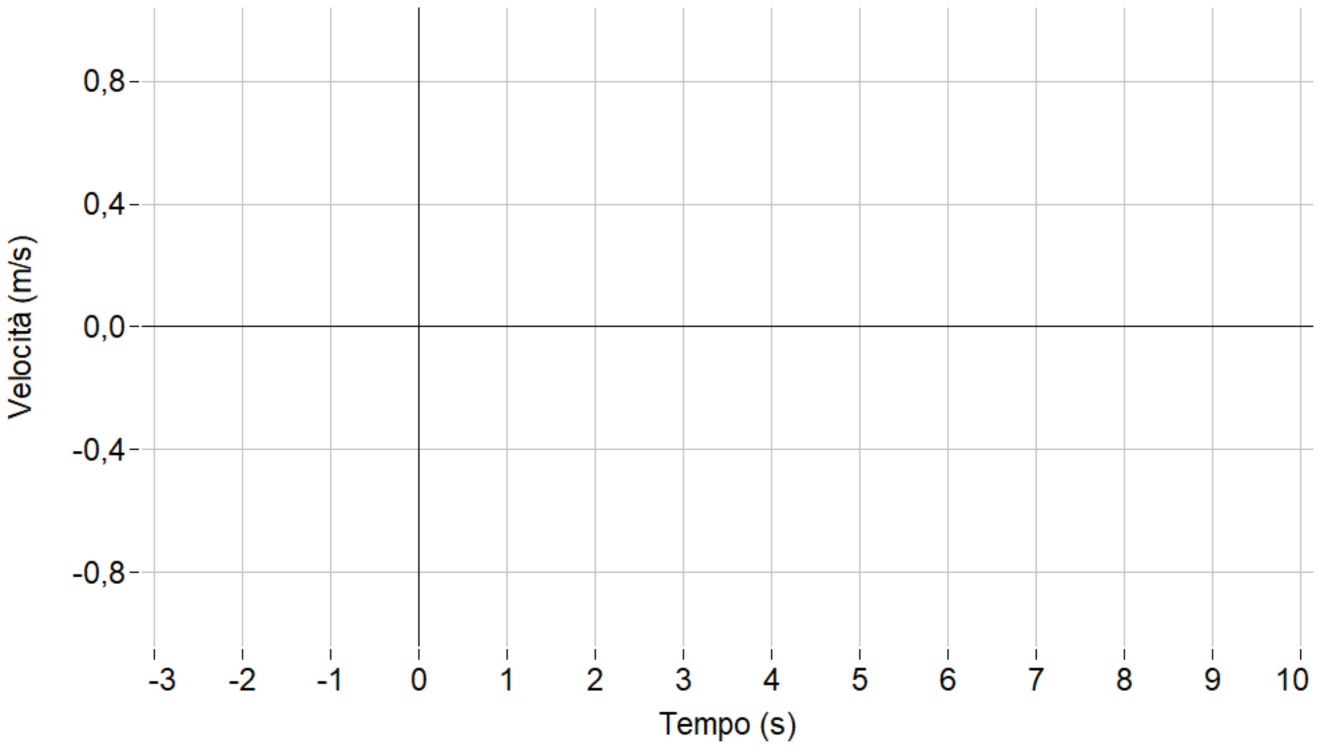
Experiment (a): what will the velocity vs. time graph look like, if the cart is held still for a couple of seconds and then released? Draw below, in different colours, both your expectation and the experimental results.



Experiment (b): the same as above, but now setting the rail at a steeper slope. Draw below, in different colours, both your expectation and the experimental results.



DRAW BOTH GRAPHS FROM EXPERIMENTS 1 AND 2 (use different colours)



Answer the following questions

1) Can you tell, by just looking at the graphs, in which case the rate of change of velocity is higher?

____

____

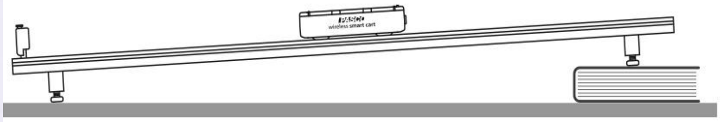
2) Calculate the slopes of the straight lines in the two graphs, after filling in the two tables below. To find velocity data you can use the “smart tool”[1] in CapStone.

TEST 1	v_1(m/s)	v_2(m/s)	t_1(s)	t_2(s)	a(m/s^2)

TEST 2	v_1(m/S)	v_2(m/s)	t_1(s)	t_2(s)	a(m/s^2)

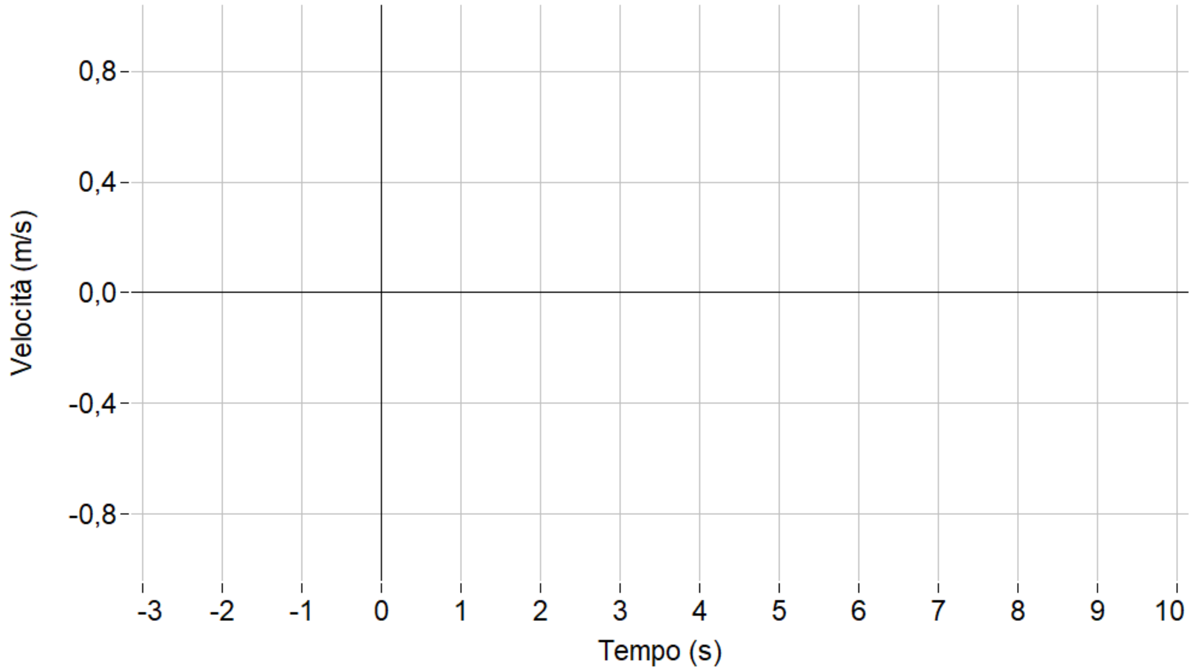
EXPERIMENT **(c):** Now we will change the “motion sensor” position, as in the picture below.

Using the Bluetooth cart, set the book below the rail, in such a way as to have an opposite slope (with respect to Experiments 3a and 3b).



What will the velocity vs. time graph look like now, if the cart is held still for a couple of seconds, and then released?

[1] One of the buttons in CapStone software, just above the graph.



Perform the experiment and compute the acceleration:

TEST 3	v_1(m/s)	v_2(m/s)	t_1(s)	t_2(s)	a(m/s^2)

Compare now the graph from this last activity to the former ones. You can see that the same falling motion of the cart along the rail (with the same slope) has two different graphical representations. The reason is that a new frame of reference was chosen, by moving the sonar sensor to a new position.

Conclusions:

____

____

____

____

Topic Motion Acceleration measurement	Age >14	Country Poland	Date v.1 December 2019 / v.2 September 2021

**The original version of the experiment (v.1) was developed using the Arduino IDE. With the advent of the Raspberry Pi Pico microcontroller and its great possibilities, the description of the experiment has been adapted to MicroPython (v.2), which will make it much easier for students to perform it.*

Function, realisation

Construction of a measuring instrument based on microcontroller and Python that measures the light intensity.

Hardware required

- Microcontroller Raspberry Pi Pico or Raspberry Pi Pico W with Micropython 1.19 or above
- Cytron Maker Pico docking station for Pico
- ADXL345 module (accelerometer)
- 3V3 LCD 1602 I2C display from Seeedstudio with Grove socket,
- PC or Mac computer.

Materials required

- 1 standard Grove wire, 1xGrove female wire
- micro usb 2.0 high speed cable(30 cm or longer)
- powerbank (for standalone version)
- paper for note

Software required

- Thonny App (thonny.org)
- library for ADXL sensor: https://github.com/DFRobot/micropython-dflib/blob/master/ADXL345/user_lib/ADXL345.py
- library for LCD: https://files.seeedstudio.com/wiki/Grove_Shield_for_Pi_Pico_V1.0/Libraries.rar

Using Thonny App You can save libraries inside lib folder in Pico memory.

Topic-Motion Optical photo gate – movement and speed measurement	Age >14	Country Poland	Date v1. December 2019 v.2 September 2021

*The original version of the experiment (v.1) was developed using the Arduino IDE. With the advent of the Raspberry Pi Pico microcontroller and its great possibilities, the description of the experiment has been adapted to MicroPython (v.2), which will make it much easier for students to perform it.

Function, realisation

Build experimental system based which can measures speed and movement.

Hardware required

- Microcontroller Raspberry Pi Pico with Micropython 1.18 or above
- Cytron Maker Pico docking station for Pico (or different docking station for Pico: Waveshare, Seeedstudio),
- 1x Seeed studio Screw Terminal (or available on the market simple screw terminal for wire connection),
- IR beam interruption sensor - LED 5mm (or 3mm LED)- 0-25 cm, illumination angle of **20°** (In our case, the sensor is embedded in a self-made wooden housing or in a housing printed on a 3D printer)
- 3V3 LCD 1602 I2C display Seeedstudio with Grove Socket,
- PC or Mac computer.

Materials required

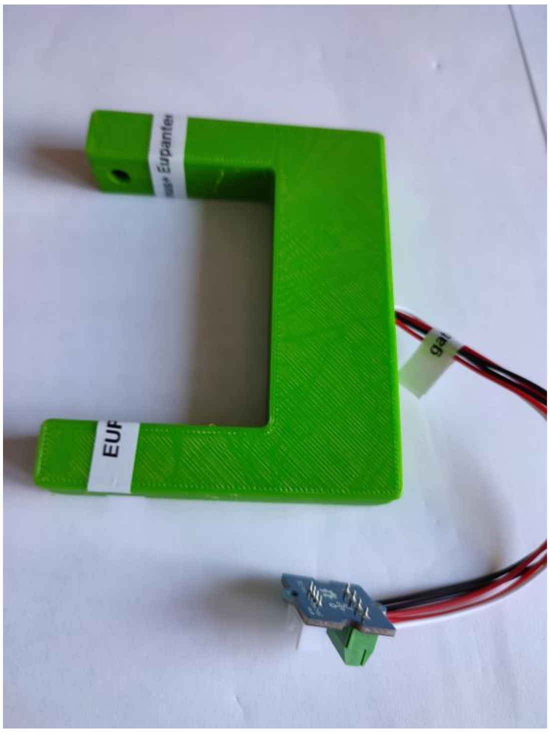
- 2x Grove wires
- micro usb 2.0 high speed (15cm or 30 cm)
- paper for note

Software required

- Thonny (thonny.org)
- llibrary for LCD1602:
https://files.seeedstudio.com/wiki/Grove_Shield_for_Pi_Pico_V1.0/Libraries.rar
- script: gate_time_pullup1lcd

Setup Hardware

Connections have been simplified thanks to the docking station and Grove cables. It is much harder to make a mistake or damage.



In above illustration: on the left IR beam sensor, on the right sensor inside 3D printed housing and Grove screw terminal

IR Beam sensor (photo gate) connection with Raspberry Pi Pico:

Gate (IR Beam)	Raspberry Pi Pico pins	Grove
Signal	GP9	yellow
not connected	GP8	white
power (+)	3V3	red

ground (-)	GND	black

LCD I2C screen connection with Raspberry Pi Pico:

LCD Pins	Raspberry Pi Pico pins	Grove wire
SCL	GP7	yellow
SDA	GP6	white
VCC	3V3	red
GND	GND	black

Setup software

Script:
MicroPython script measures how long the beam was interrupted and gives the result in milliseconds. After understanding the code and making small modifications, you can use the motion measurement system that you have built in other applications.

Please note in the script the **pull_up** option for the pin where the gate is connected.

Timing is performed by instructions:

```
timer_start = utime.ticks_ms()
period_time = utime.ticks_diff (utime.ticks_ms(), timer_start)
```

where
utime.ticks_ms() them measures the number of milliseconds since the script was run.
The second instruction computes the difference between the two timestamps.

Low state (0) of the beam pin means "gate broken"
High state (1) means "gate open"

```
while True:
    val=gate.value()
```

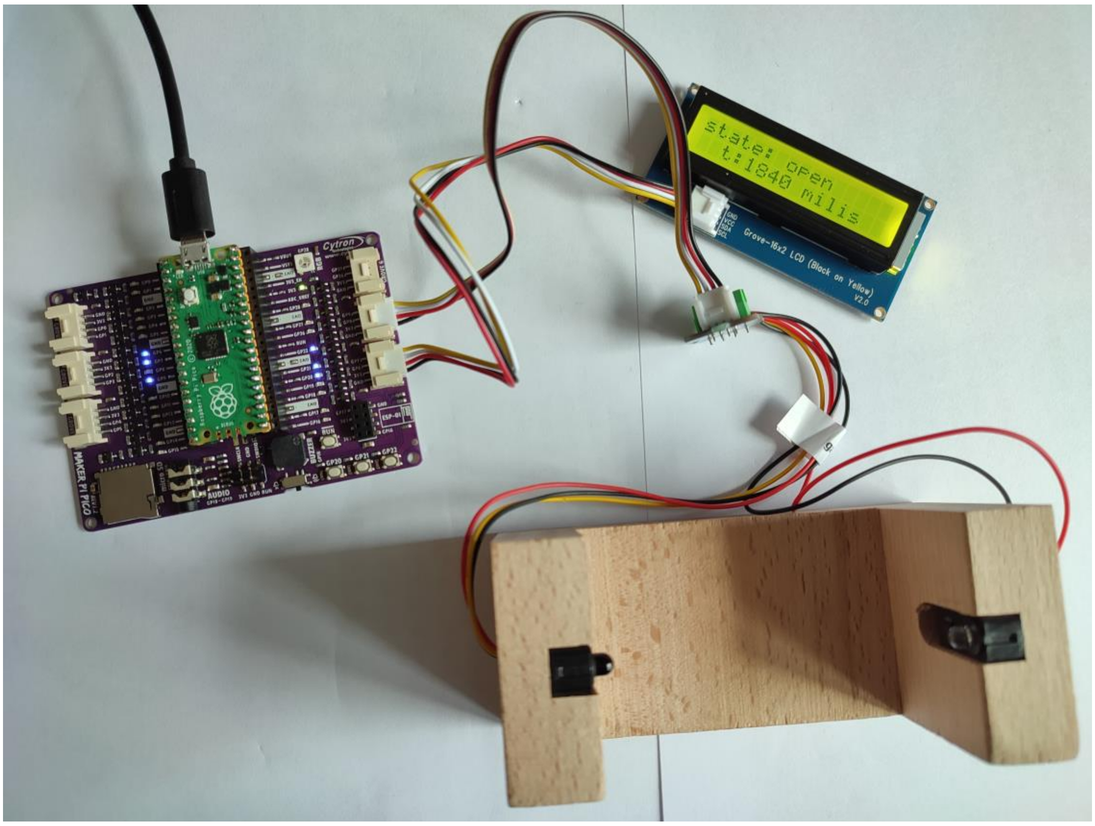
```
from lcd1602 import LCD1602
from machine import I2C,Pin
import utime
# setting I2C bus
i2c = I2C(1,scl=Pin(7), sda=Pin(6), freq=400000)
#LCD connected to GP6, GP7
d = LCD1602(i2c, 2, 16) #initialisation of LCD
d.display()
utime.sleep(1)
d.clear()
d.print('Erasmus+ ')
utime.sleep(1)
d.setCursor(0, 1) #second row(1) selected for print, we count from 0
d.print('Eupantec2019.eu')
utime.sleep(1)
d.clear()
d.setCursor(0, 0)
d.print('Optical Gate')
d.setCursor(0, 1)
d.print('time')
utime.sleep(1)
d.clear()
d.setCursor(0, 0)
#Gate connected with Grove wire to GP8, Gp9
BEAM_PIN=9
print("Beam Ready")
#Important: PULL_UP option must be set
gate=machine.Pin(BEAM_PIN, machine.Pin.IN, machine.Pin.PULL_UP)
val=gate.value()
print("state:",val)
```

```
print("state:",val)
while True:
    val=gate.value()
    if val==0:
        timer_start=utime.ticks_ms()
        while val==0:
                print("gate broken")
                d.setCursor(0,0)
                d.print('state: broken')
                val=gate.value()
                if val==1:
                    #calculating the time
                    period_time=utime.ticks_diff(utime.ticks_ms(), timer_start)
                    d.clear()
                    d.setCursor(2,1)
                    d.print("t:"+str(period_time))
                    d.print(" milis")
                    utime.sleep(0.1)
    else:
        print("state: open")
        d.setCursor(0,0)
        d.print('state: open')
    utime.sleep(0.1)
```

Demonstration of the system operation:

Go further- ideas, testing:

-place an item in the gate for a while, then remove it and read the time



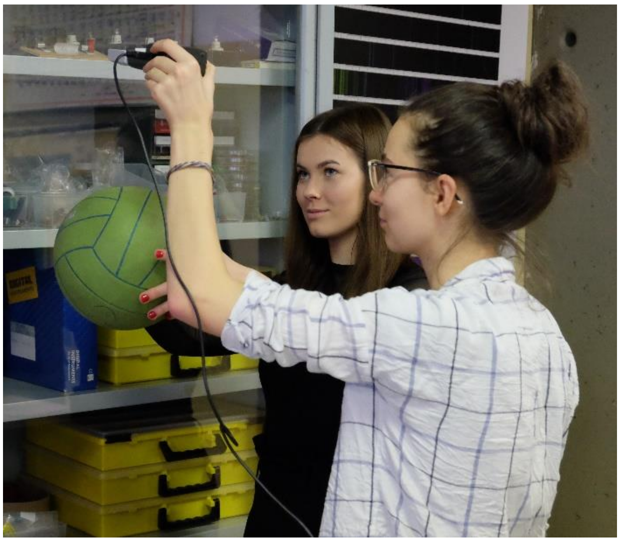
-You can check how the sensor reacts to a glass filled with currant juice
-You can change the code for measure pendulum period

-You can apply 2 photo gates to count speed

-design a system that sends the gate state to a website or via bluetooth to a mobile application on your phone.

Topic	Age	Country	Date
Free fall Measuring *g* Terminal velocity	>14	Portugal	Dec 2019

Does the heaviest object fall first?



In this experiment, we´ll drop several objects with different masses and follow their movements through a position sensor. We'll do it in two parts:

Part I - drop different **sports balls** with significantly different masses;

Part II - drop different numbers of **cup cake** paper forms.

PART I - Sport balls

Drop different sports balls under the sensor and follow their movements with a position sensor.

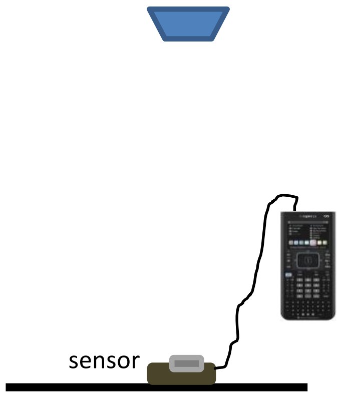
Check the graph of the distance of the ball to the sensor in function of time, in the graphic calculator, and fill in the table of registration on the next page.

<u>Consult the document in the attachments in order to better understand how to use the equipment, if necessary.</u>

Ball details			Initial instant (when starts falling)	Final instant (when the floor is reached)	Duration of fall	Acceleration
sport	mass/g	Diameter (aprox.)/cm	t_i/s	t_f/s	t/s	g/m·s^{-2}

Co-funded by the Erasmus+ Programme of the European Union

PART II – paper forms



Drop a different number off paper cup cake forms directly above the position sensor, from about your shoulder height.

Check the graph in the graphic calculator and fill in the table of registration.

Consult the document in attachments in order to better understand how to use the equipment, if necessary.

cup cakes			Initial instant (when starts falling)	Final instant (when the floor is reached)	Duration of fall	Terminal velocity
Number of cup cakes	Mass/g	Diameter (aprox.)/cm	t/s	t_f/s	t/s	vt/m·s^{-1}

Questions:

Part I

1.1- In the sports balls' falling movement, has the time of fall significantly changed when the mass of the balls was varied?

(further exploration: percentaged, how much as the masses varied ? And the duration of fall?)

1.2- Has the mass off the falling ball affected the acceleration of the motion?

(further exploration: percentaged, how much as the masses varied? And the acceleration?)

1.3- "An object falls with a constant acceleration of 10 $m \cdot s^{-2}$". What does this mean?

Part II

2.1- Draw an outline of the duration of fall *vs* number of forms graph.

2.2- Draw an outline of terminal velocity vs number of forms graph.

2.3- In the cupcake falling movement, has the duration of fall significantly changed has the number and mass varied?

2.4- What is the relationship between the mass of the objects dropped (number of cup cake forms) and the terminal velocity acquired?

2.5- What would it be the final velocity of fall if f there were 10 filters falling?

Global Questions

3.1 – In which situation - sports balls or in the cup cake forms – has the air represented a significant obstacle in the motion?

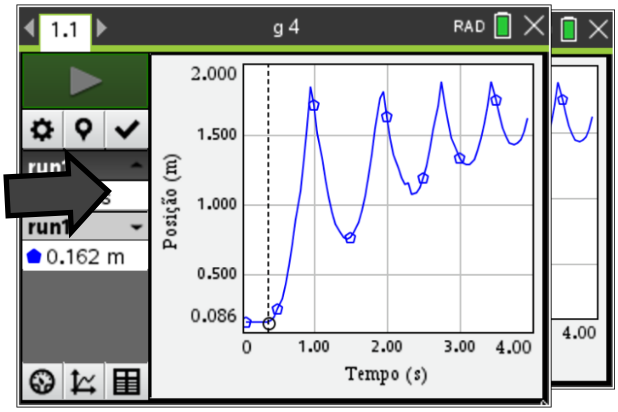
3.2 – Does the heaviest object fall the first?

Attachment 1 - Consulting the graphs for obtaining the times

1. Connect a CBR 2™ to the TI-Nspire™ handheld.
2. Place the ball under the CBR 2.
3. Click the Start button in the DataQuest app to begin sampling.
4. When you hear the CBR 2 begin clicking, drop the ball.
5. When the ball lands on the floor, click the Stop button .
6. On the main screen, select graph view and then select point you need, by moving your cursor.

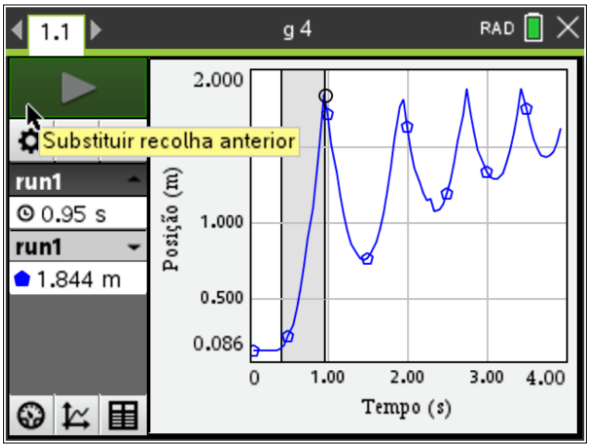
Instant when the ball starts to fall

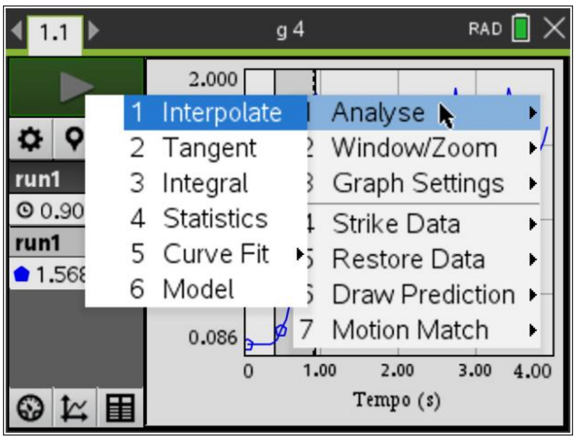
Instant when the ball reaches the floor

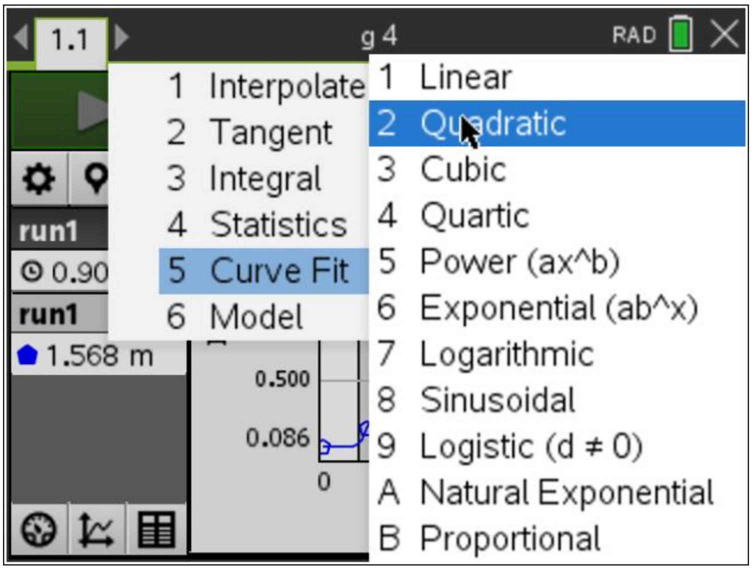


Attachment 2 - Consulting the graphs for obtaining the acceleration

1. On the main screen, select graph view and then select the point you need, and then select the region that represents the ball falling by moving your cursor to the starting point of the drop and clicking to set the left bound. Move the cursor to the right to expand the selection region, and click to select the right bound
2. Analyse the parabolic segment by selecting **Click Menu> 4.analyse> 6.curve fit> 2.quadratic**

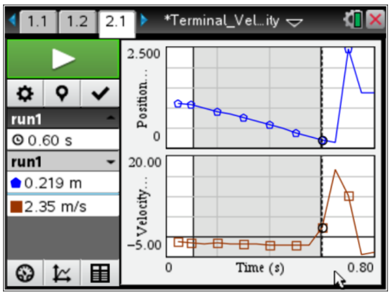






Attachment 3 - Consulting the graphs for obtaining the terminal velocity

1. Connect a CBR 2™ to the TI-Nspire™ handheld.
2. Place the CBR 2 on the floor facing upward.
3. Hold a cupcake paper form directly above the probe at your shoulder height.
4. Click the Start button in the DataQuest app to begin sampling.
5. When you hear the CBR 2 begin clicking, drop the ball.
6. When the ball lands on the floor, click the Stop button .
7. On the main screen, select graph view and then select point you need, and then select the region that represents the coffee filter falling by moving your cursor to the start point of the drop and clicking to set the left bound.
8. Move the cursor to the right to expand the selection region, and click to select the right bound.



9. Analyse the linear segment by selecting **MENU > 4.Analyze > 6.Curve Fit > 1.Linear** and finding the velocity of the single cupcake paper form.