Projectiles Math Lab

Graphing, Predicting,
Robot Paintball, Curve Fitting
And MORE!

Why Projectiles?

If you have to ask, you won't get it...

Projectiles are FUN!

And you can learn quite a lot of Math and Physics, too...

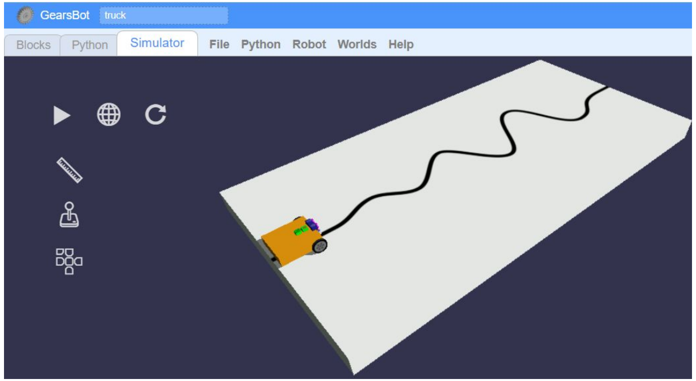
Gears

Gears is a Virtual Robotics platform.

We are going to use it to model some projectile rocketry before launching the real deal.

Let’s open the Gears platform

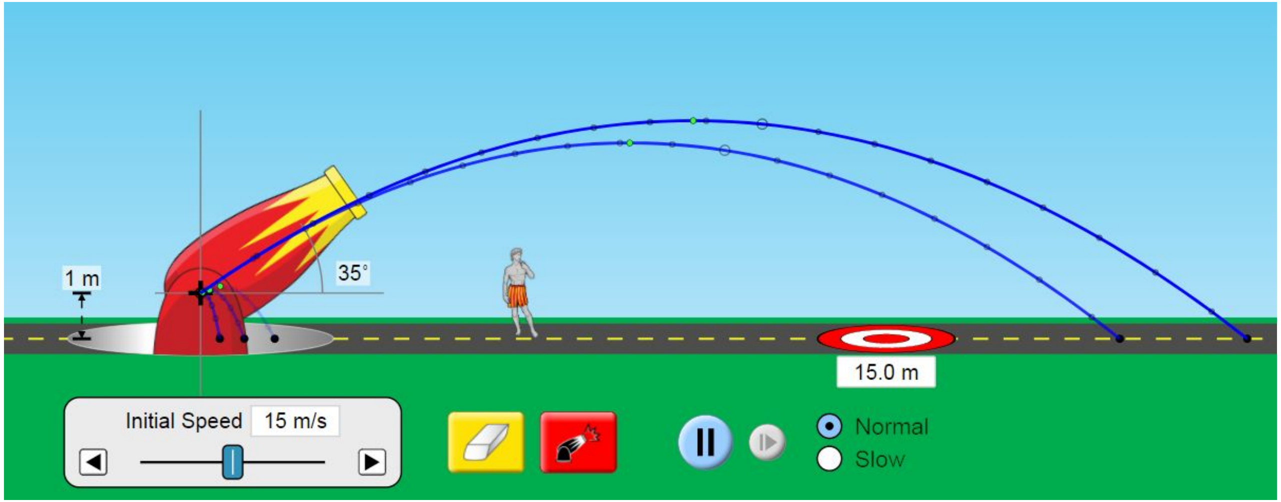
http://a9i.sg/gears



Projectile Basics

When you shoot an object, the two main factors in determining its path and distance to impact are:

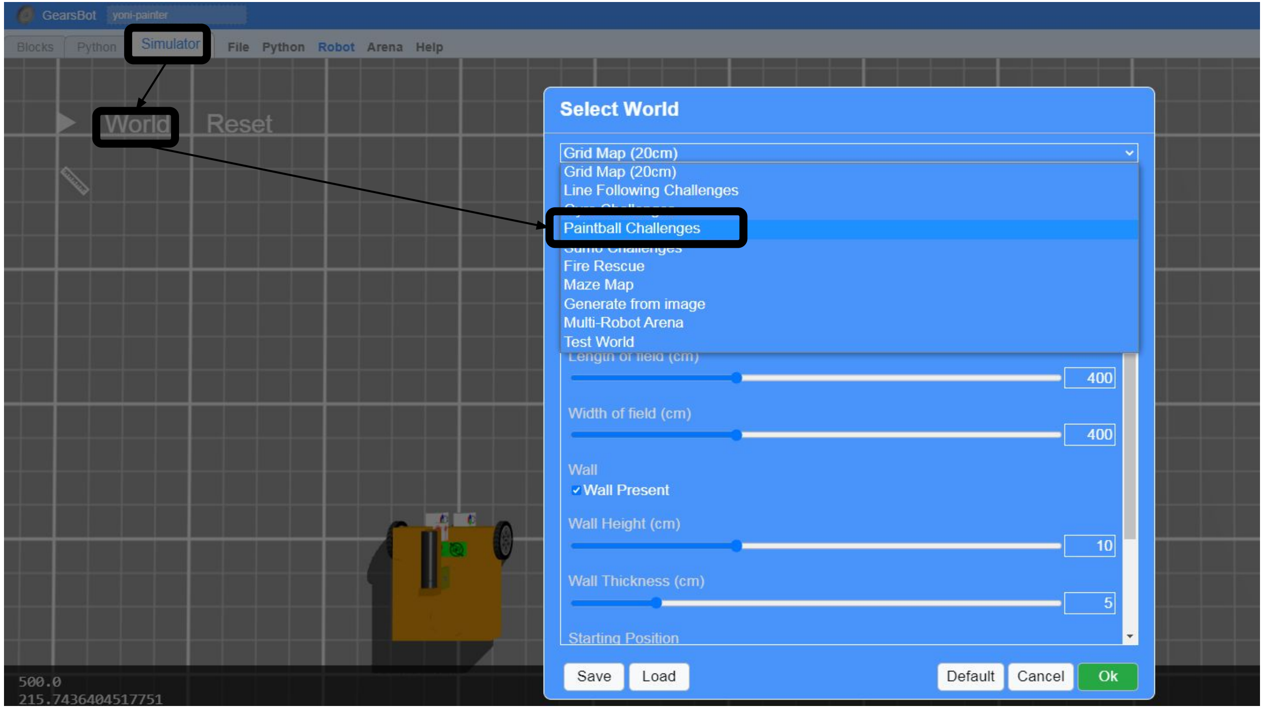
- Initial Speed
- Projectile Angle



Click image to experiment with ***general*** *projectile physics (NOTE: not exactly applicable to GearsBot Paintball)*

Setup Simulator

- Select Simulator
- Select World
- Choose **Paintball Challenges**



Grid Map (20cm)

Length of field (cm) 400

Width of field (cm) 400

Wall
☑ Wall Present

Wall Height (cm) 10

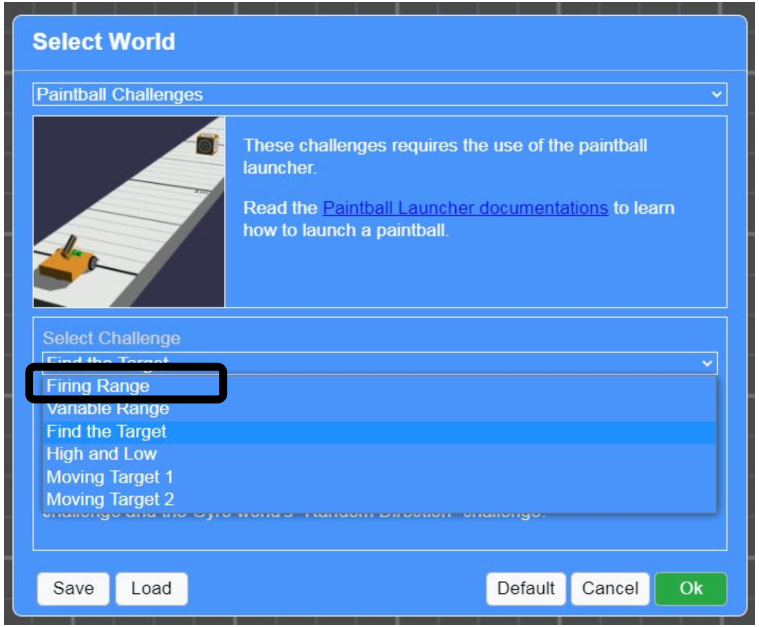
Wall Thickness (cm) 5

Starting Position

```
500.0
215.7436404517751
```

Setup Simulator - Firing Range

- Change to **Firing Range** Challenge



Select World

Paintball Challenges

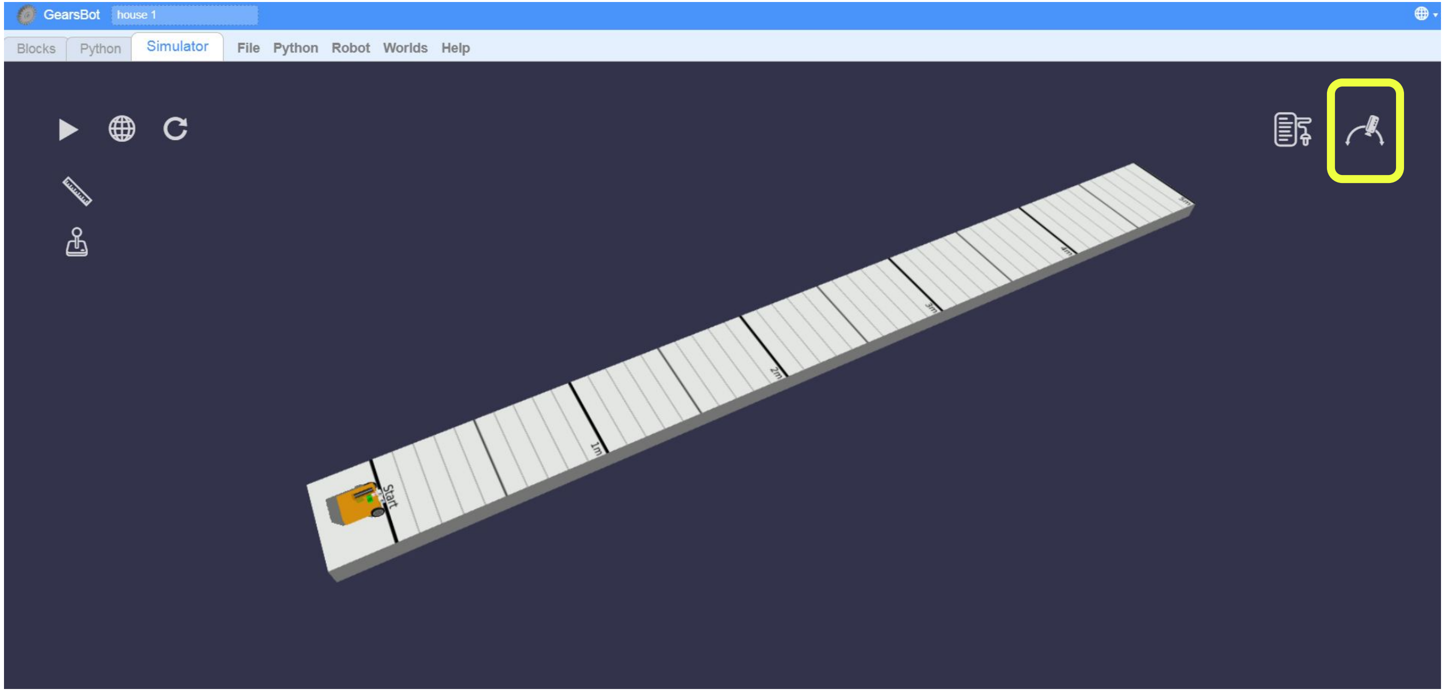
These challenges requires the use of the paintball launcher.

Read the Paintball Launcher documentations to learn how to launch a paintball.

Select Challenge

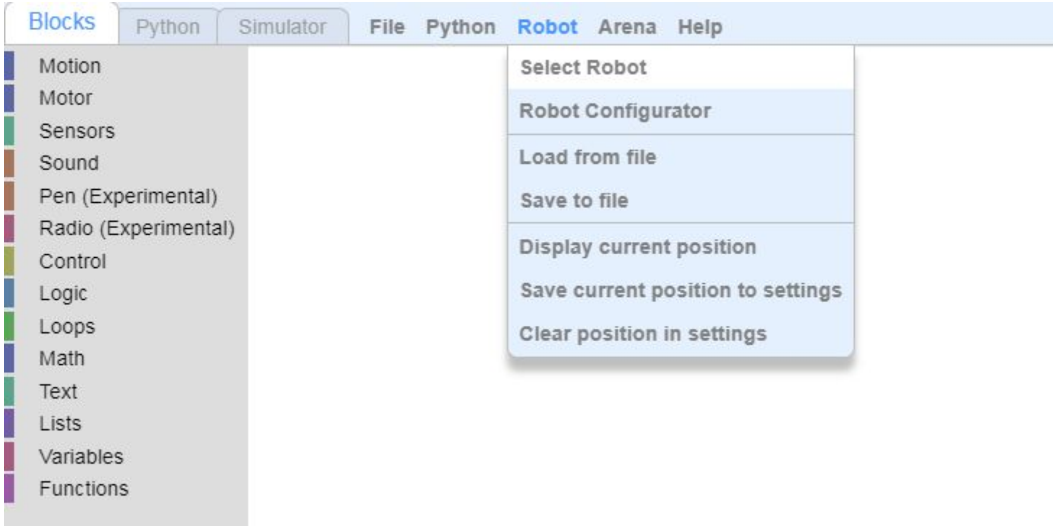
Setup Simulator - Firing Range

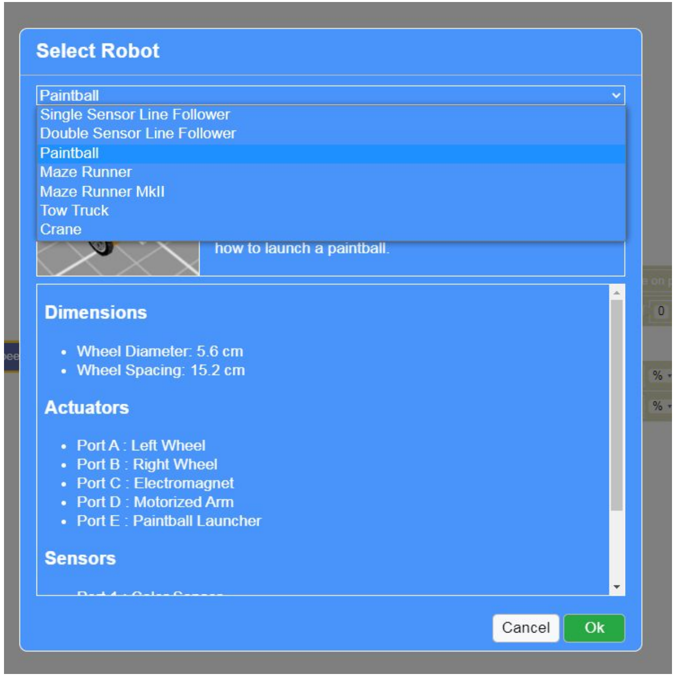
- Change Camera to get a good view of entire Firing Range:



Select Robot

- Select the **Paintball** Robot





how to launch a paintball.

Dimensions

- Wheel Diameter: 5.6 cm
- Wheel Spacing: 15.2 cm

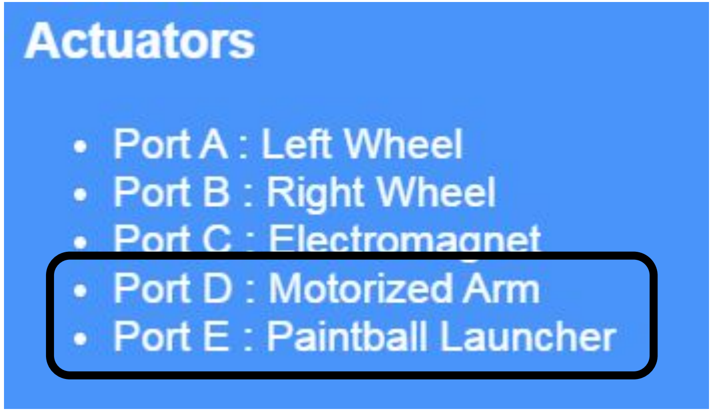
Actuators

- Port A : Left Wheel
- Port B : Right Wheel
- Port C : Electromagnet
- Port D : Motorized Arm
- Port E : Paintball Launcher

Sensors

Paintball Robot

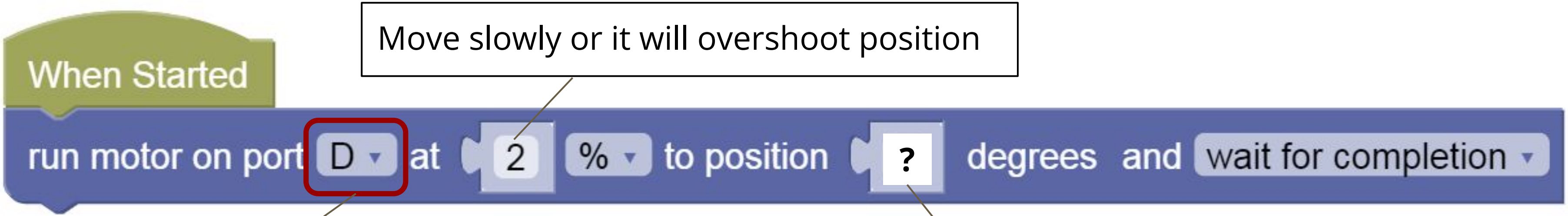
- Note which Motors you need to control to shoot balls!
  - **Port D:** Moves canon up and down (Angle)
  - **Port E:** Spring-loaded mechanism to shoot (Speed)



Actuators

- Port A : Left Wheel
- Port B : Right Wheel
- Port C : Electromagnet
- Port D : Motorized Arm
- Port E : Paintball Launcher

Projectile Basics - Vary Angle



```
When Started
run motor on port D at 2 % to position ? degrees and wait for completion
```

Move slowly or it will overshoot position

Motor D Controls Arm

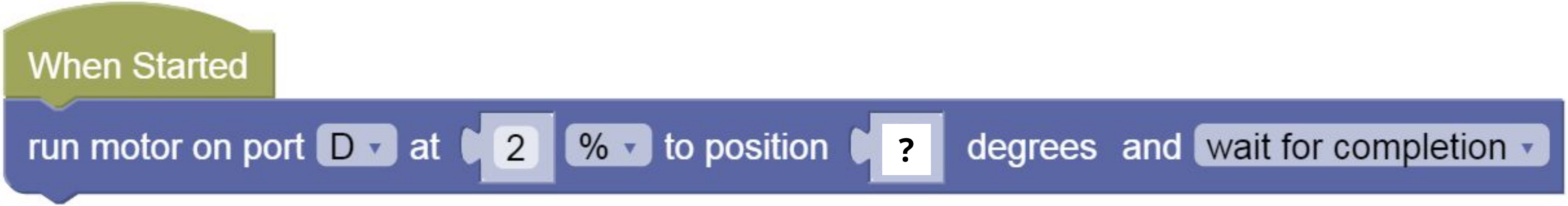
Experiment with moving Arm to 30, 45, 60, and 75 degrees

Projectile Basics - Vary Angle

↻

Make sure to RESET each time.

Experiment with running the program twice without resetting to see the unwanted behavior.

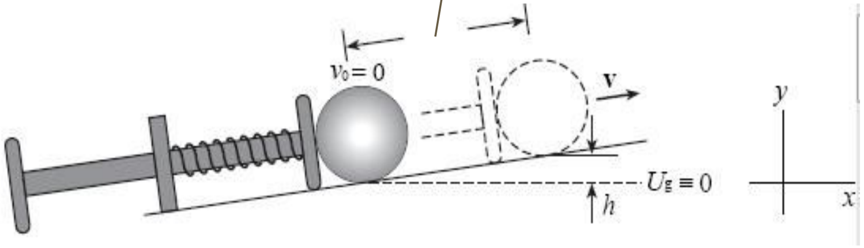


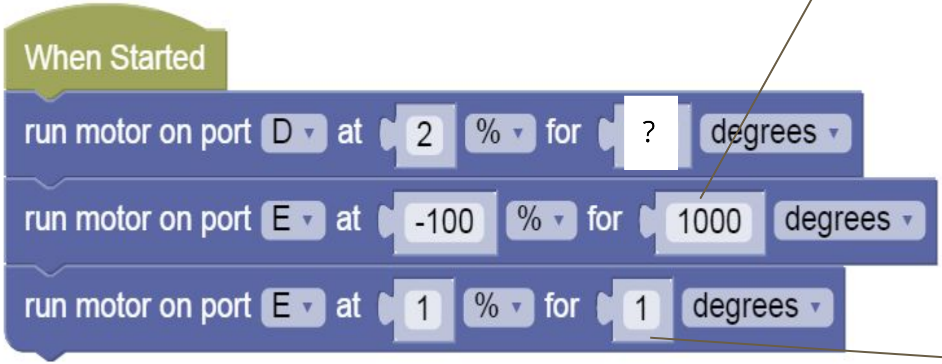
```
When Started
run motor on port D at 2 % to position ? degrees and wait for completion
```

Projectile Basics - Firing Range

- Now to shoot, you need to wind the internal spring back
- And then let go...

This is how much we are loading the spring....



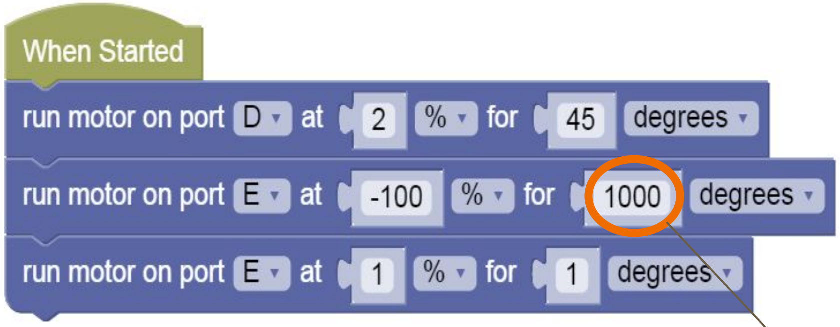


```
When Started
run motor on port D at 2 % for ? degrees
run motor on port E at -100 % for 1000 degrees
run motor on port E at 1 % for 1 degrees
```

This is just to release the spring mechanism... eg SHOOT

Projectile Basics - Experiment with Velocity

- Experiment with changing **Velocity**
- Try different values from 0 to 1000

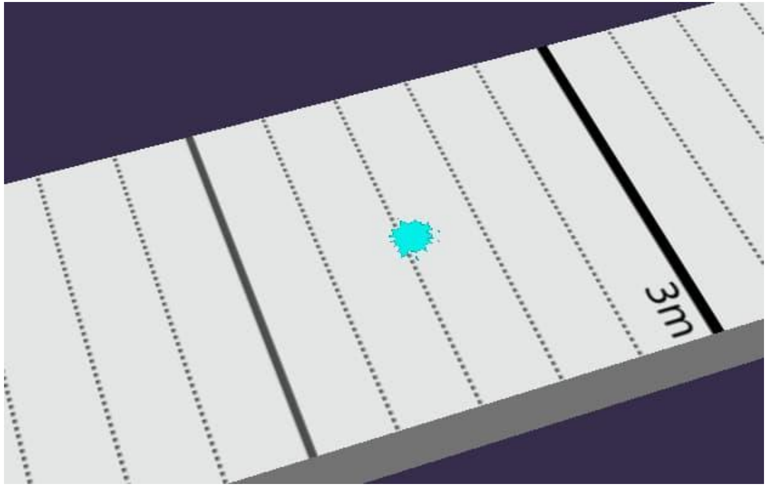


```
When Started
run motor on port D at 2 % for 45 degrees
run motor on port E at -100 % for 1000 degrees
run motor on port E at 1 % for 1 degrees
```

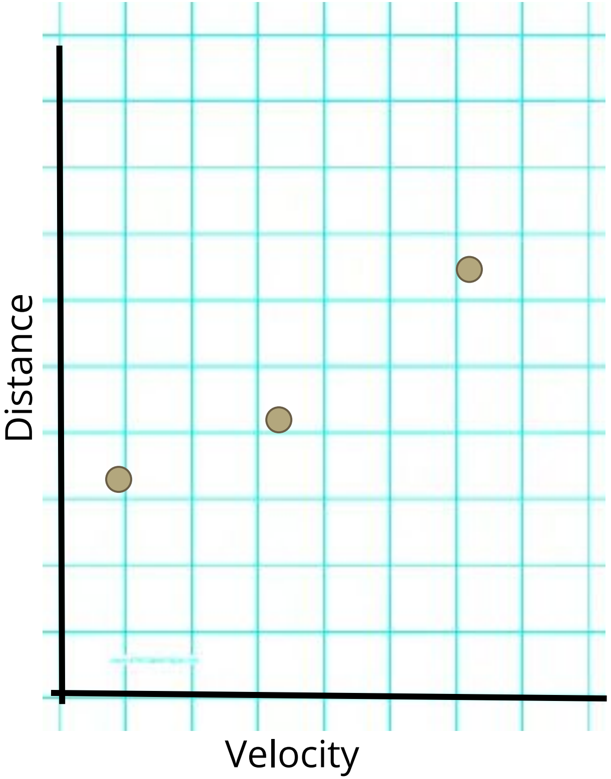
Velocity depends on this setting

Some Math, Finally - Graphing

Keep varying the **Velocity**, and record on your graph paper the distance travelled. Can you draw a curve line that fits the points? What is the shape of this line?

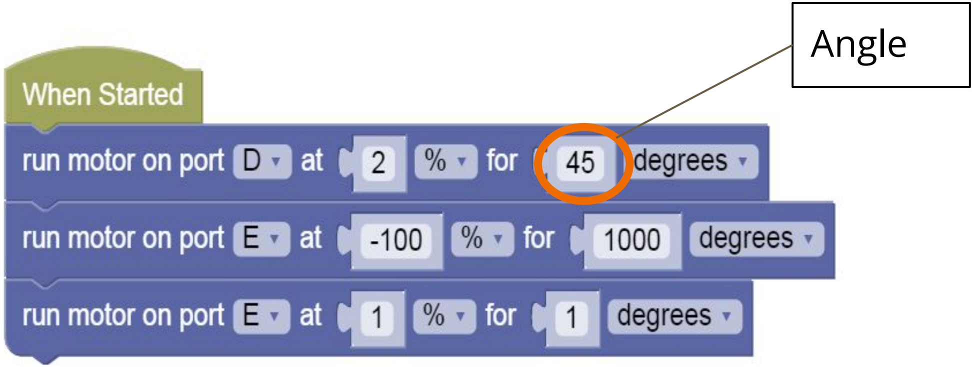


In this example, the distance is around 270cm



Projectile Basics - Experiment with Angle

- Experiment with changing **Angle**
- Try different values from 0 to 90

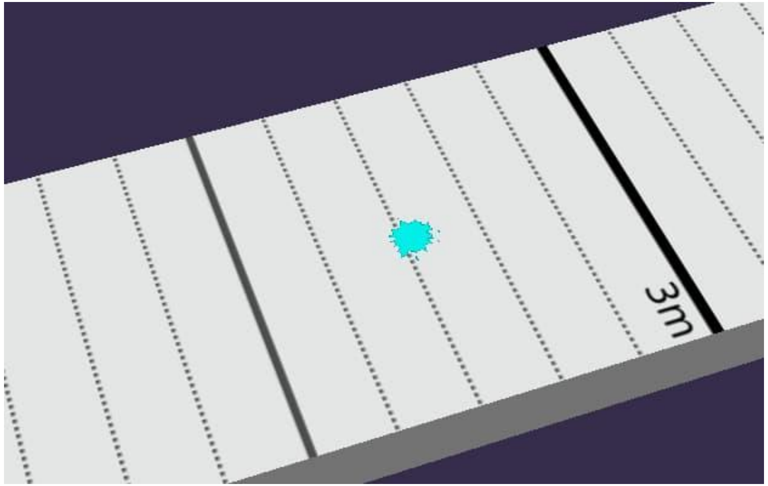


```
When Started
run motor on port D at 2 % for 45 degrees
run motor on port E at -100 % for 1000 degrees
run motor on port E at 1 % for 1 degrees
```

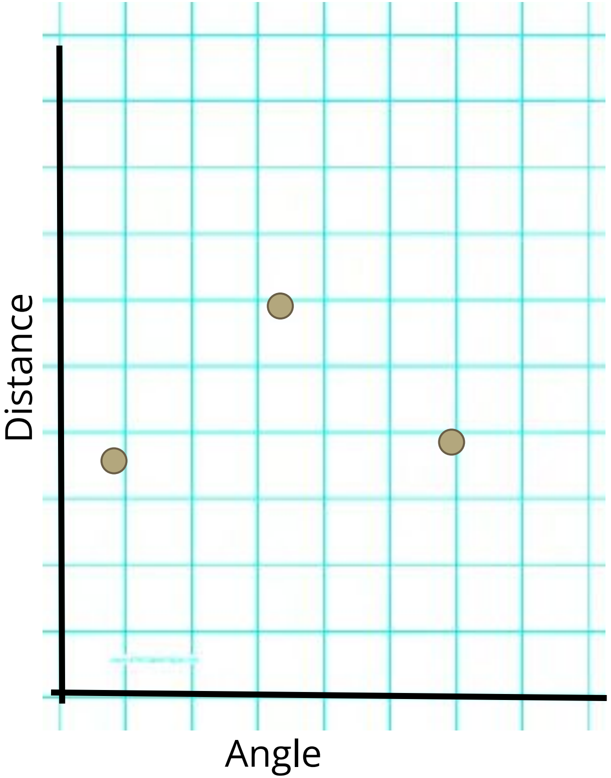
Angle

Some Math, Finally - Graphing

Keep varying the **Angle**, and record on your graph paper the distance travelled. Can you draw a curve line that fits the points? What is the shape of this line?



In this example, the distance is around 270cm



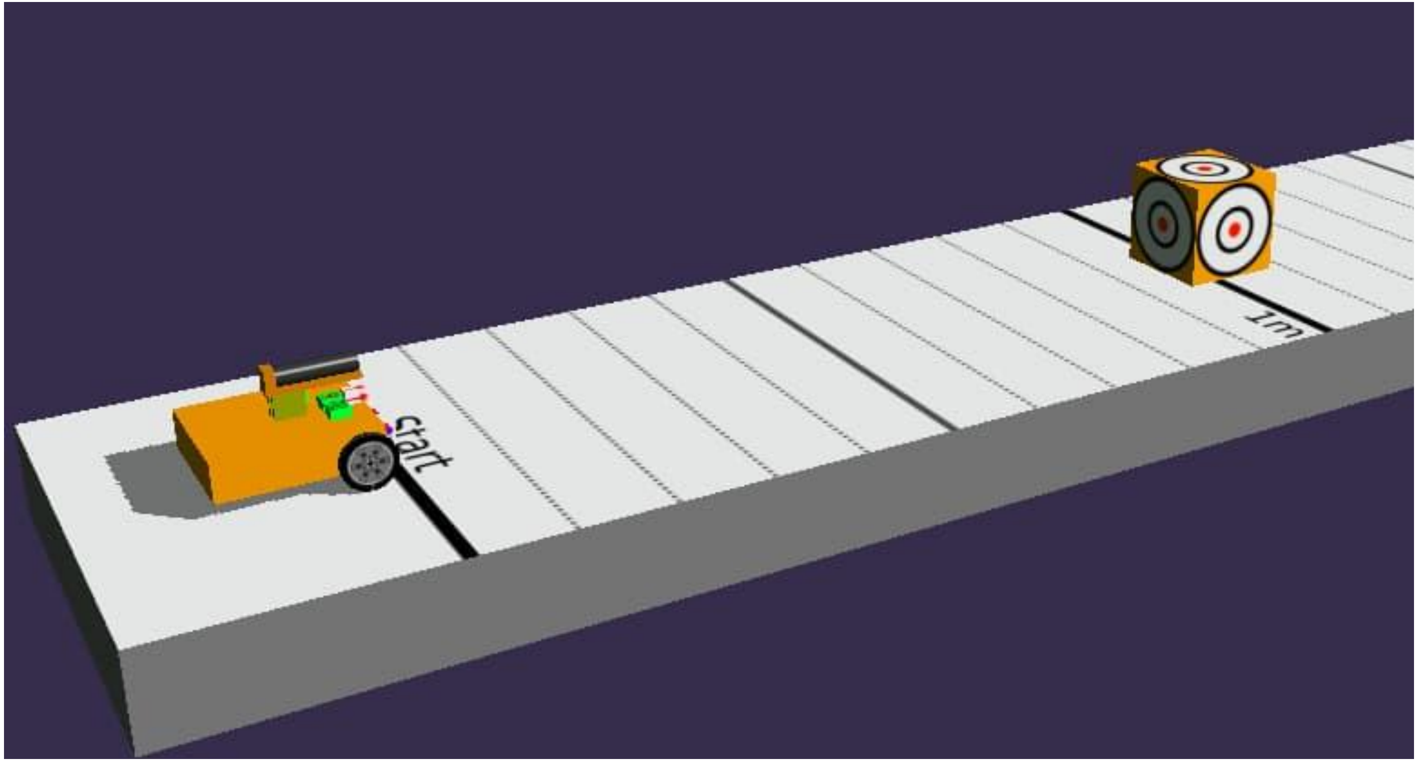
Best Angle

Use the above technique to find the **peak-distance angle**.

That is, the angle that gives you the farthest distance for your shot, given a constant initial velocity.

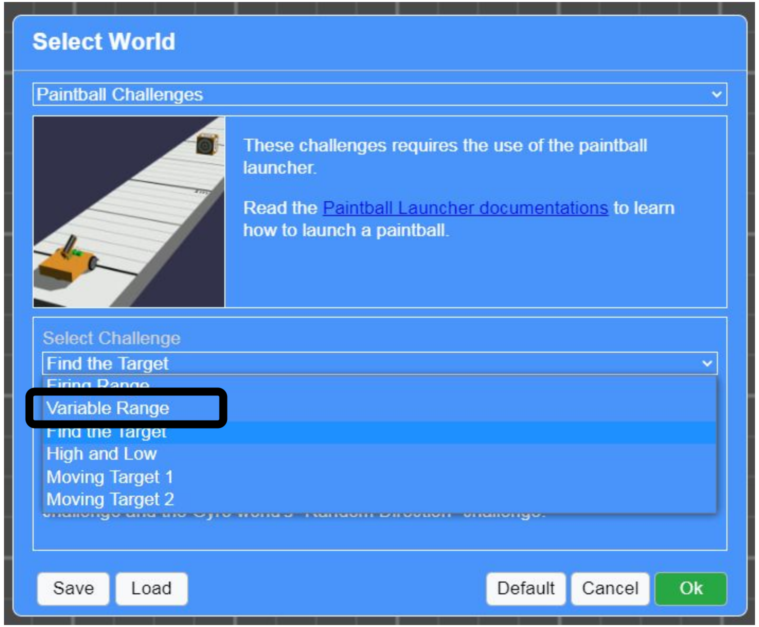
?

Hitting a Target



Hitting a Target

- Switch to the **Variable Range** challenge
- This challenge will provide you with a target to hit
- Target position changes on every reset



Select World

Paintball Challenges

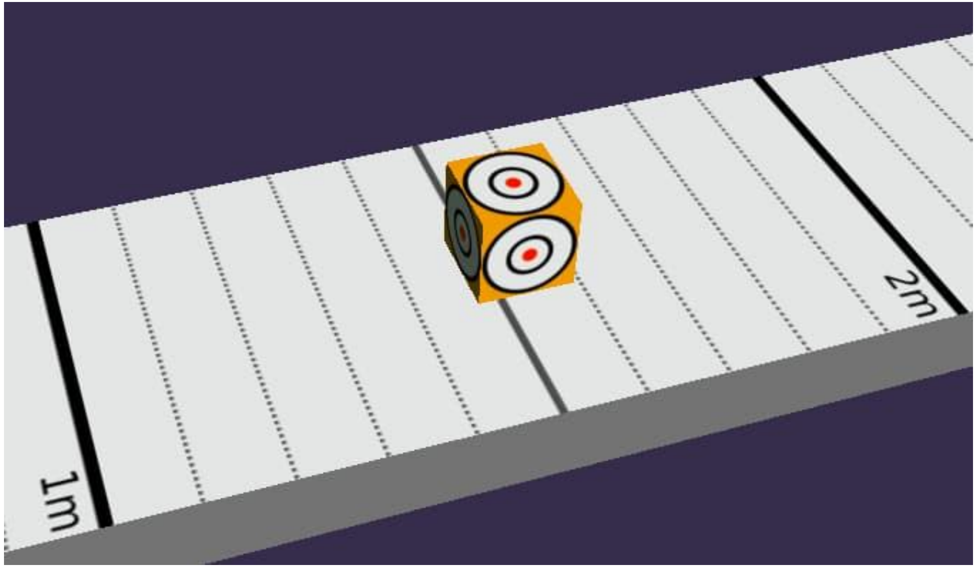
These challenges requires the use of the paintball launcher.

Read the Paintball Launcher documentations to learn how to launch a paintball.

Select Challenge: Find the Target

Hitting a Target

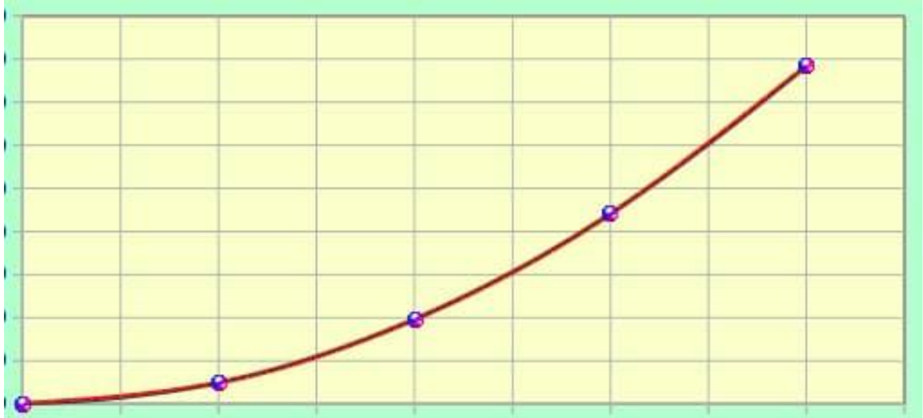
- Visually check the distance to the target
- Use the graphs you have prepared to determine the best **angle** and **velocity** to launch the paintball
- Try with **both** graphs.
    - Use 45 degs, and determine the correct velocity
    - Use 1000 power, and determine the correct angle



In this example, the target is around 150cm away

Hitting a Target Autonomously

- To hit the target autonomously, the robot will need to decide for itself, the right **angle** and **velocity** to use…
- …but the robot can't read your graph, so you'll need to convert it into an equation!
- What is the equation of the best fit curve you have drawn?



Distance - Velocity

- The relationship between distance and velocity looks like this…

  Distance = Velocity^2 x K

  Where K is a constant.

- We'll need to solve for the value of **K**

Solving: Distance = Velocity^2 x K

- Pick one of the point on your graph
- Read the **distance** and **velocity** at the point
- Substitute these values into the equation and solve for **K**...

  Distance = Velocity^2 x K
- Example (distance = 230cm and velocity = 850)

  $230 = 850^2 \times K$

  $230 = 722500K$

  $K = 0.000318$ (...don't copy this; it's not correct. Calculate yourself)
- You should repeat this for each of your points, and determine what is the best **K** to use

Calculating Velocity

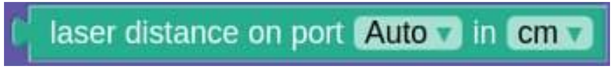
- Rearrange the equation to make **Velocity** the subject…

  Distance = Velocity^2 x K

  Velocity^2 = Distance / K

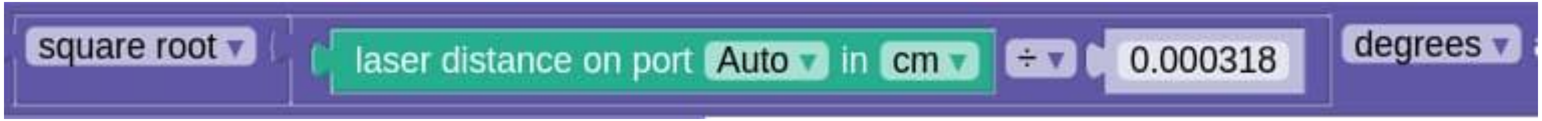
  Velocity = sqrt(Distance / K)

- The distance to the target can be determined using the laser distance sensor on the robot…



laser distance on port Auto ▾ in cm ▾

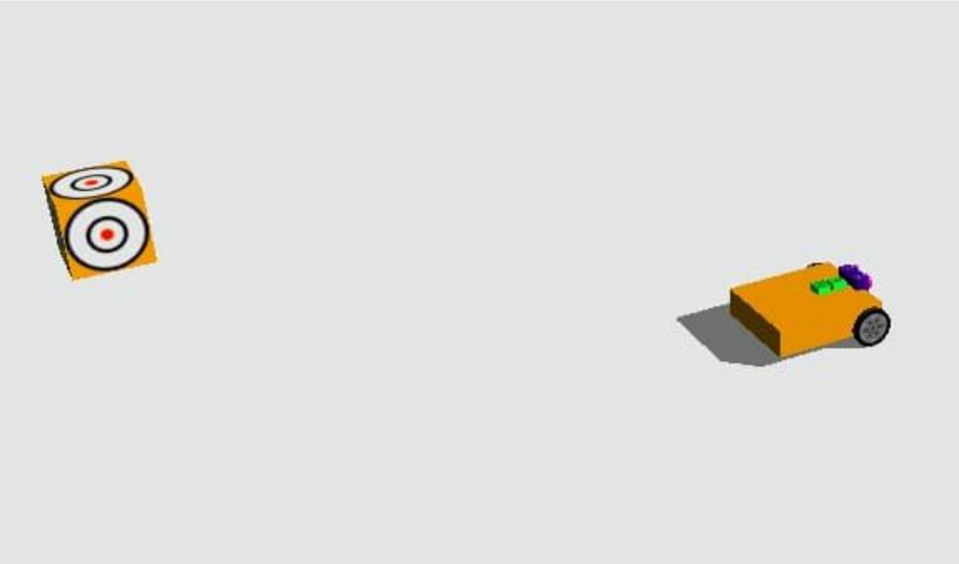
- …and you can write the equation in blocks as…



square root ▾ laser distance on port Auto ▾ in cm ▾ ÷ ▾ 0.000318 degrees ▾

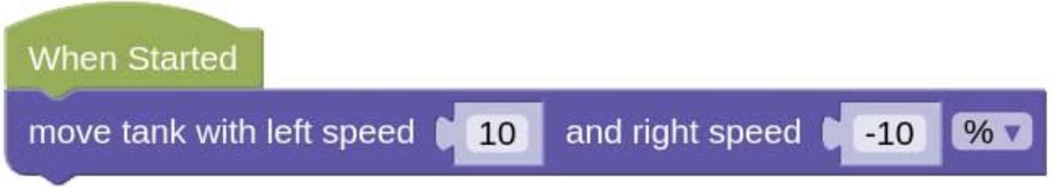
Turning and Detecting Targets

- In the “Find the Target” challenge, the target is at a random position (...not in front of the robot)
- You will need to turn the robot to face the target
- How can you detect the target?

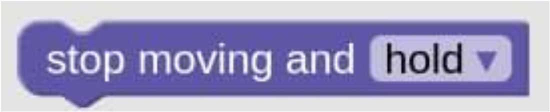


Turning the Robot

- You can turn the robot using a “move tank” block (...under the Motion category
- Set one wheel to move forward (positive number) and the other to move back (negative number)
- The robot will keep moving until it receives a stop command



```
When Started
move tank with left speed 10 and right speed -10 % ▾
```

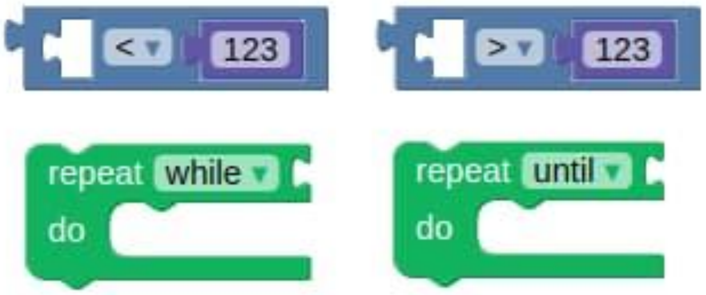


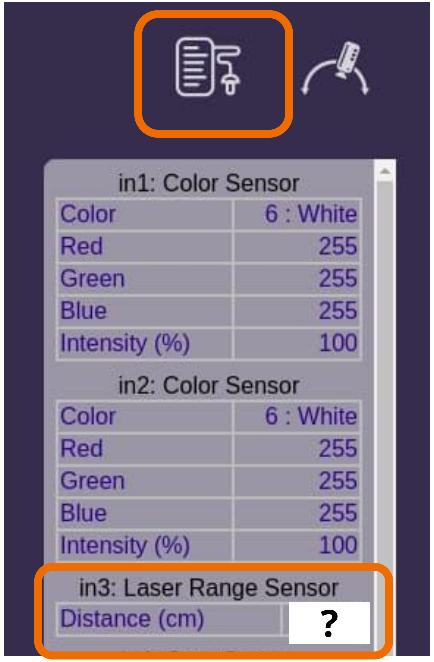
```
stop moving and hold ▾
```

Use a “stop moving” command to stop the robot.

Detecting the Target

- Use the “Laser distance” sensor
- What distance is reported when there is nothing in front of the robot?
- You can use these blocks to determine if there is a target in front of the robot. Can you figure out how?





in1: Color Sensor

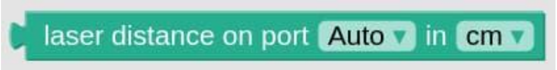
Color	6 : White
Red	255
Green	255
Blue	255
Intensity (%)	100

in2: Color Sensor

Color	6 : White
Red	255
Green	255
Blue	255
Intensity (%)	100

in3: Laser Range Sensor

Distance (cm)	?



Find the Target Challenge

- Combine the autonomous target hitting program with the target finding program
- Test your program and make sure your robot can find and hit the target by itself!

Bonus Challenge: High and Low

- Four targets
  - All are 200cm away
  - Target 1: 100cm above robot
  - Target 2: 50cm above robot
  - Target 3: Same level as robot
  - Target 4: 50cm below robot
- Experiment and find how the velocity / angle affects the height of the paintball
- Plot it out, what is the shape of the graph?