



Guide to timing cars (v6)

Contents

About timing in Race To The Line	2
The break-circuit based timing gate	2
Making a simple timing gate	3
Wiring a switch-based timing gate	5
Coding a simple timing gate.....	6
Example Makecode for a simple timing gate.....	7
Video analysis (a recommended backup)	8
Using video analysis:	8
Stop watch averages (a last resort backup).....	11
Video backup	11
Troubleshooting issues	12

About timing in Race To The Line

The purpose of timing at a school race day is to determine the team/s that will go forward to represent the school at the next stage.

Race operations, including timing, are under control of the school and the trained rocketeers.

Where possible, times and team names should be uploaded to the Dendrite platform. This can be done using the Dendrite website. The Learning Partnership team can also upload data if you encounter problems and are unable to.

We strongly recommend **always keeping a paper record** of times and speeds achieved at a race event and suggest you or a student team capture a video of each car's run from start line to brake to provide backup timing evidence in case there are any issues with other methods.

The break-circuit based timing gate kit

A simple timing gate can be made with a micro:bit. Timing gates can use IR beams or other sensors, but these can be hard to align and can be affected by the environment.

A kit to make a break circuit based timing gate is provided. Break circuits are easy to set up and an ideal activity for engaging students.

Assembly of the timing gate should take no more than 60-90 mins if tools and wood are available.

Timing Gate Kit Contents:

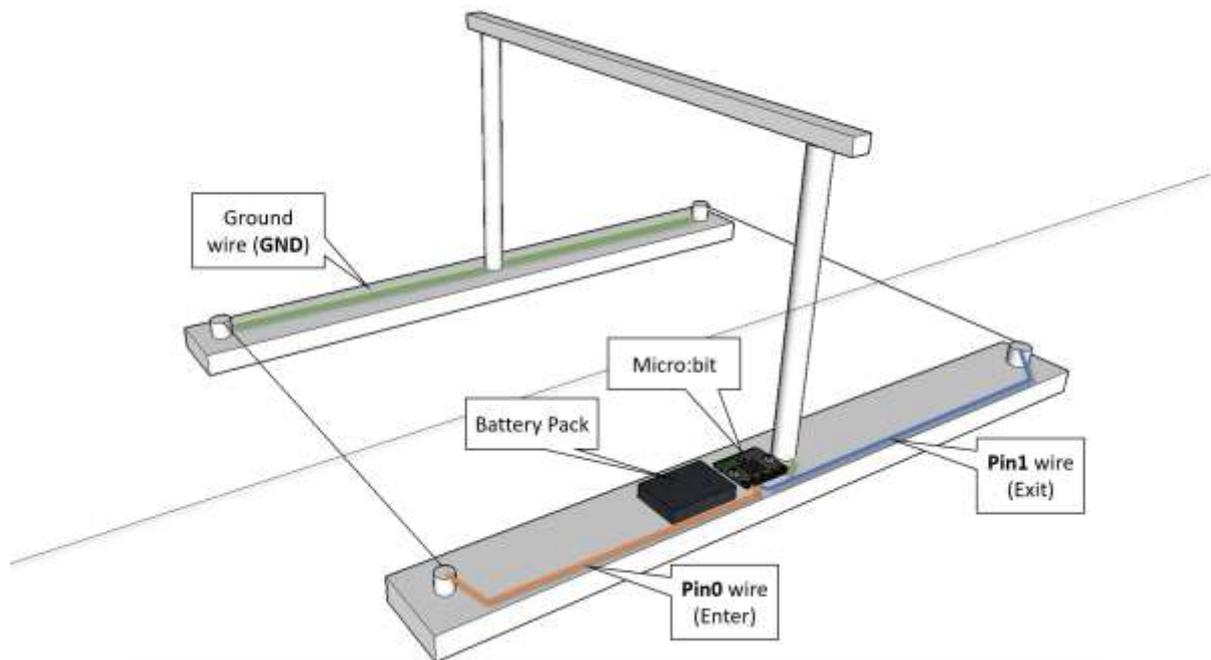
Quantity	Item	Purposes
1	Microbit	To detect circuit breaks & calculate & display times
1	Battery Box	To power the microbit and timing gate
3	M3 machine screws	To connect Pin0, Pin1 and GND to the gate circuits
3	M3 nylon spacer	To secure and insulate the connections
3	M3 ring crimps (red)	To connect circuits to the micro:bit
3	M3 nuts	To secure the crimp
4	M5 50mm countersunk bolts	To provide terminal posts, two on each side
4	M5 flanged nut	To secure the bolt to the timing gate
8	M5 large washers	To provide clip points for breakable connections
4	5mm ring crimp (blue)	To connect circuits to the micro:bit
4	M5 nut	To secure the ring crimp to the bolt
4	M5 butterfly nut	To secure the tether under the washer when racing
1	Reel of wire	To make the main connection
2	Crocodile clip test cables	To test break circuits

Extra resources needed to complete the timing gate.

- A 2.4 m length of Strip wood (or similar) cut into two 1.2m lengths (*2400mm x 68mm x 15mm recommended*)
- Assorted scrap wood, plastic plumbing pipe, or similar, to make a cable bridge
- Drill with a 5mm wood bit
- Screws, paint and varnish to complete the gate.

Making a simple timing gate

Two circuits can be made between **Pin 0** and **Pin 1** on the micro:bit to the **GND** Pin. A short amount of code can be used to determine when these circuits are made or break. If these circuits can be broken by a car (e.g. foil strip or a wire dislodged from a crocodile clip) they can be detected and the 'events' used to trigger reliable and accurate starts and stops for timing on the micro:bit.



Here students are testing their timing gate made from a 2.2m length of strip wood cut in half to make a 1m long timing gate. Foil strips held by crocodile clips and pairs of washers held by butterfly nuts on bolts are broken by the passing car and can quickly be reset between each run.



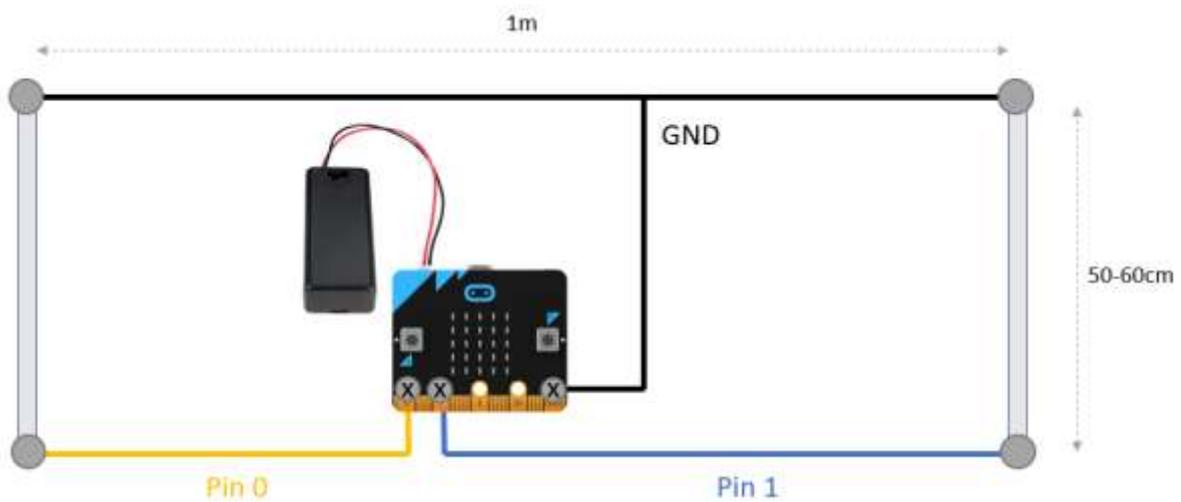
Connect trigger wires to **Pin0** and **Pin1**. Both circuits return to **GND**.

Note: M3 countersunk bolts can be used to secure wires to the micro:bit pin holes.



Set the gate at least 50cm wide to ensure cars do not hit it. In this team's design, a plastic tube carries the wires returning to GND over the track and tether.

Wiring a switch-based timing gate



- 1) Download and test the code and on the microbit using crocodile clips in the holes.
- 2) Thread three M3 countersunk bolts through the holes marked **0**, **1** and **GND** in the micro:bit and secure them underneath with the nylon spacers.
- 3) Make connections between the micro:bit **Pin 0** and **Pin 1** to two 5mm posts on the microbit side of the timing gate. *Use the small red 3mm ring crimps for the micro:bit ends and the blue 5mm ring crimp for the other end.*
- 4) Make connections between **GND** on the microbit and the two posts on the others side of the timing gate. *NOTE: You will need to make a bridge to carry the GND wire over the track and the path of the cars. QUESTION FOR STUDENTS (could it fold or disassemble?)*
- 5) Create *breakable connections* between the posts opposite each other on the timing gate. These can be foil strips clamped between washers, wires clamped to a washer by clothes pegs that pull away, copper conductors that swing aside or crocodile clip leads gripping a large washer on each side. A pair of crocodile clips wires is supplied in the kit.

Coding a simple timing gate

Code for students to edit and use for a simple timing gate is here:

https://makecode.microbit.org/_AyE323EsXE3q

NOTE: This code will measure the **time in milliseconds**. If the break gates are 1m apart the display will show in **MILLISECONDS/METRE** for a car. The lowest number is the fastest.

Using the sample program

- 1) Download the program and install the .hex onto the timing gate micro:bit.
 - a. *Plug in the micro:bit via USB It should appear like a USB drive*
 - b. *Click the Download button in Makecode to generate a .hex file*
 - c. *Drag to .hex file to the micro:bit. A light should flash as the code downloads and then the micro:bit should reset and be ready to use.*
 - d. *OPTIONAL: You can use the 'Share' option in MakeCode to save and share any modified version of the code at a URL.*
- 2) Assemble the timing gate and connect the wires and foil strip/wires that will be dislodged by the car as it passes.
- 3) Switch on the micro:bit
- 4) Press the A button on the micro:bit to test the circuit. A cross means check the wiring a tick means it is OK (if using the sample code)
- 5) Press the B button and two lines will appear representing the gates
- 6) As the car passes through and breaks the circuits (*Pin0 first then Pin1 second*) the lines will disappear, and the time taken (in ms) will scroll several times and can be captured. Shorter times are faster.
- 7) Press the A & B buttons together to reset the gate for the next car, checking the circuits are working before firing the next car.

There are several ways to extend and improve the sample code that could also be investigated:

TRANSMIT THE TIME TO ANOTHER MICRO:BIT

The micro:bit radio commands allow text and number to be transmitted to any other micro:bit using the same radio channel.

CALCULATE AND DISPLAY THE VELOCITY

The timing gate records the time taken for the car to break both circuits. If the timing gate length is 1m both time and distance are known and velocities such as M/S or MPH can be calculated and displayed.

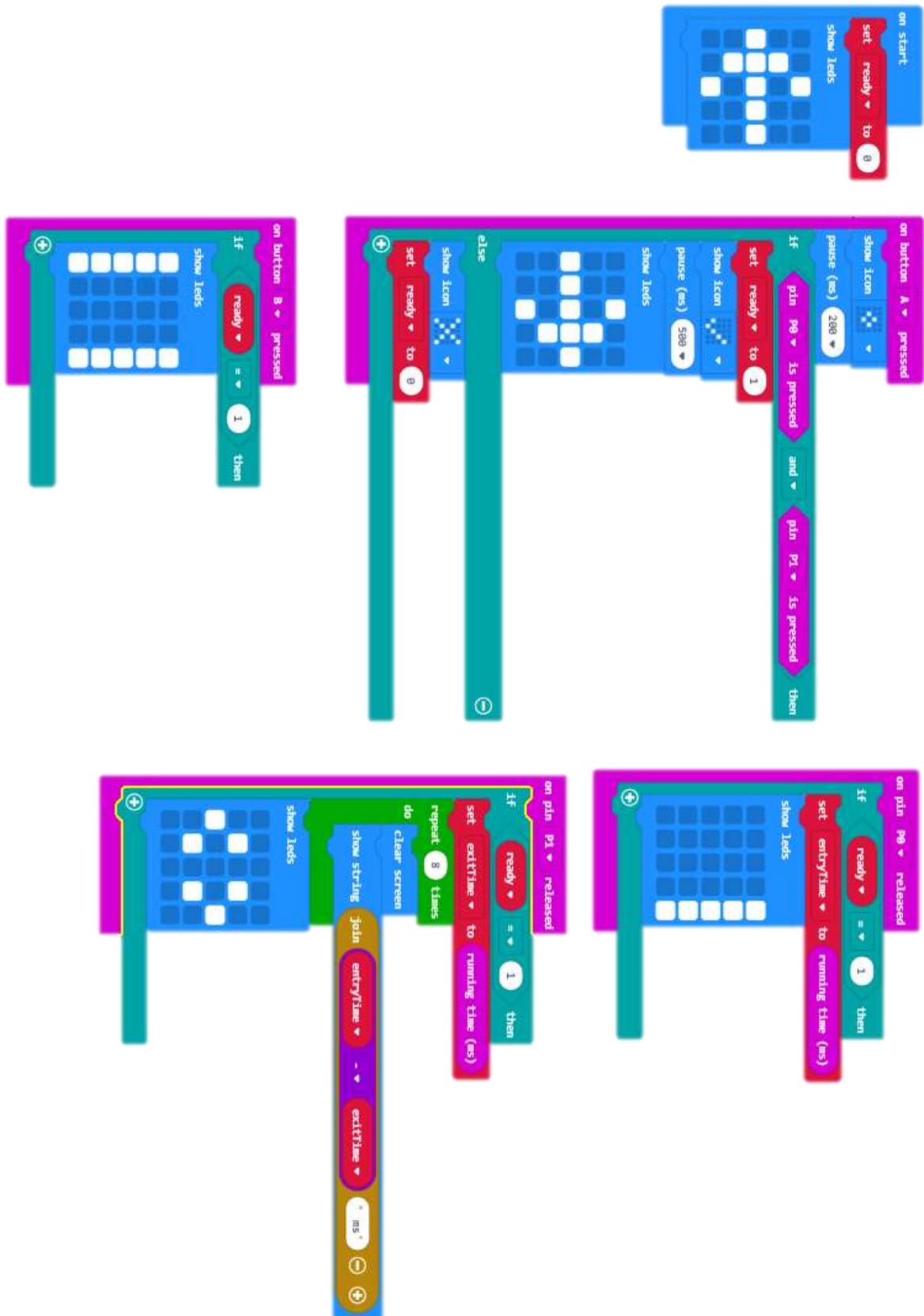
DISPLAY TIMES IN MICROSECONDS FOR MORE PRECISION

It is possible several similar designs of car may go through with the same time. You could adapt the code to use microseconds rather than milliseconds

DEVELOP CODE IN ANOTHER LANGUAGE SUCH AS PYTHON

Python will allow the use of floating point numbers so that times with decimal points can be displayed.

Example Makecode for a simple timing gate



Video analysis (a recommended backup)

Video analysis is a very reliable method of capturing the velocity of a moving car in m/s.

We recommend the **Vernier Video Physics** application on iOS devices. (Small Cost). This can be used trackside on a mobile apple device and is ideal to be used by a student timing team.

A presentation on video analysis can be found in the competition's education resources area.

Vernier Video Physics app info:

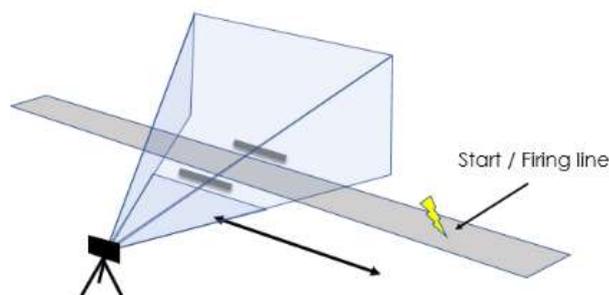
<https://www.vernier.com/products/software/video-physics/>

App store link: <https://itunes.apple.com/us/app/vernier-video-physics/id389784247>

Using video analysis:

The tablet or iPhone camera records the car moving past. The position of the car in each frame of video is recorded and graphed and the velocity across the view area is calculated and displayed. The velocity of the car in the middle of the view is used as the ranking measure.

- 1) Set up a recording position approx. 5m from the track observing a timing area that should be marked with two 'speed trap' marks near the tether line 1m apart and that can be seen. (this is the reference distance) The camera should be held still and perpendicular to the travel of the car and the centre of the view should be the middle of the 1m speed trap area. The device should be in landscape mode. **The speed trap should be 25m from the firing line.**



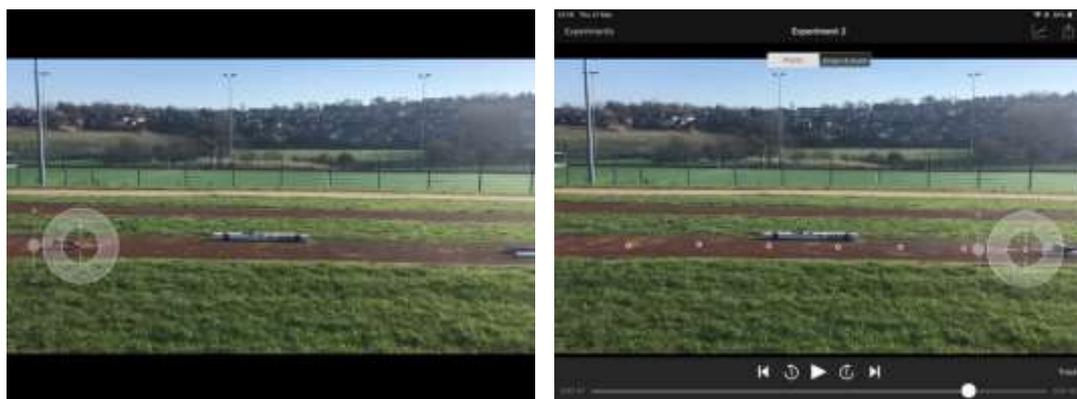
- 2) Open the Vernier Video Physics app and Press the **+ button** at the upper right to add a new video recording (you can also open existing recordings and do analysis afterwards)



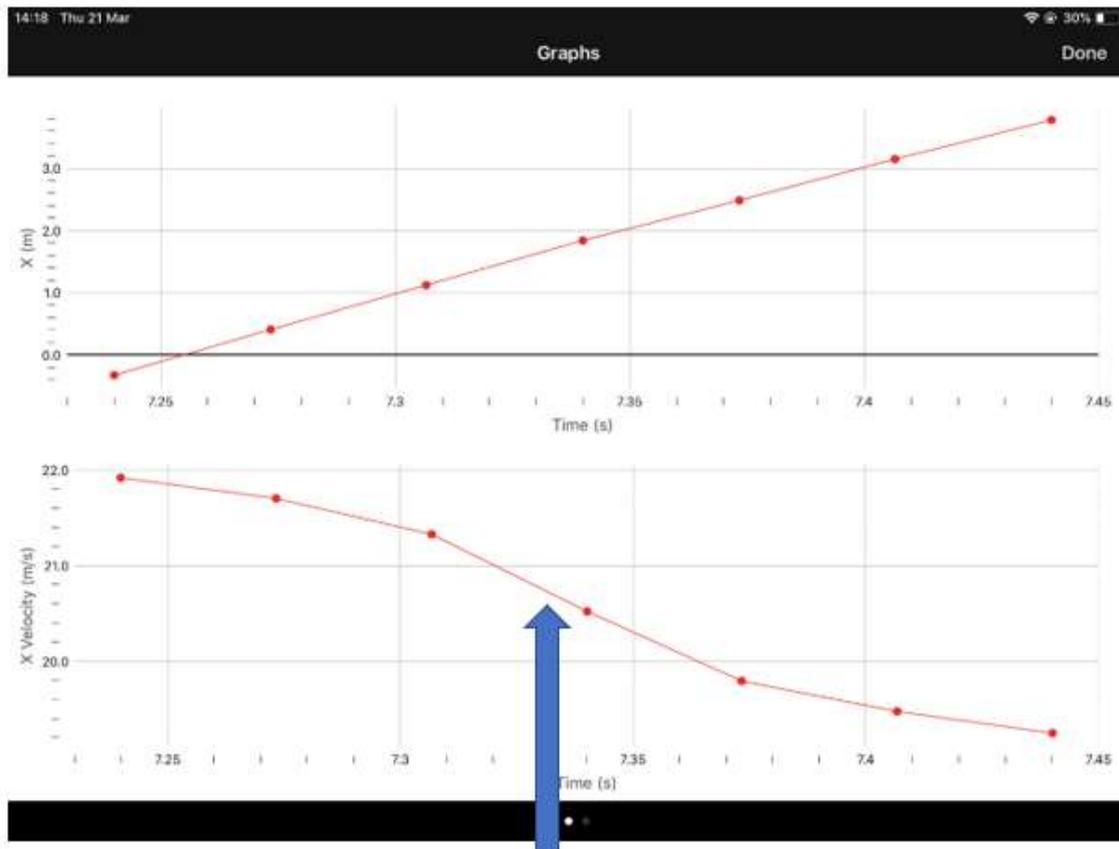
- 3) On the “Track Live” countdown announcement, **begin recording** and say the name / number of the team (*a tip in case of issues later cars can be identified*). **After the car has passed, stop the recording.**
- 4) You can preview the video at this point but choose the “**Use Video**” option bottom right.
- 5) Use the “**Origin and Scale**” option to set the two white circles connected with a line to the ends of the 1m speed trap. This calibrates the analysis.



- 6) Drag the video slider at the bottom till the car appears coming from the side. Use the “Points” option and drag the marker to the nose of the car and then tap. A dot will appear and the video will advance to the next frame. Move the marker to the nose of the car in the new frame and repeat until the car leaves the area. This will be around 5-10 dots depending on the speed of the car and will take just take a few moments.



- 7) Tap the **graph icon** top right to show the motion and velocity data, swipe to the left to show the velocity graph. Record the speed indicated on middle of the **X velocity line**.



This has recorded a velocity of **20.6 m/s** at the centre of the timing area.

- 8) Keep a record of the velocity for the team on paper and you can enter it manually into the app if you want. The higher velocities through the timing area are the winners.

TIPS FOR GOOD QUALITY VIDEO ANALYSIS

- Have a team practice with the app in advance. A skilled student team should be able to analyse each car's speed while the next car is being set up to fire.
- Say the name of each car on the recording as they can all look the same.
- Have a second mobile device capture the same view. In case of issues this secondary video sources can be analysed afterwards.
- Try to ensure bright light. The video images of the cars can be blurred if the lighting is poor and the position of the cars harder to determine.
- (Optional) If you can image the entire track from the side (e.g. from a balcony or upper building window) it is possible to build a graph of the total motion of the car through acceleration and deceleration and determine the peak velocity.

Stop watch averages (a last resort backup)

While not as accurate, an average of times taken on several stopwatches can be used as a measure. A minimum of three parallel captured times is recommended.

This method works best if cars can travel the full length of the tether, providing the longest run times.

Students and rocketeers record the time from ignition to arrival in the gate. If all the times are similar, then an average can be taken. If one measure does not closely correspond with the other two, it can be discounted. Desk timers and stopwatches with 100th second resolution are ideal.



Video backup

The switch based timing gate and video analysis methods are recommended.

If video is kept of each run showing the whole run of the car, options exist to measure the time taken using video software that shows video clip lengths in seconds and fractions of a second.

This is not something that can be used trackside but can be used afterwards. *Not recommended as the main method.*

Troubleshooting issues

Micro:bit not working

Possible flat battery. If the micro:bit has been left on overnight it can drain a battery. Replace the battery.

Program downloads but does not work or work as expected.

It may be necessary to update the micro:bit firmware. Some micro:bits may have older firmware (*the code that resides on the micro:bit all the time*). While pressing the reset button on the back of the micro:bit connect the USB cable. The micro:bit will appear on the computer as a folder called "MAINTAINANCE" – drag the latest firmware file to the micro:bit to update. Full details and the latest firmware files are here: <https://microbit.org/guide/firmware/>

I get a Bitlocker or non-encrypted drive alert

Many schools have installed software to comply with GDPR. These require USB devices to be encrypted. Micro:bit cannot operate via USB. The IT dept. should be able to create a profile or exception for micro:bit if you show them the issue.