Speed and Velocity

Objectives:

- Measure the position of an object by using the Time-of-Flight VL53L0X Motion Sensor connected at the Raspberry Pi;
- Use the PhyPiDAQ-Software to display position versus time in real-time for an object moving at a constant speed.
- Compare the speed of an object moving at a constant speed to the slope of the line on a position-time graph.
- Compare and contrast the position versus time diagrams for three different types of motion:
  - moving in the positive direction versus moving in the negative direction
  - moving fast versus moving slow
  - a constant speed motion versus a gradually changing speed
- Describe the rules of transposing a position-time graph into a velocity-time graph (and vice versa).
- Employ spreadsheets like LibreOffice or Excel to analyse the shape and slope of position-time and velocity-time graphs based on the measurements stored in .csv files.
- Introduce the velocity as vector quantity and explain the similarities and differences between speed and velocity.

Materials and Equipment:

Raspberry Pi 3B+ or 4B	Variable Speed Motorized Cart
PhyPiDAQ-Software for Data collection	Reflector of metal, white or light colour, for reflecting the Laser beam towards the sensor's receiver;
VL53L0X Motion Sensor	Jumpers to connect the sensor to the Raspberry Pi
Breadboard	Clamps
Tripod Stand	Rod

Set-up:

Connect the VL53L0X Motion Sensor to the Raspberry Pi according to the circuit shown in Figure 1. The students used the Fritzing open-source program to build this circuit.

Connect monitor, mouse and keyboard to the Raspberry Pi and fasten the breadboard, as shown in the Figure 2 or in the Figure 3. Alternatively, you can use any other materials from your lab to fix the sensor into the right position to run the measurements.

Place the Variable Speed Motorized Cart about 3cm in front of the sensor. Stick the reflector on the side facing the sensor. The sensor's sensing element points down the length of the track. The Laser beam falls perpendicularly on the reflector. According to the datasheet of the VL53L0X Motion Sensor, a light-coloured reflector provides a better quality of the measurements.

A Variable Speed Motorized Cart from the Phywe Company with three-stage switch for forward, off, and reverse motion is used in this experiment. The wheels are geared to operate at the constant speed, which is adjusted through the speed control slider. Therefor this Motorized Cart is used to experience uniform motions with constant speed. Alternatively, one can use other Motorized Cart from other company.

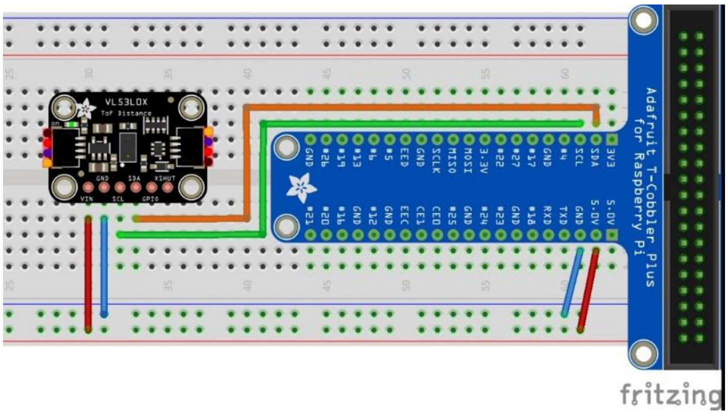

Figure 1 Connecting the VL53L0X Motion Sensor to the Raspberry Pi. The VL53L0X communicate to the Raspberry Pi via I²C-Interface.

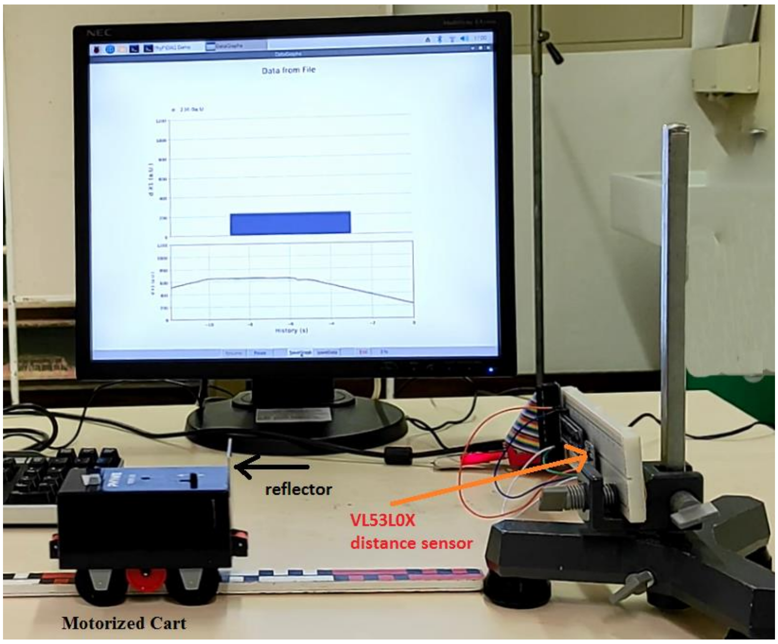

Figure 2 Set-up of the experiment with VL53L0X Motion Sensor and Variable Speed Motorized Cart

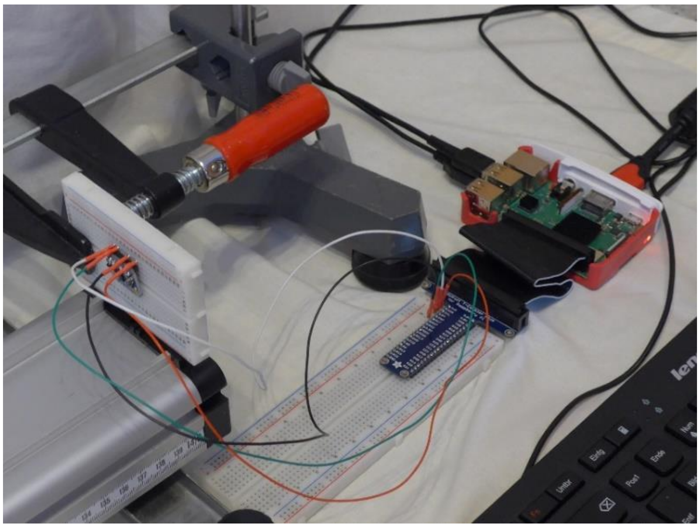

Figure 3 Connecting the VL53L0X Motion Sensor to the Raspberry Pi

Procedure:

In order to visualise the Cart's position in real time one has to go through the following steps. Watch this video https://youtu.be/Vd5KpSkuN2w to get an insight into the procedure of configuring the experiment.

On the Graphical Interface of the PhyPiDAQ Software open the Work directory where the configuration files are going to be saved. In the `DAQ config`: text box introduce the pre-configured file `Distance.daq` from the directory **/home/pi/git/PhyPiDAQ/examples**.

Alternatively, choose the `default.daq` file and edit this according to the VL53L0x_postion_vs_time.daq saved as .pdf file and uploaded in the **Activity-Based Physics** web page**.** To activate the VL53L0x Motion Sensor remove the #-Sign from the command DeviceFile: config/VL53LxConfig.yaml and press the `reload device config` button at the bottom of the `PhyPi Config` window. Click on the `Device Config` Button at the top of the window to change into the Sensor's configuration window. Once there one can edit the .yaml configuration of the VL53L0x Motion Sensor according to the VL53LxConfig.yaml saved as .pdf file and uploaded in the **Activity-Based Physics** web page. Here one can choose between the two sensor types and enter the desired measuring range. Note to click on `EditMode` at the top of `PhyPi Config window` to change or modify the commands in the configuration files.

After all the stages are finished, one can start the measurement by pressing the `StartRun Button` on the main graphical interface of the PhyPiDAQ. The measurement window opens. This video shows the experimental run with the display of position of the Motorized Cart in real-time: https://youtu.be/E0Rt2T9bhUE

Observe the measured Position versus Time in real-time on the measurement window while the Variable Speed Motorized Cart is set to move forwards, stands still at certain distance to the sensor, and then it is set to move backwards.

In order to read position's values at different instants one can activate the `DisplayModule: DataGraphs` in the .daq configuration of the experiment. On the measurement window the bar chart gives the Cart position at a certain instant, as shown in the Figure 4.

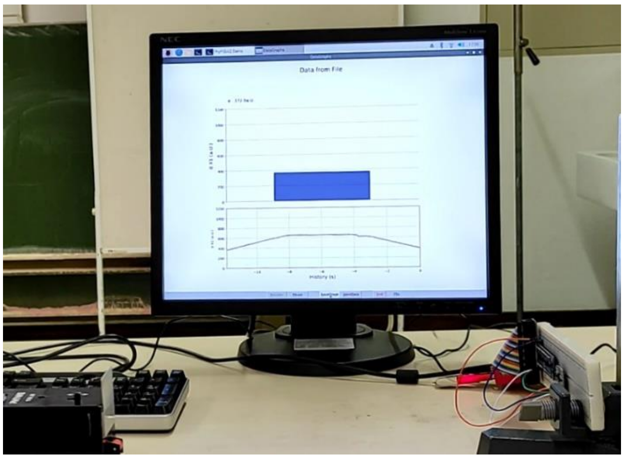

Figure 4. Position vs. Time graph and bar chart for instant positions displayed with the `DataGraphs Module.`

Save the Position vs. Time graph on the measurement window by pressing the `SaveGraph` Button on the taskbar. The picture called DataGraph.png is saved in the same folder containing the. daq and .yaml configuration as well as the recorded measurements in the .csv file.

Press the `End` Button to complete the measurements. To repeat the measurements, choose to don't exist the PhyPiDAQ in the coming out menu window. The same. daq configuration is meant to be used to run again the experiment, for instance by setting the Motorized Cart at a higher speed.

Data Acquisition:

The PhyPiDAQ-Software enables measurements' collection in .csv files. In the Distance.daq configuration one has to activate `DataFile: testfile.csv` as well as `CSVseparator: ';'` by removing the #-Sign. `Testfile` can be renamed to the purpose of the experiment. Add the #-Sign to disable the `DataFile: null` command. On the Raspberry Pi's Terminal the name of the .csv file comes up.

Another option is to press the `SaveData` Button on the task bar of the measurement window, as shown in the Figure 5.

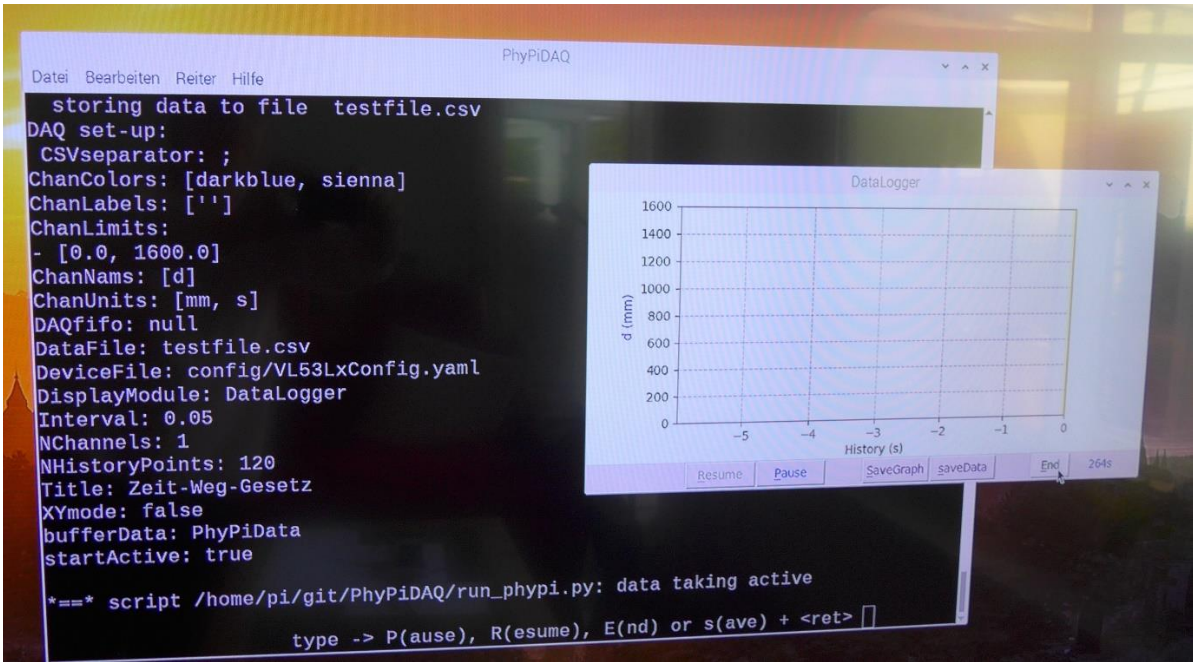

Figure 5. Raspberry Pi's Terminal and PhyPiDAQ measurement window. The recorded data are stored in .csv file. Alternatively, one can save data in .dat file by pressing the `SaveData` *Button on the task bar of the measurement window.*

Data Analysis:

The concept behind this setup is to analyse a combination of motions at different constant speeds and in different directions. The Motorized Cart is set first to move away from the motion sensor. It increases its position relative to the motion sensor, until it is stopped by hand for few seconds and then returns to the starting position.

- Observe the position vs. time graph in real-time and contrast it to the real motion of the Motorized Cart. Make predictions about the changes in the shape and slope of the graph, if the Cart moves faster, slower, in the positive or negative direction.
- Use the position vs. time graph saved as DataGraph.png file by pressing the `SaveGraph` Button in the task bar of the measurement window to determine the average speed and the average velocity for some given time intervals. Assign positive time values on the horizontal axis starting from 0s on the left corner of the coordinate system.
- In the example sown in the Figure 6. annotate the Cart's position vs. time graph with descriptions of motion at each phase of data collection.

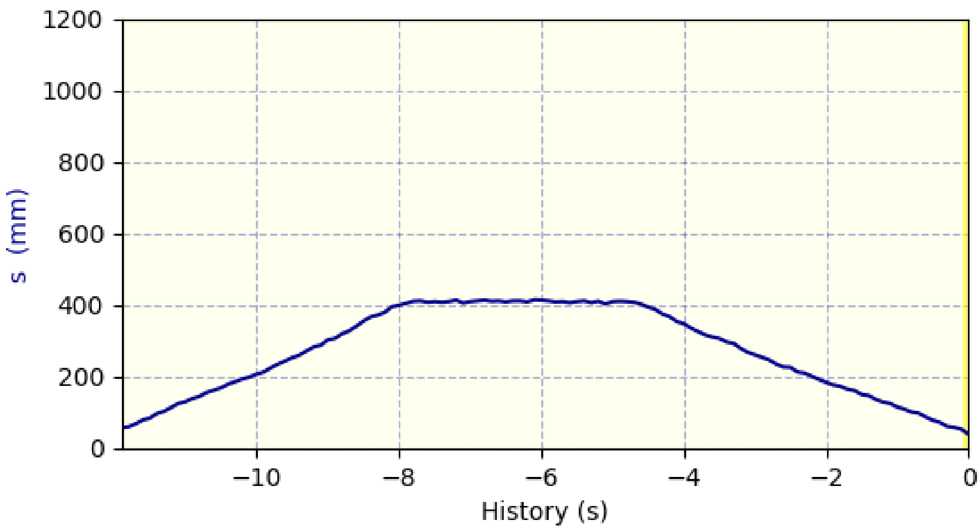

Figure 6. Position vs. Time graph saved as DataGraph.png file in the PhyPiDAQ-measurement window.

- Read the position of the Cart at the given instants and record the position values into the Table.1

Table 1: Position and traveled distance of the Motorized Cart moving back and forth at different instants

Time t(s)	Position s(mm)	Distance d(mm)
0.0		
4.0		
4.4		
7.0		
8.0		
12.0		

- Use the data in Table 1 to calculate the Cart's average speed and the average velocity for the time intervals shown in Table 2. Record your results into Table 2.
- The average velocity is defined as the rate of change of position with time $\bar{v} = \frac{\Delta x}{\Delta t}$, whereas the speed is defined as the change in distance, regardless of the direction of that distance, divided by the change of time it took to travel that distance: $v_{avg} = \frac{\Delta d}{\Delta t}$

Table 2: Average speed and average velocity of the Motorized Cart moving back and forth

Time Interval, **Δt(s)**	**Average Speed** **v_{avg}(m/s)**	**Average Velocity** **$\overline{v}$(m/s)**
0.0 – 4.0		
4.4 – 7.0		
8.0 - 12		
0.0-12.0		
2.0 – 10.0		
2.0 – 6.0		

- Was the Cart's average speed and average velocity the same in any of the time intervals in Table 2?
- How do the values of speed and velocity moving away from the motion sensor compare to the values of speed and velocity moving toward the motion sensor?
- What is that interval, and what about the graphs suggests that average speed and average velocity are the same during it?
- What is that interval, where the average velocity is zero, but the average speed differs?
- Was the Cart's average speed positive but average velocity negative in any of the time intervals in Table 2?
- How can you tell from the position vs. time graphs where the cart's speed was 0 m/s?
- The instantaneous velocity at any moment is the slope of the position versus time graph at that moment. What does the position vs. time graph in the Figure 6. suggests about the average and the instantaneous velocity at each phase of motion? Why are they similar or different?

- Now open the position_vs_time_Motorized_Cart.csv file in Excel or in LibreOffice installed on the Raspberry Pi, that allows a more accurate analysis of the recorded position at different instants. In the Distance.daq configuration the logging interval has been set at 0.1 seconds, meaning that the Cart's position was recorded every 0.1s.

- Use the position values in the position_vs_time_Motorized_Cart.csv file to calculate the velocity over shorter time intervals, such as for Δt=0.1s, during the time intervals listed in the Table.2. Perform the calculation by using the Excel or LibreOffice spreadsheet like in the Figure 7.

	A	B	C	D	E
1	# PhyPiDAQ Data recorder 211005-1456				
2	# logging interval 0.1				
3	# s:(mm)		t(s)	s(mm)	v(mm/s)
4	38		0	38	
5	44		0.1	44	=(D5-D4)/0.1
6	58		0.2	58	
7	61		0.3	61	
8	69		0.4	69	
9	80		0.5	80	

Figure 7. Calculate the Cart's velocity over an 0.1 s time interval, by using the position values collected in .csv file.

- How does the cart's instantaneous velocity at different instants compare to its average velocity during the 0.0 – 3.6 s time interval? Answer the same question for the 4.4 – 7.0s and for the 8.0 - 12s time interval.
- Identify some instants during the 0.0 – 3.6 s time interval, where the Cart's instantaneous velocity equals its average velocity.

What is that interval, where the Cart has positive position and negative instantaneous velocity?

- In Excel or LibreOffice one can get the slope of the position versus time graph by selecting the Trendline in the scatter chart of the measured data. In the Figure 8. the best fit line and formula are displayed for the Cart moving forward and in reverse.

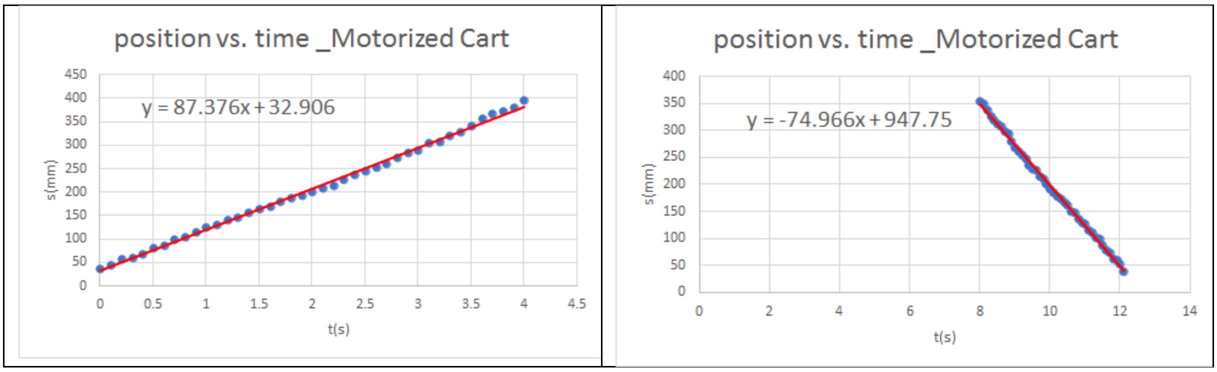

Figure 8. Best Fit Line and Formula for the Cart's motion forward and in reverse.

- For the 0.0-4.0s time interval, compare the slope of the line for moving away from the sensor to the average velocity calculate in the Table 2. Interpret the similarities and the differences.
- What is the physical meaning of the y-intercept on the graph?
- Repeat the procedure for the line moving in reverse.
- How one can determine the position of an object at any later time outside of the plotted time interval, if you know that that object moves at a certain constant velocity?
- Deduce the equation of motion at constant velocity by solving the velocity equation for the displacement and then for the final position. Compare the general formula $s = s_0 + v(t - t_0)$ to the best fit line equation for the forward and in reverse motion in the Figure 8.
- Identify the initial position, initial time and the velocity values for the forward and in reverse motion. Calculate the Cart's position at different instants and compare the obtained values to the recorded position in the .csv file.
- Compare the forward and reverse velocity. Graph the velocity vs. time for the three phases of the motion. Compare your graph with the Figure 9.
- How are the Cart Position vs Time graph and Cart Velocity vs. Time graph related?
- Given the velocity versus time graph, how is possible to find the distance travelled over an interval of time?
- The velocity vs. time graph in the Figure 9. has been drawn based on the calculated values of velocity for each time interval of the multistep motion presented in the Figure 8.

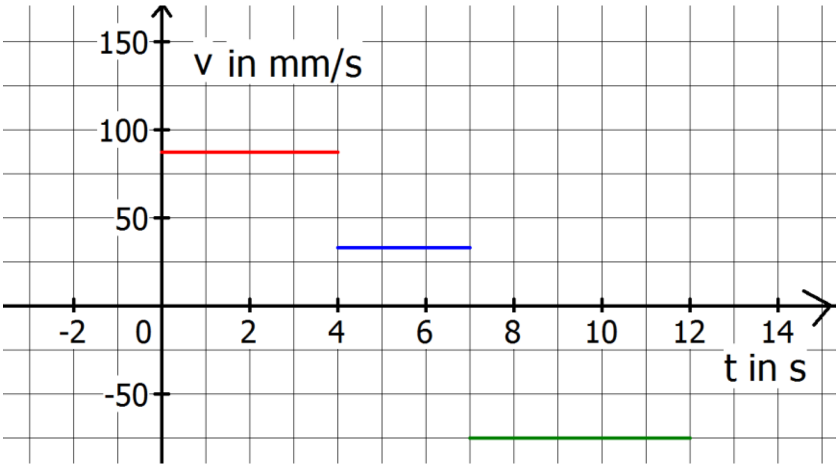

Figure 9. Simplified velocity vs. time diagram based on the velocity values in the Figure 8.

Determine the distance travelled on each time interval and on the entire multistep motion.