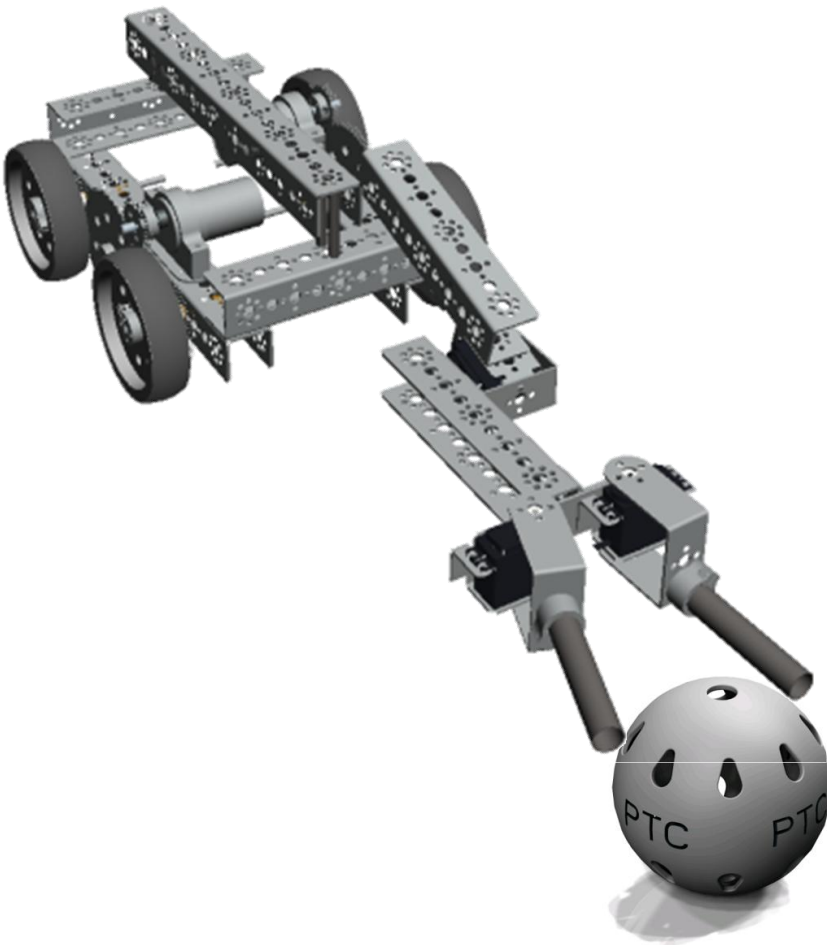
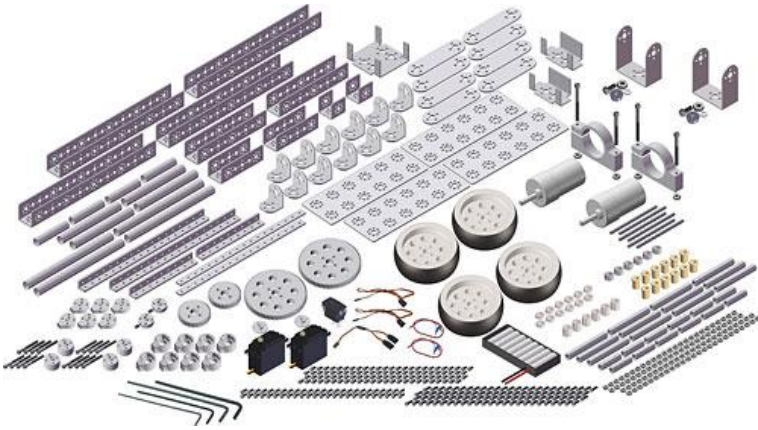


# FIRST Tech Challenge

## Robotics

Pro|ENGINEER Wildfire 4.0

Schools and Schools Advanced Edition



**FIRST Tech Challenge** - Robotics



W4-SE-L2-02-09-1.0

**Written by**  
**Tim Brotherhood**

These materials are © 2009, Parametric Technology Corporation (PTC)  
All rights reserved under copyright laws of the United Kingdom, United States and other countries.

PTC, the PTC Logo, Pro|ENGINEER, Pro|DESKTOP, Wildfire, Windchill, and all PTC product names and logos are trademarks or registered trademarks of PTC and/or its subsidiaries in the United States and in other countries.

**Conditions of use**

Copying and use of these materials is authorized only in the schools colleges and universities of teachers who are authorized to teach Pro|ENGINEER in the classroom.

All other use is prohibited unless written permission is obtained from the copyright holder.

**Acknowledgements** ***FIRST***

PTC -, Andy Deighton, Mark Fischer, Phil Darlington, Alister Fraser, Vali Baci, Robin Saitz

LabVIEW – Ben Zimmer

Control – Al Kicks

**Feedback**

In order to ensure these materials are of the highest quality, users are asked to report errors to the author.

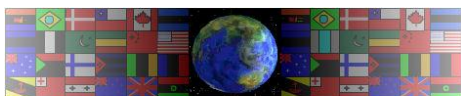
[tbrotherhood@ptc.com](mailto:tbrotherhood@ptc.com)

Suggestions for improvements and other activities would also be very welcome.

**Product code** W4-SE-L2-02-09-1.0

**Dedicated to** George F. Brotherhood

The following organizations share a common goal of promoting STEM subjects and endorse PTC's schools curriculum.



WORLD ASSOCIATION OF TECHNOLOGY TEACHERS



department for  
children, schools and families



# Contents

Robotics.....	1
Contents .....	3
Introduction .....	5
FIRST .....	5
Prior experience.....	5
Briefing one – CAD and PLM.....	6
Briefing two – Pro ENGINEER modeling .....	8
Task one - Working with <b>Pro ENGINEER</b> . .....	8
Task two - Material properties .....	9
Task three - Assembly.....	14
Task four - Mechanisms .....	21
Task five - Part modeling.....	26
Task six - Sheet metal modeling .....	28
Task seven - Manufacture.....	31
Briefing three – Stability and structures .....	32
Task one - Four points of contact – Car .....	34
Task two - Three points of contact – Trike .....	40
Task three - Two points of contact – Bicycle .....	41
Task four - Two points of contact - Segway™ .....	43
Task five - Single point of contact – Unicycle .....	46
Briefing four – Electric motors.....	49
DC motor and servo motor .....	49
DC geared motor.....	50
RC Servo motor .....	53
Servo feedback for DC motor.....	56
Briefing five - Mechanisms.....	57
Task one - Gearing.....	57
Task two - Torque .....	61
Briefing six - Movement and control.....	69
Task one - Wheels .....	70
Task two - Friction .....	72
Task three - Inertia .....	76
Task four – Turning .....	79
Task five - Inclines.....	87
Task six - Centre of gravity.....	91

Briefing seven - Systems and control .....	95
Briefing eight - Engineering drawings .....	97
Task one - Creating a drawing .....	98
Task two - Adding dimensions .....	102
Task three – Pictorial view .....	103
Task four - Adding notes .....	104
Briefing nine – presentation images.....	107
Task one - Getting started.....	107
Task two - Initial render .....	108
Task three – Final render .....	110
Extension one – Position encoding.....	112
Extension two – Structural analysis .....	113
Appendix one - Web links .....	114
Appendix two – Servo motor specification .....	115
Appendix three – DC geared motor specification.....	116

## Introduction

This project will provide you with the background information, knowledge and understanding needed to use **Pro|ENGINEER** to design, assemble and simulate a virtual robot. While the context used for these materials is the **FIRST Tech Challenge** competition, teams entering other robotics competitions including **FIRST Robotics Challenge** should also find the exercises useful.

## FIRST

**FIRST** (For Inspiration and Recognition of Science and Technology) was founded by inventor Dean Kamen to inspire an appreciation of science and technology in young people. Based in Manchester, the not-for-profit organization designs accessible, innovative programs to build self-confidence, knowledge and life skills while motivating young people to pursue opportunities in science, technology, engineering and math (STEM).

### FIRST Tech' Challenge

The **FIRST Tech Challenge** (FTC) offers high school students the traditional challenge of a **FIRST** competition, but with a more accessible and affordable robotics kit.

- Design, build and program robots
- Use a modular robotics platform
- Apply real-world math and science concepts
- Develop problem-solving, organizational and team-building skills
- Compete and cooperate in alliances at tournaments
- Qualify for scholarship funds

FTC makes it easy and fun to create and test innovative designs and explore robotics concepts.

### FIRST Tech Challenge – Robot design

Full details on the current FIRST Tech Challenge can be found on the official **FIRST** web site at: <http://www.usfirst.org/roboticsprograms/ftc/default.aspx?id=968>

## Prior experience

The techniques in this document range from introductory to advanced. It is assumed you have completed the introductory course in **Pro|ENGINEER** and are familiar with part modeling and simple assembly. Any intermediate and advanced techniques will be taught step by step with screen snapshots and tools fully illustrated.

Teams wanting to improve their skills with **Pro|ENGINEER** can access comprehensive web-based training resources at <http://first.ptculms.com>. Team members can register and access training courses specific to modeling, assembly, detailing and mechanisms.

This will help strengthen their skills and should be used in parallel to this document



**Note:** You should keep the Quick Reference document close whenever you are using Pro|ENGINEER. It lists most of the toolbars, navigation and mouse operations in an easy to find format. This is linked from the home page in the internal browser.

## Briefing one – CAD and PLM

Engineering and technical teams generate much of an organizations' most valuable intellectual property, including customer requirements, engineering calculations, detailed CAD designs, and product documentation. To work efficiently this information must be shared with other departments but also protected. Managing this and other related information can be very difficult.

PTC can help with Product Development Systems (PDS) which offer a number of linked functions including;

1. Create product information
2. Communicate product information to different audiences
3. Collaborate regardless of location
4. Control product development processes
5. Configure product content



At the heart of a PLM system is a database or vault containing all the information related to a product.

PTC provides all **FTC** teams with access to Windchill ProjectLink or Product Point to help you develop your robot designs.

If your team is planning to use Windchill you should now familiarize yourself with **Project Link** or **Product Point** before continuing with the rest of these exercises.

Once your team has been invited to the Windchill project you can access a wide range of materials to help them make best use of PLM to manage their work including:

Introductory video: [http://www.ptc.com/appserver/wcms/replay/index.jsp?&im\\_dbkey=97874&im\\_language=en](http://www.ptc.com/appserver/wcms/replay/index.jsp?&im_dbkey=97874&im_language=en)

Quick start: [http://www.ptc.com/WCMS/files/99574/en/Quick\\_Getting\\_Started\\_Guide.pdf](http://www.ptc.com/WCMS/files/99574/en/Quick_Getting_Started_Guide.pdf)

Support videos: [www.ptc.com/go/firstworkshops](http://www.ptc.com/go/firstworkshops)

Full guide: [http://www.ptc.com/WCMS/files/77175/en/Getting\\_started\\_guide.pdf](http://www.ptc.com/WCMS/files/77175/en/Getting_started_guide.pdf)

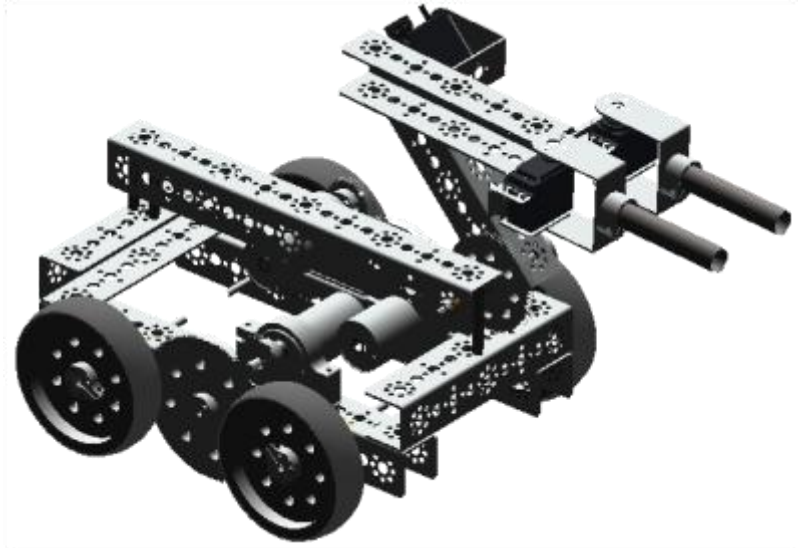


**Note:** only teams formally registered with **FIRST** have automatic access to Windchill Project Link or Product Point. Schools working on other robotics challenges can find out how they can access PLM from PTC by contacting [schools@ptc.com](mailto:schools@ptc.com)

This briefing is now complete.

## Briefing two – Pro|ENGINEER modeling

In this section we will show you how to open a robot modeled in **Pro|ENGINEER** and interrogate how it was assembled. This will help later when you are using mechanisms to simulate robot movement.



The model used here is constructed from the parts available in the standard Tetrix™ kit available from PitsCo™ (Part #: W39108) <http://www.pitsco.com/>

### Pro|ENGINEER CAD files.

The kit of parts for the FIRST Tech Challenge has been modeled using **Pro|ENGINEER**. You or your teacher can download these from:

<http://www.catalogds.com/db/service?d=first&c=browse> or [www.ptc.com/go/firstkop](http://www.ptc.com/go/firstkop)

The models needed for the exercises in this document are different from the parts kit mentioned above and are provided in ANSI or ISO units in each team's Windchill project. This is the preferred way to access the **FTC\_Robot** folders and files.



**Note:** Teams and individuals who do not have access to Windchill PLM can download the **FTC\_Robot** folder zip from:

<https://pds.ptc.com/Windchill/netmarkets/jsp/project/view.jsp?oid=project%7Ewt.projmgmt.admin.Project2%3A183631245&u8=1>

Extract the zip file structure to your work area on your local hard disk or local area network.

### Task one - Working with Pro|ENGINEER.

**Pro|ENGINEER** is very powerful and as you work a wide range of file types will be created. Unless these are properly managed you may not be able to find your work or your models may not regenerate properly next time you open them.



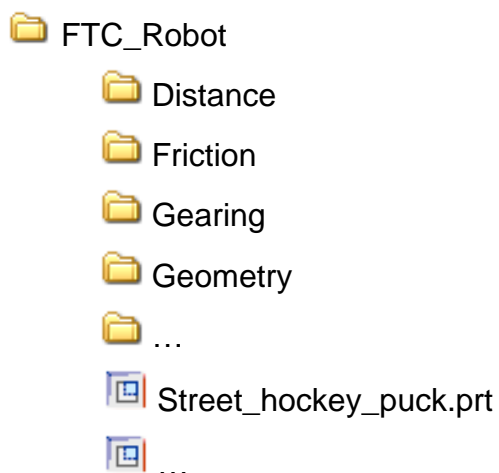
There are two things you can do to prevent these problems.

- Set the **working directory** every time you start work on a model. Throughout this project guide we assume you have set the working directory as the location of the PTC robot; e.g. **FTC\_Robot** or one of the sub-folders.
- Save any changes you make to models regularly, especially when prompted and whenever there is a save option in a dialog. These include; parts, assemblies, analysis playbacks, colors and appearances, materials, etc.

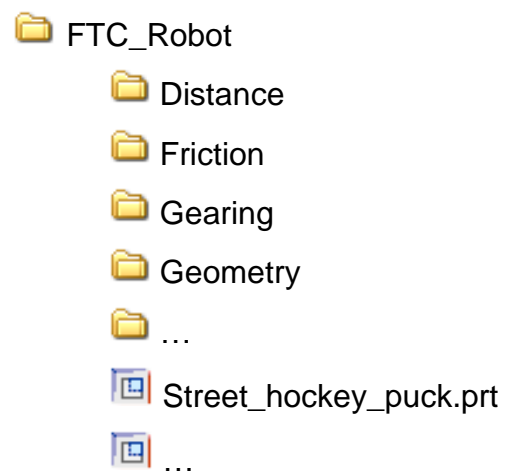
## Tutorial files

There are two sets of sample files depending whether you wish to use ISO or ANSI standards.

### FTC\_ISO



### FTC\_ANSI



1. Choose which standard you will use and access the files from your team's Windchill project. Teams without access to Windchill should have extracted the files to their working directory.

From this point on this document assumes you know which standard you are using and whether you are accessing the sample files from Windchill or a local working directory.

## Task two - Material properties

The exact size, mass and shape of all objects (parts) will need to be taken into account when designing your robot. For example, we will show you how to interrogate the street hockey puck from the FTC 2008 season.



**Street hockey puck**



**Air ball**

Pro|ENGINEER can accurately represent the behavior of models providing the correct properties are applied to parts. Street hockey pucks are injection molded from the thermo-plastic PVC and a Pro|ENGINEER model of the puck has been created.



The FIRST Tech Challenge 2008 focused on street hockey pucks. What properties are we most interested in? It helps to list the actions you want to carry out on the object then list the properties that will affect that action.


Action	Material property
Slide	Coefficient of friction, surface texture
Grip	Coefficient of friction, surface texture
Lift	Mass (density), inertia

In the parts folder you have a Pro|ENGINEER model of the ‘air ball’ used in the 2009-10 season. Once you have completed the next section you should use the techniques to check the mass of the ball.

### Material properties



You will now be shown how to interrogate the Pro|ENGINEER model of street hockey puck to find out about its properties.

1. Start Pro|ENGINEER
2. The navigator window on the left of the screen will be displaying  folders.
3. Navigate to until you can see the **FTC\_Robot** listed in the browser window.
4. In the Browser window, right mouse click on the **FTC\_Robot** folder and select **Set working directory**.
5. In the Navigator panel, click on  Working Directory and you should see the contents of the **FTC\_Robot** folder listed in the browser panel confirming the correct folder has been set.

6. Open the part **street\_hockey\_puck.prt**
7. In the top toolbar use these buttons  to turn off the visibility of datum information.
8. Use the middle mouse button to click on the model and drag to **rotate** it in the graphics window.
9. Move the mouse cursor over the middle of the model and roll the mouse wheel to **zoom** in and out.



Don't worry if the model disappears from view! The following two actions will sort things out and return you to viewing the model.

10. In the top toolbar click on  the **Saved views** tool and from the drop-down menu select **Trimetric**.
11. In the top toolbar click on  the **Refit** button.

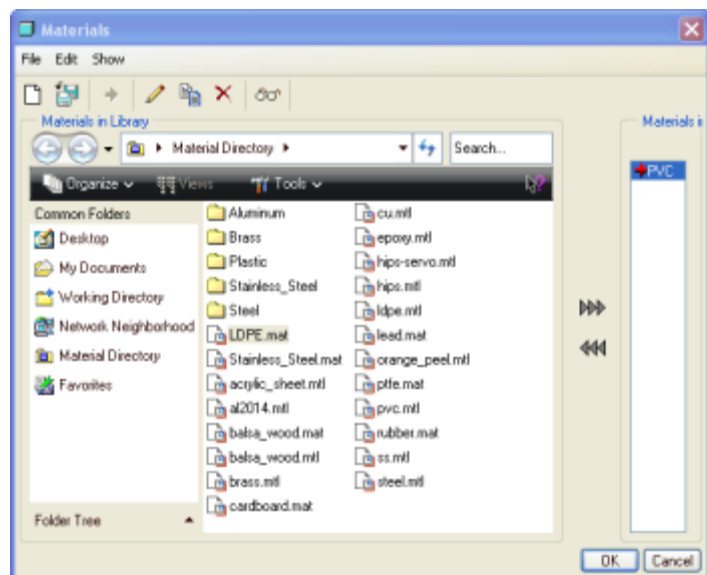
When viewing objects in 3D, we call these 'get you out of Jail buttons'!

It is easy to see the physical shape of the object, created using the modeling tools in **Pro|ENGINEER**. Not so obvious is the information about material properties and surface texture.

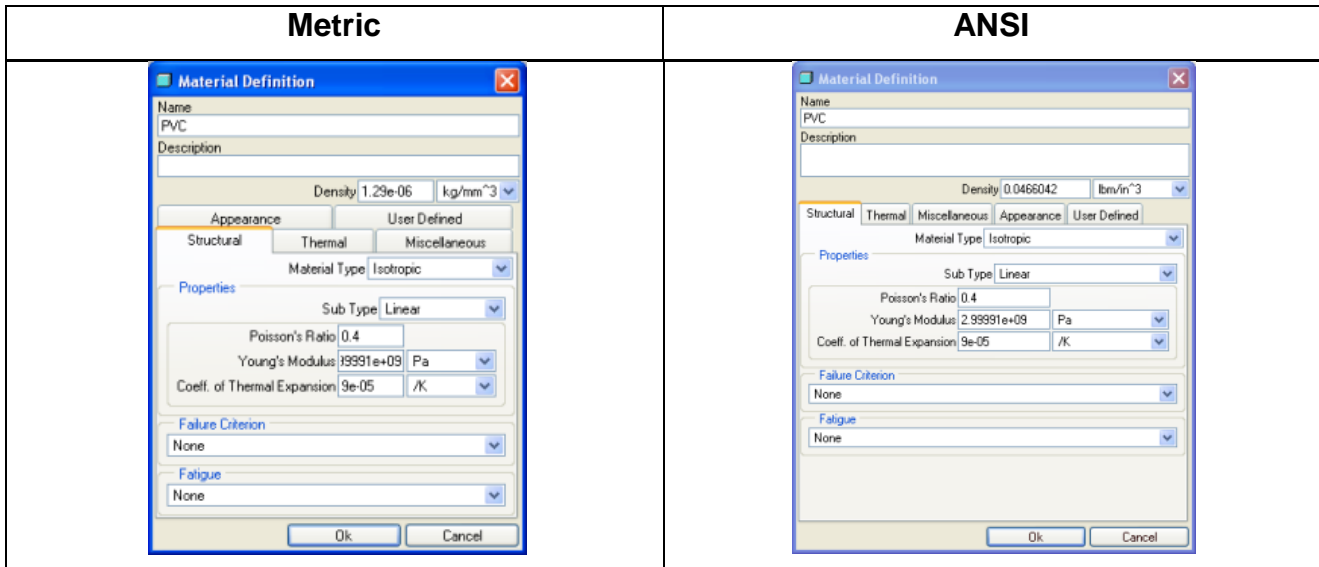
12. In the **Main** tool bar across the top of the screen click on **Edit > Setup** and from the Menu Manager on the right of the screen select **Material**.

In the column on the right of the Materials dialog you will see that **PVC** is selected for this part. What properties does **Pro|ENGINEER** understand for PVC?

13. In the Materials dialog, move the mouse over the **PVC** material in the right column, right click and from the floating menu select **Properties**.



The **Material definition** dialog will open.



- Click on each tab and you will see the many properties Pro|ENGINEER stores for each material.



At the moment this will mean very little to you but would become very important if you decide to become an engineer.

The physical properties we are most interested in here is the density, shown in the dialog above the tabs.

Metric	ANSI
Density = <b>1.29e-06 kg/mm<sup>3</sup></b> . This is telling us that a <b>cubic millimeter</b> of PVC has a <b>mass</b> of <b>0.000 00129 kilograms</b>	Density = <b>0.0467 lb/in<sup>3</sup></b> This is telling us that a <b>cubic inch</b> of PVC has a mass of <b>0.0467 pounds</b>


One of the best sites on the web for information on materials is [www.matweb.com](http://www.matweb.com) A search for **PVC** on the site lists a confusing number of entries but if we select '**Overview for PVC, Molded**', MatWeb lists the average density as:

Metric	ANSI
Density = <b>1.29 g/cc</b> This is telling us that a <b>cubic centimeter</b> of PVC has a <b>mass</b> of <b>1.29 grams</b>	Density = <b>0.046 lb/in<sup>3</sup></b> This is telling us that a <b>cubic inch</b> of PVC has a mass of <b>0.046 pounds</b>

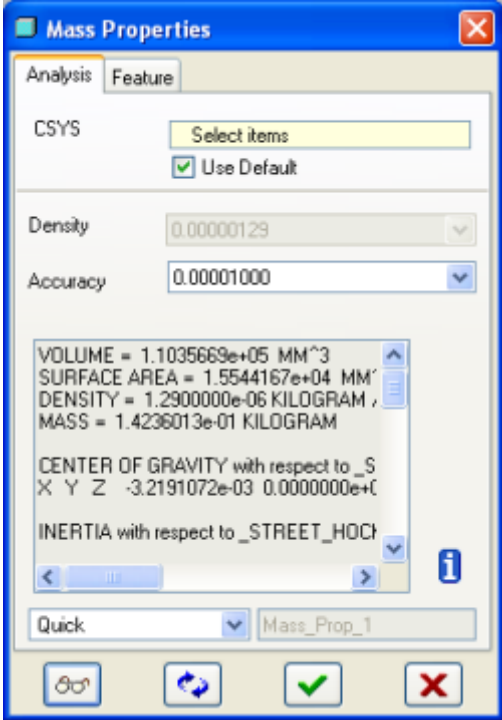
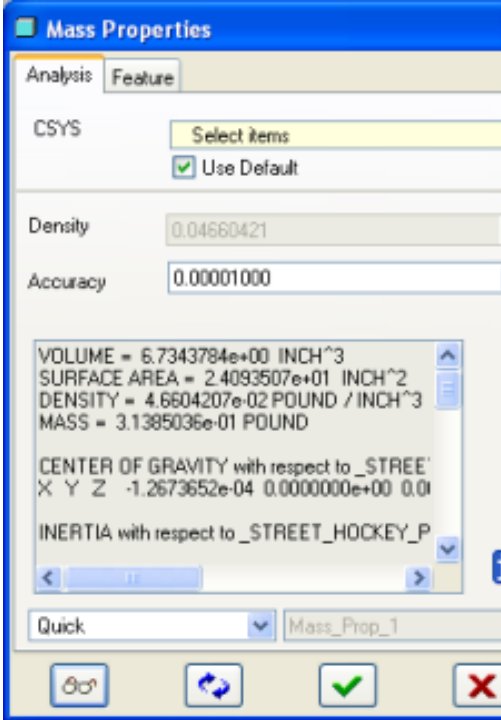
- Click  to close the **Material Definition** dialog, then  to close the **Materials** dialog, then **Done** to close the **Menu Manager**.

### Material properties applied to FTC

Because Pro|ENGINEER stores information about PVC and this material has been assigned to the street hockey puck, we can ask Pro|ENGINEER to analyze the model and measure the mass of the puck.

16. The **Street\_hockey\_puck.prt** should still be open on screen.
17. In the top toolbar click on **Analysis** then **Model** and **Mass Properties**
18. In the Mass Properties dialog click on  to perform an analysis of the model.



Metric	ANSI
	
<p>The results of the analysis are:</p> <ul style="list-style-type: none"> <li>Volume = 110,000 mm<sup>3</sup></li> <li>Surface area = 15,500 mm<sup>2</sup></li> <li>Density = 0.000 0129 kilograms / mm<sup>3</sup></li> <li><b>Mass = 0.142 kilograms</b></li> </ul>	<p>The results of the analysis are:</p> <ul style="list-style-type: none"> <li>Volume = 6.73 in<sup>3</sup></li> <li>Surface area = 24 in<sup>2</sup></li> <li>Density = 0.0466 pounds /in<sup>3</sup></li> <li><b>Mass = 0.31 pounds</b></li> </ul>

## Challenge

Use the same technique to measure the mass of other components including the complete robot model - **PTC\_robot.ASM**

You will need the mass of the robot later to work out the coefficient of friction when simulating things like wheel-spin.

## Task three - Assembly

Assembly using **Pro|ENGINEER** is very straightforward. Components are added to the model and 'constraints' used to link components. **Pro|ENGINEER** uses two main types; **assembly constraints** are most often used to fix two components together and **mechanism constraints** allow known movement between components.

If you would like to learn about assembly constraints and mechanical connections take the following **Online Courses** in PTC University Coach: <http://first.ptculms.com>


- **Assemblies > Bottom Up Design**
- **Introduction to Mechanisms > Start with Overview and Creating Mechanisms.**

In addition, PTC's schools curriculum contains projects covering aspects of mechanisms including:

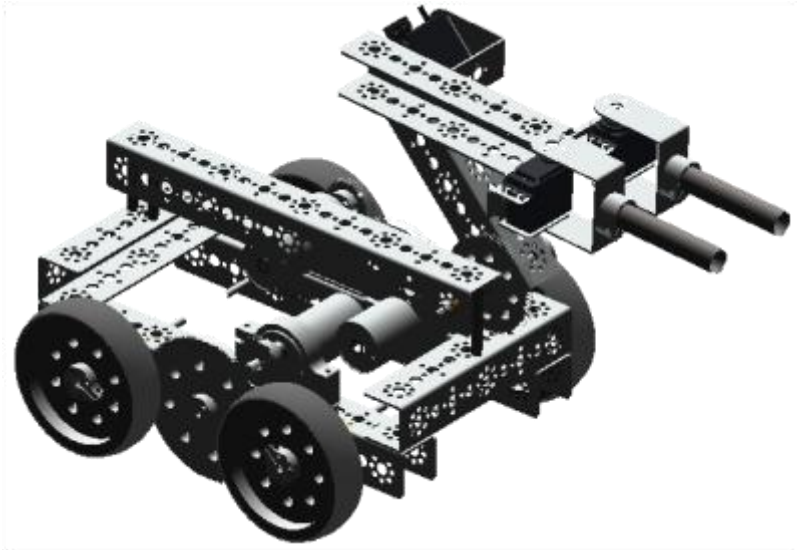
- **Cam Toy/ Wind Sculpture** (assembly, mechanism joints and cam connections)

## Assembly applied to FTC

You or your teacher should already have downloaded the PTC robotics sample files and made them available in your work area in a folder called **FTC\_Robot**.

1. Start **Pro|ENGINEER**.
2. Browse to find the **FTC\_Robot** folder and set it as the **Working Directory**.
3. Open the assembly **PTC\_Robot.asm**.
4. If necessary make datum information invisible – In the Main toolbar across the top of the **Pro|ENGINEER** screen toggle these buttons .

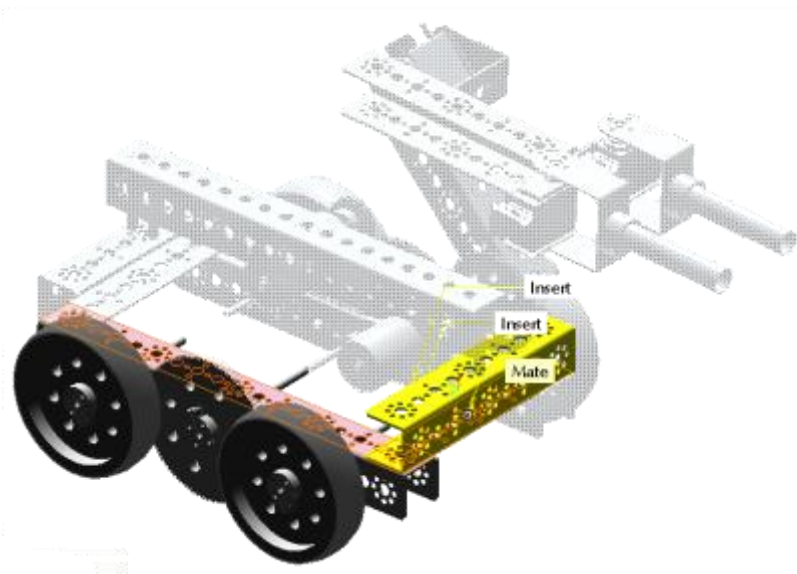
You will see a robot similar to the one shown here.



5. In the model tree, locate the first sub assembly entry **DRIVE\_TRAIN\_SUB.ASM** and click on the **+** to the left to see the components used in the sub assembly.

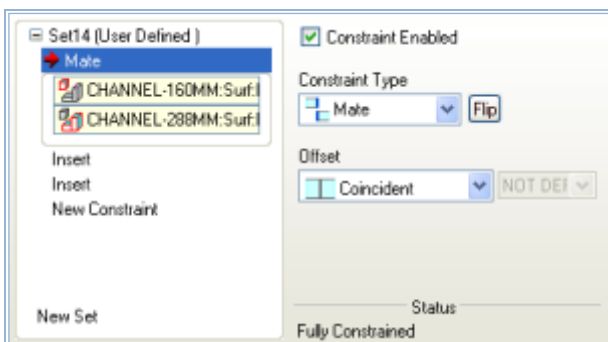
6. Right click on the entry **CHANNEL-160MM.PRT** and choose **Edit Definition**.

The model will change with many components grayed out and labels listing the assembly constraints used to add the channel component to the model.



7. In the Dashboard open the **Placement** tab.

The placement tab list the three constraints, mate and two inserts.



8. Click on each entry in turn and you will see the constraint highlighted on the model.


In the illustrations above, the Mate has been selected in the Placement panel and the surfaces are highlighted in red on the model.

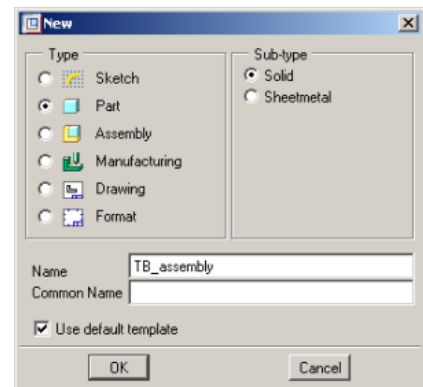
9. In the dashboard, click on  to close the placement dashboard.

10. In the **File** pull-down text menu select **Erase** then **Current** to close the current model and remove it from the computer's volatile memory.


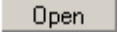
### Assembling robot parts

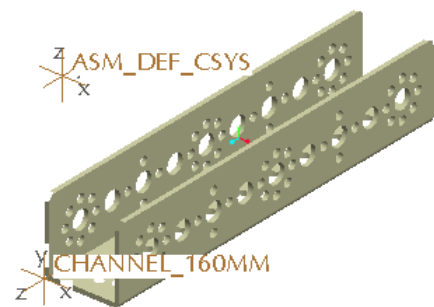
This section will show you how to add components to a static model with assembly constraints.

11. **Pro|ENGINEER** should still be running with the **FTC\_Robot** folder set as the **Working Directory**
12. In the **File** menu select **New** and the new file dialog will open.
13. In the dialog change the file **Type** to **Assembly**.
14. Type in a name for your assembly then click .



**Note:** you cannot use spaces or reserved characters in **Pro|ENGINEER** filenames.

15. An empty assembly is created.
16. In the assembly toolbar on the right of the screen, select  to add a new component to the assembly.
17. In the list of files select channel-160mm.prt then click .
18. The component will appear in the modeling window of **Pro|ENGINEER** colored gray and the assembly dashboard will open across the top of the screen.

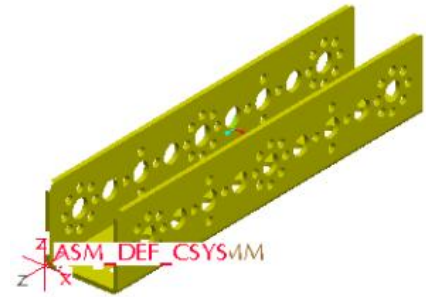


The first component in an assembly is normally fixed in position as a reference for other components to be added.



- In the assembly dashboard, open the constraint type list shown outlined in red above, and select **Default**.


The component will be located in the assembly by lining up all three axes of the assembly and part coordinate systems.



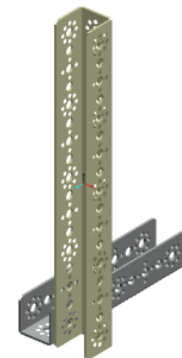
**Note:** The channel component has turned a mustard color to show it is **fully constrained** in the assembly. This is confirmed in the dashboard.

### Adding another component

As you have just seen, assembly involves selecting pairs of surfaces and creating constraints between them. In the following example the two pieces of channel will have three constraints; a **mate** and two **inserts**.


- In the assembly toolbar on the right of the screen, select  to **add a new component** to the assembly.
- In the list of files this time select **channel-288mm.prt** then click .

The component will appear in the modeling window of Pro|ENGINEER colored gray and the assembly dashboard will open across the top of the screen.



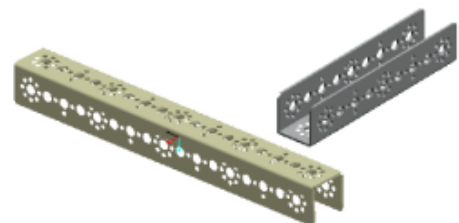
### COMPONENT PLACEMENT CONTROLS


Allows reorientation of components during placement


COMPONENT DRAG	»	CTRL + ALT + 
SPIN	»	CTRL + ALT + 
MOVE	»	CTRL + ALT + 

When adding components to an assembly it is good practice to orient the new part close to where it will be located. This is done using component placement controls explained in the Quick reference card.

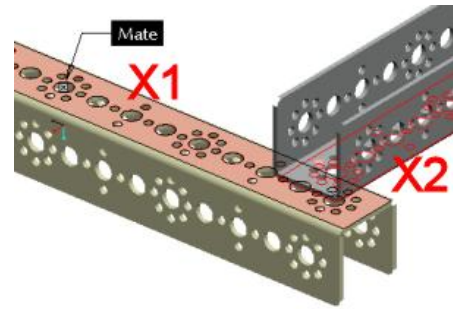
- Use the component placement controls to **rotate** and **move** the new component until it looks like this.





23. **Left click**  on the top flat surface of the long channel **X1**.

24. **Left click**  on the underneath flat surface of the short channel **X2**.


The long channel will move to **mate** the two surfaces




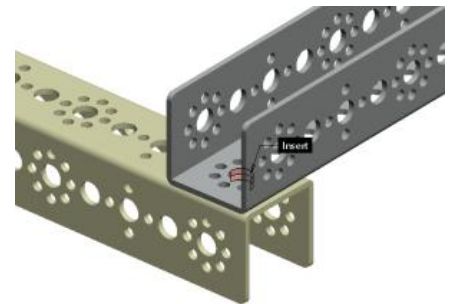
 **Note:** You can select geometry ‘through’ the model using the **Right-click**  filter.

Right click repeatedly in the area you are interested in until the geometry you wish to select pre-highlights in light blue. Without moving the mouse left click to select the geometry.

This takes a bit of practice but is much quicker than constantly rotating the model until you can see the geometry you want to select.

25. **Left click**  to select the cylindrical surfaces in one of the large holes at the end of a channel part.

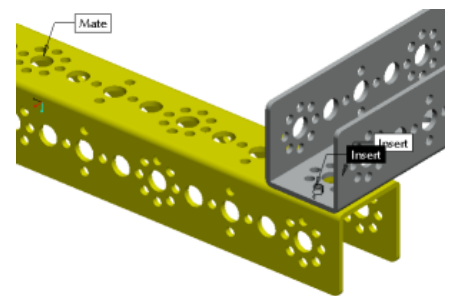
26. **Left click**  to select the cylindrical surfaces in the hole in the end of the other channel




As two cylindrical surfaces were selected, **Pro|ENGINEER** will correctly guess and apply an insert constraint.

27. Repeat the process to add an insert constraint between a pair of small holes in the ends of the two channels.

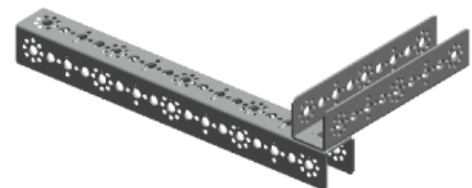
Notice the long channel has turned mustard color to denote it is fully constrained in the assembly.



28. In the Assembly dashboard, click  on to finish placing the long channel.


The assembly dashboard will close and the new component entry is added to the model tree.

29. Save your model



## Adding a bearing

A bearing should be fixed in the assembly ready accept an axel later.

30. In the assembly toolbar on the right of the screen, select  to **add a new component** to the assembly.

31. In the list of files this time select **axle\_bronze\_bushing.prt** then click .

The component will appear in the modeling window of **Pro|ENGINEER** colored gray and the assembly dashboard will open across the top of the screen.


32. Use the component placement tools to re-orient the bush and position it next to the fourth large hole from the end of the long channel.

33. Select the cylindrical surface of the bush

34. Select the internal cylindrical surface of the large hole and an insert constraint will be created.

35. Select the side of the channel for one half of the mate.

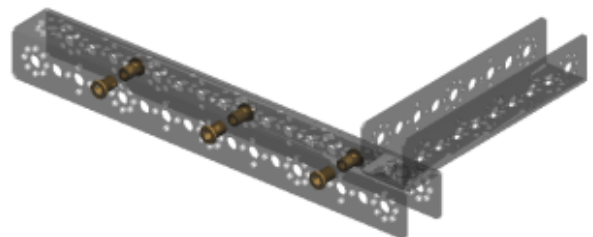
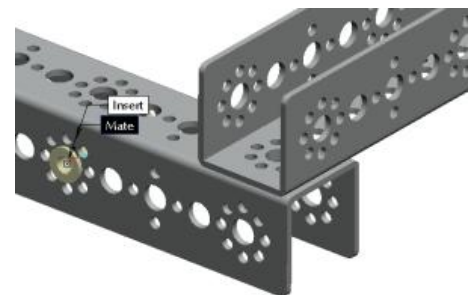
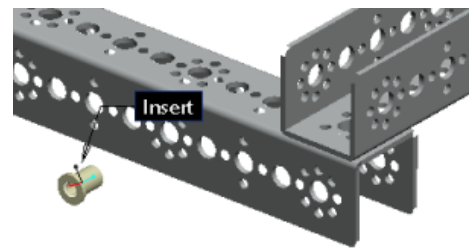
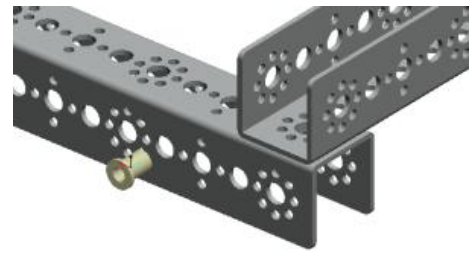
36. Zoom in to select the underside of the flange on the bush for the other half of the mate.

37. In the Assembly dashboard, click  on to finish placing the long channel.

The assembly dashboard will close and the bush component is added to the model tree.

38. Add five additional bushes in the locations shown here.

39. Save your model



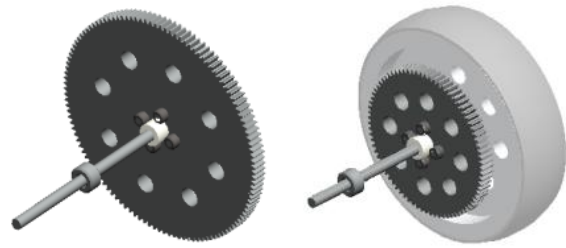
## Adding a moving component


Although mate, align and insert assembly constraints will allow movement, **Pro|ENGINEER** has mechanism constraints specifically designed for common mechanisms including pin, slider and cam.

Using mechanism constraints will support advanced techniques like motion analysis, simulation and mechanism dynamics.

In this section you will add one of two sub assemblies using mechanism connections.

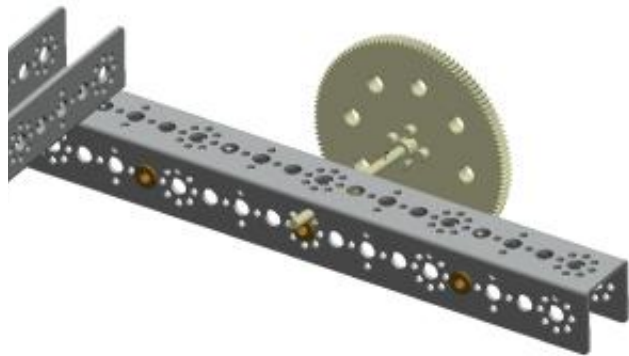
The first contains an idler gear on an axel; the other has a wheel, gear and axel.



40. In the assembly toolbar on the right of the screen, select  to **add a new component** to the assembly.

41. In the list of files this time select **large\_gear\_sub.prt** then click .

42. Use the component placement tools to re-orient the sub-assembly as shown here.



43. In the assembly dashboard open the mechanism list and select **Pin**.

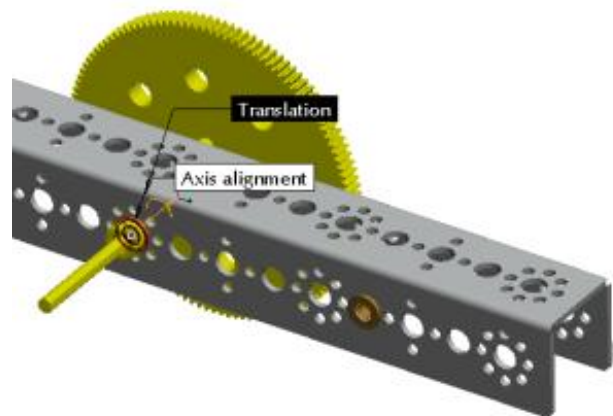


44. In the assembly dashboard open the Placement drop-down panel and leave it open for the next few steps.

45. **Left click**  to select the inside of the middle bush.

46. **Left click**  to select the outside of the axel.

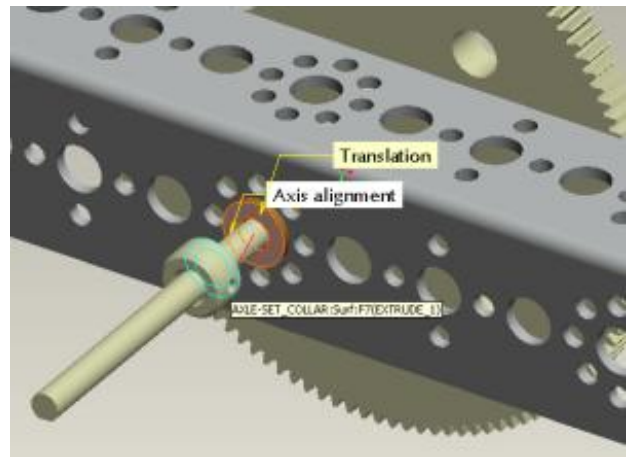
47. The axel will be centered in the bush.




The sub assembly will be free to move through the bush. The next step adds 'translation' constraints to stop axial movement.

48. Select the two surfaces shown here for the translation references.

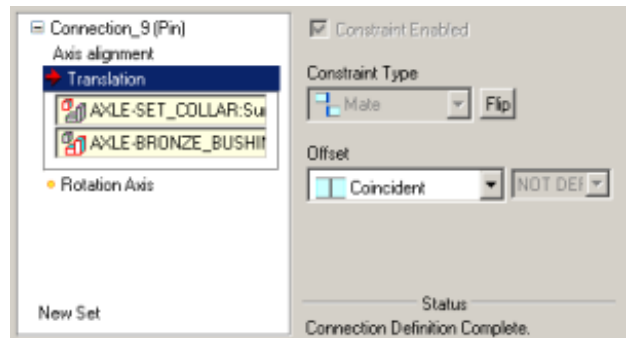
The gear axel sub assembly will move into position and still allow rotation



49. The placement panel should now be displaying the surfaces used to create the pin connection.

50. In the dashboard click on  to complete the mechanism connection.

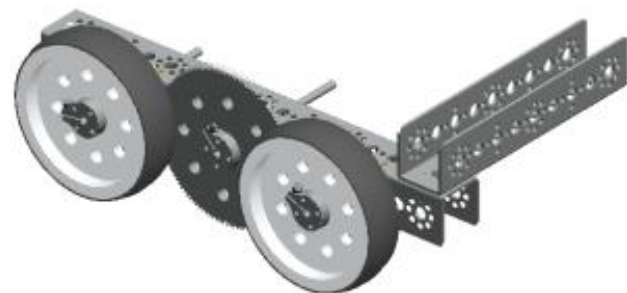
51. Save your model.



52. Use the same technique to add two instances of the sub assembly **Wheel\_large-sub.asm** creating a pin connection.

53. Save your model.

Later you will be taught how to create gear connections.



### Task four - Mechanisms

A powerful set of tools are available in Pro|ENGINEER to build, operate and test mechanisms.

You can learn how to create mechanical connections by taking the following **Online Courses** in PTC University Coach: <http://schools.ptculms.com/>

- **Introduction to Mechanisms**

In addition, PTC's schools curriculum contains projects covering aspects of mechanisms including:

- **Cam toy/Wind sculpture** (assembly, mechanism connections, cam connection)
- **Mars rover** (assembly, mechanism connections, cam connection)



## Mechanical connections applied to FTC


Pro|ENGINEER has a wide range of mechanical connections for modeling realistic mechanisms.

1. The **PTC\_Robot.asm** should still be open on screen.

## Testing mechanisms

You can try out the kinematics (movement) of mechanical connections (and mechanisms) simply by dragging them on screen.

2. In the top toolbar select  the **Drag components** tool.
3. Move the mouse cursor over the perimeter of one of the drive wheels and **Left click**  .

You will see a small diamond shaped target  on the edge of the wheel.

4. Move the mouse to rotate the wheel and notice how the wheel rotates.

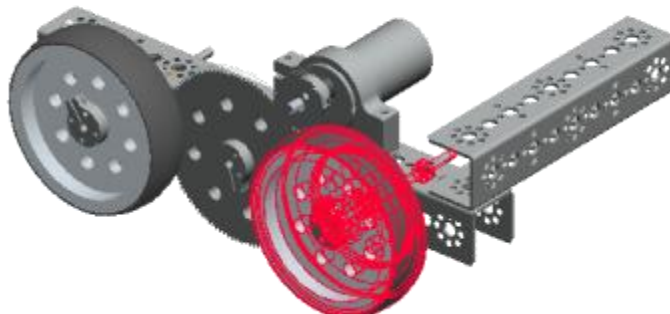
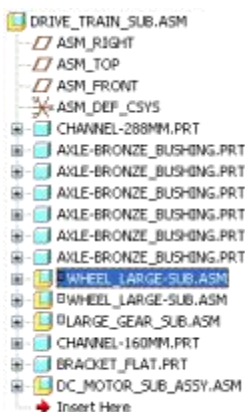
You will also be able to see gears along one side rotate according to the gear connections. You will learn how this is done later.

5. When you have finished dragging the wheel, go to the **Drag** dialog and click **Close**.

## Interrogate mechanical connections

6. In the model tree for the **PTC\_ROBOT.ASM**, locate the first entry named **DRIVE\_TRAIN\_SUB.ASM**, Right-click  and, from the floating menu, select **Open**

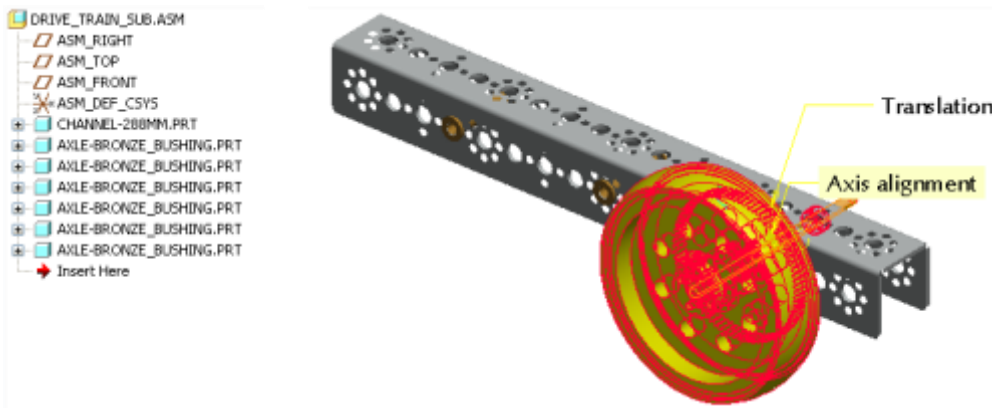
7. In the model tree for the sub assembly locate the sub assembly **WHEEL\_LARGE-SUB.ASM**



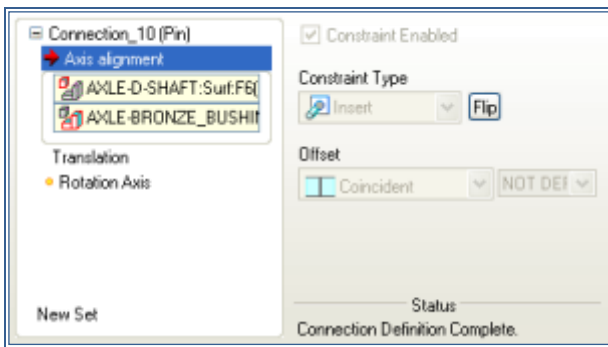
**Note:** There is a small rectangle in front of the entry. This shows the sub assembly is 'packaged' and can move in at least one axis. This is why you could rotate the wheel by dragging in the model.

8. Right click over the entry in the model tree and from the floating menu select **Edit Definition**.

9. In the assembly dashboard open the **Placement** panel.



Entries below this in the model tree will be suppressed and you will see the Pin connection for this sub assembly highlighted on the model and listed in the Placement tab of the dashboard.




10. **Left Click**  on each entry in the Placement tab and notice the geometry highlighting on the model.

11. In the assembly dashboard left click on  to close then **close** the **sub assembly** to get back to the PTC\_Robot.

12. **Close** the PTC\_Robot.asm

13. Open the **File** pull-down menu, Left click  **Erase** and from the floating menu click on **Erase not displayed...**

14. Left click  on **OK** to close the dialog.

You should now have a clear idea how the concepts you learned in Coach have been applied to the PTC Robot.

### Creating gear connections

Pro|ENGINEER has sophisticated mechanism simulation tools. The next section will show you how to create gear connections between the driving wheels of the robot.



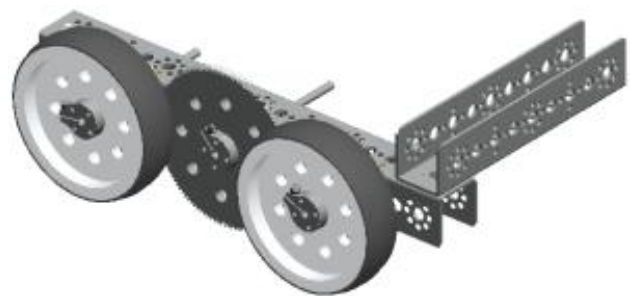
**Note:** Gear connections can only be created on rotating axles created using **pin** connections.

The robot model you will be working on has gear pairs with 120 teeth and 80 teeth. Pro|ENGINEER lets you enter the gear information in a number of ways including pitch circle diameters or as a ratio. This example will use a ratio of 3:2 derived mathematically as follows.

$$\frac{120}{80} = \frac{3}{2}$$

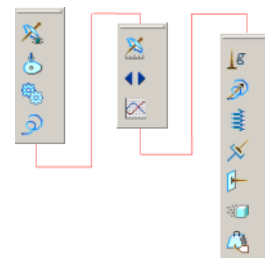
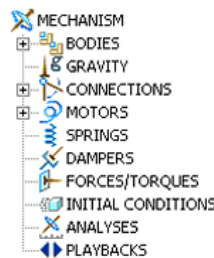
- Open the assembly model you created earlier or TB\_ASSEMBLY.ASM

The three axels have pin connections and are able to rotate independently. You will now add gear connections to link them together.




- In the main toolbar open the **Applications** pull-down text menu and select **Mechanism**

An extra mechanisms section will appear in the model tree and the mechanism toolbars open on the right of the screen.

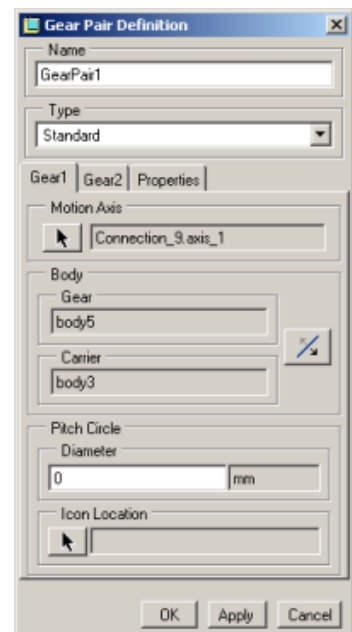
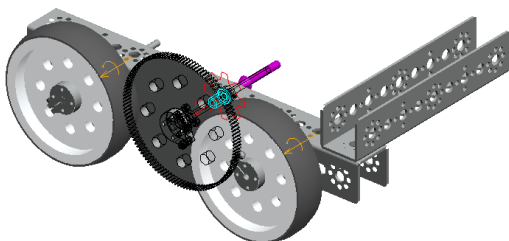


- The model will also be showing the pin connections on each of the axels.

- In the mechanisms toolbar on the right of the screen click  to create a new **Gear** connection.

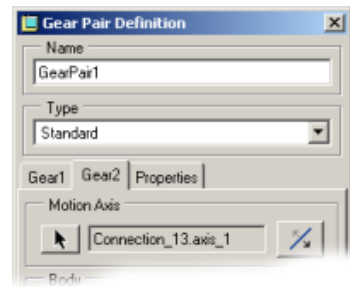
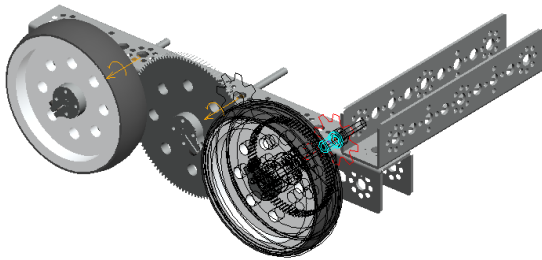
The gear pair definition dialog opens with the **Gear1** tab selected.

- On the model, click on the pin connection symbol on the central gear axis.

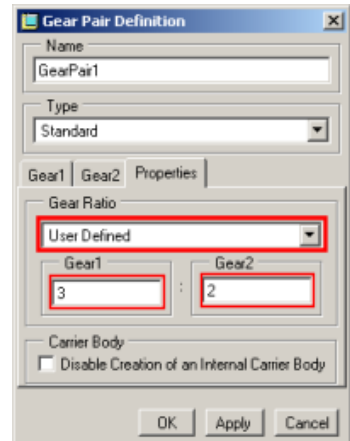




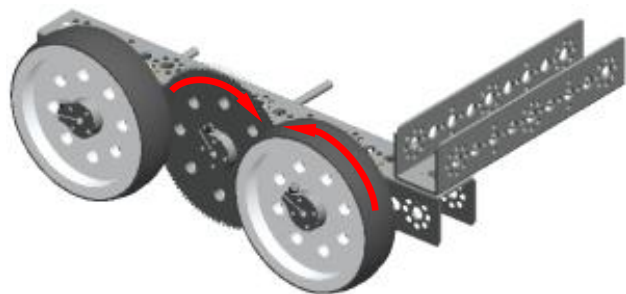
- In the dialog select the **Gear2** tab then select the pin connection axis on one of the wheel axels.



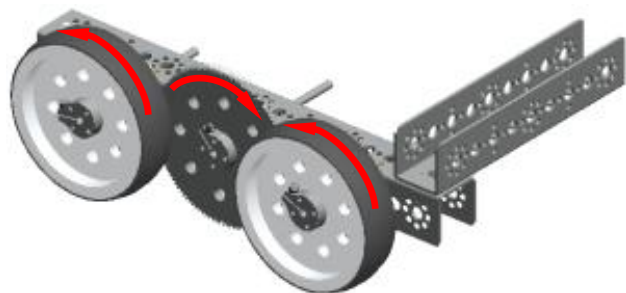
- In the dialog select the **Properties** tab.
- Change the Gear ratio option to **User Defined**
- In the **Gear1** field enter the value **3**
- In the **Gear2** field enter the value **2**
- Click OK to close the Gear Pair Definition dialog and complete the gear connection



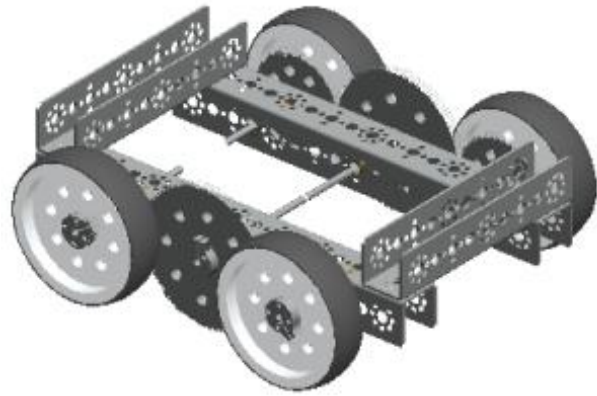
- Test** the mechanism by holding Ctrl+Alt and dragging the central gear.  
The wheel should rotate at a speed governed by the gear ratio.



- Add another **gear pair** between the central gear and the other wheel.
- Test** the mechanism works correctly.
- Save** the sub-assembly.



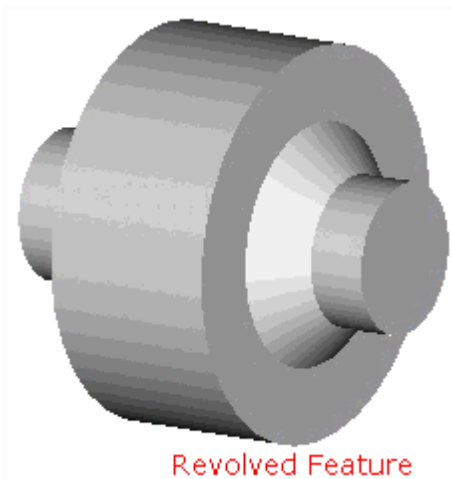
A complete robot chassis can now be created by starting a new assembly then adding two instances of the sub-assembly.



### Task five - Part modeling

You can learn how to model components by taking the following **Online Courses** in PTC University Coach: <http://schools.ptculms.com/>

- **Modeling > Basic Modeling and Manipulating Sketches and Rounds and Basic Modifications.**

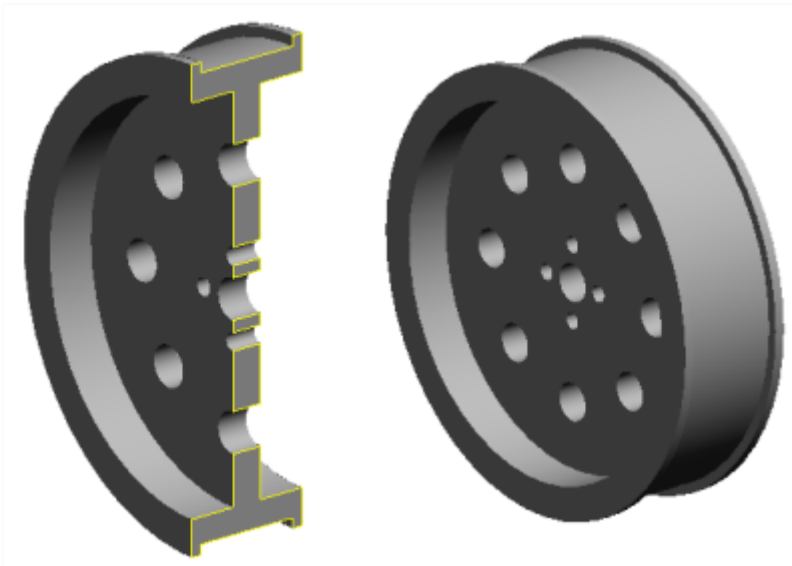


In addition, PTC's schools curriculum contains projects covering aspects of part modeling including:

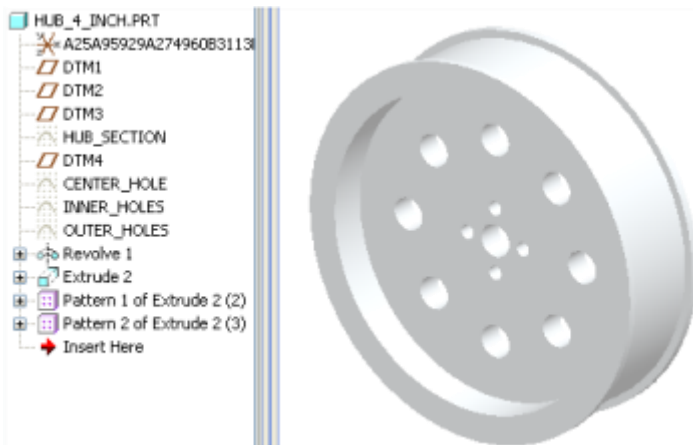
- **Sports bottle** (modeling parts)
- **Starchaser Rocket** (modeling parts)
- **RP Car** (modeling parts, teams modeling concurrently)

### Solid part modeling applied to FTC

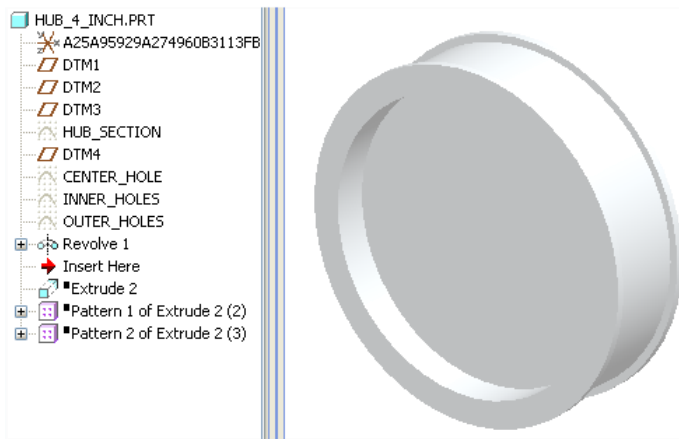
We will explore the wheel hub component to learn how Pro|ENGINEER models parts.



1. Make sure the **FTC\_Robot** folder is set as the **Working Folder**.
2. Open the part **hub\_4\_inch.prt**
3. Look at the **model tree** and you will see the 'history' of how the wheel hub was created.

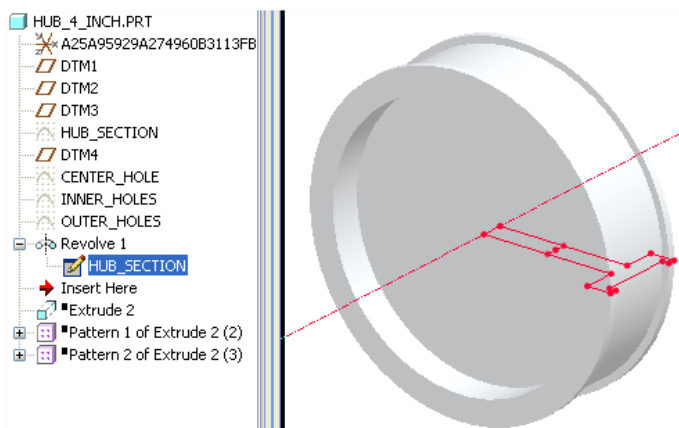


4. Drag the **Insert Here** flag to below the **Revolve 1** entry.



The model will update to show the model at this stage with entries below the Insert flag suppressed. The small black squares next to these entries show they are suppressed and not being shown in the model.

- Click on the + to expand the revolve feature then click on the internal sketch **HUB\_SECTION**.



The section sketch and center line used to create the revolve will be highlighted on the model.

- Drag the **Insert Here** flag down one entry at a time and investigate how each part of the model was created.
- Repeat the previous section on the `dc_motor_mount.prt` to see alternative features in use.
- Close** all models and **Erase not displayed**.

## Task six - Sheet metal modeling

PTC University Coach contains tutorials on how to model components using the sheet metal tools in Pro|ENGINEER. The **Online Courses** can be found at:

<http://schools.ptculms.com/>

- **Sheet metal design > Basic Concepts and Attached Walls.**



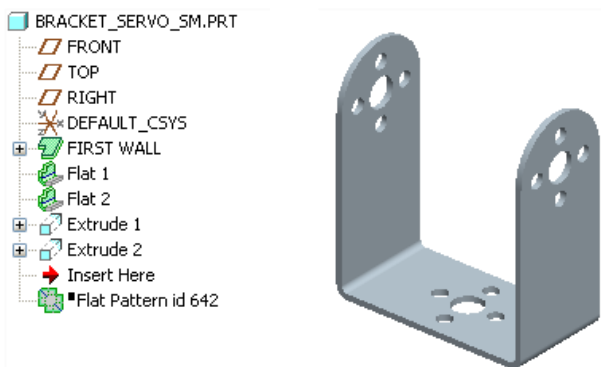
In addition, PTC's schools curriculum contains projects covering aspects of sheet metal modeling including:

- **Periscope** (Sheet metal conversion from solid - Rip)
- **Deodorant bottle** (Sheet metal adding 'walls', nets)

### Sheet metal modeling applied to FTC

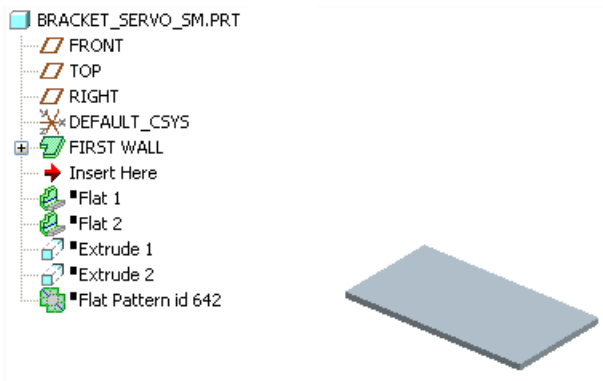
To understand how sheet metal tools were used to create components for the Tetrix™ kit we will look briefly at the one of the servo brackets.

1. Make sure the folder **FTC\_Robot** is set as the **Working Directory**
2. Open the part **bracket\_servo\_sm.prt**




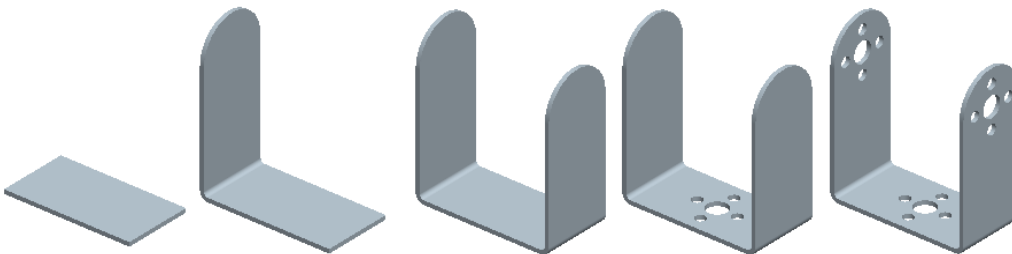
Notice the **Insert Here** flag is not at the bottom of the model tree? This will be explained shortly. First let's look at the rest of the model tree.


3. Drag the **Insert Here** flag to below the **FIRST WALL** feature. The remaining features will be suppressed.



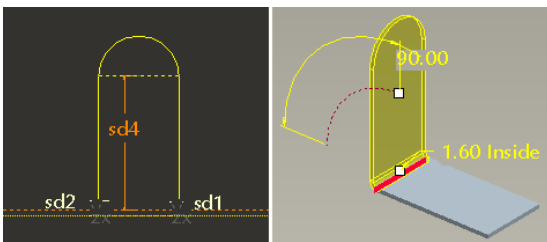
This is the first flat shape on which all the other shapes will be attached.



4. Drag the  **Insert Here** flag down one step at a time to see how the component was created.



5. In the model tree **Right-click**  on the **Flat 1** entry and from the floating menu choose **Edit Definition**.

The **Create Flat Wall** dashboard will open showing how the feature was created and the model will display the edge selected for the wall attachment. Open the **Shape** tab and you will see the profile sketch used to define the outline of the wall.



6. In the assembly dashboard left click on  to close
7. Finally, drag the  **Insert Here** flag to the bottom of the model tree.

The final entry in the model tree flattens the sheet metal component.



The fold/unfold process takes into account the material properties which **Pro|ENGINEER** stores in a 'bend table' for each type of material.

8. **Close** all models and **Erase not displayed**.

### Task seven - Manufacture

Note to reviewers - Teams can only use parts in the kit so this section is superfluous and will be deleted unless a good argument is made otherwise?

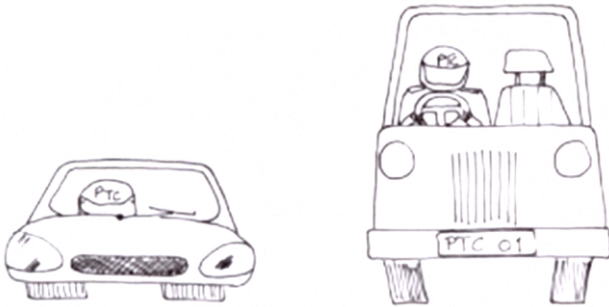
1. Full scale drawings > 2D print for manufacture templates
2. 2D CNC output – model > drawing > dxf > laser/slot drill CNC
3. 3D CNC output – model > stl > post process > 3 axis machining
4. 3D CNC output – model > Expert machinist 2.5D

This briefing is now complete.

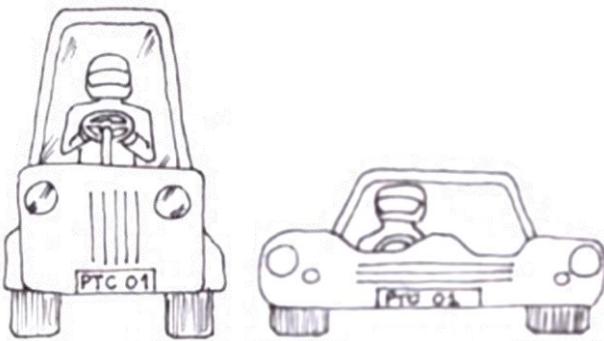
## Briefing three – Stability and structures

The centre of gravity of a model can be important for a number of reasons. A simple way of thinking about stability is the width of a model compared with its height.

Tall models are more likely to fall over.



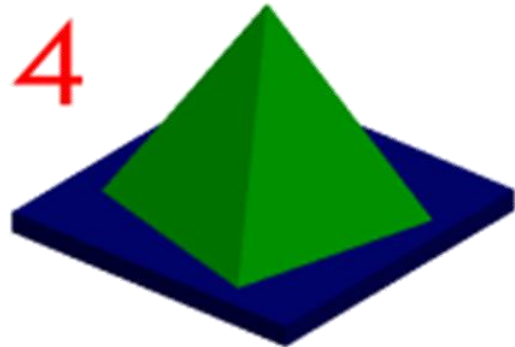
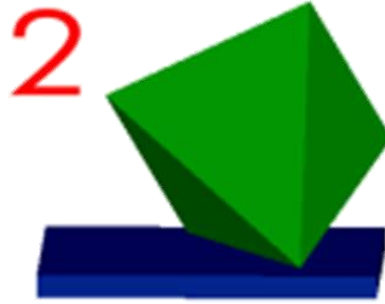
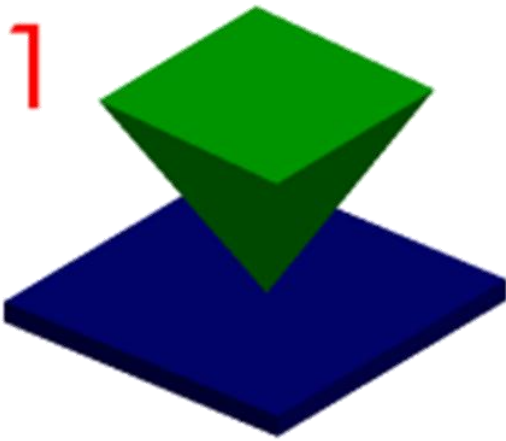
Making your design wider can also help prevent it from toppling over.



To understand the concept of stability, imagine a square based pyramid and how you might stand it on a surface. Below are four positions

1. The vertex
2. An edge
3. A triangular face
4. The square face





In which, if any, of the positions would the pyramid fall over?

1. Diagram one is the most unstable and would fall over in any direction.
2. Diagram two is the next most unstable and would fall over along an axis running through the edge resting on the base.
3. Diagram three is the first to be stable. The pyramid would not fall over.
4. Diagram four is also stable. The pyramid would not fall over.

These solids will be animated with exciting new techniques in **Pro|ENGINEER Wildfire 5.0**. For a sneak preview of what is possible look at:  
<http://www.youtube.com/watch?v=0GJHz7Jvad4>

### Stability in the context of FTC

In the context of FTC we will explore the stability of various simplified robot designs based on Tetrix™ components. This should help you apply the principles of stability to your **First** Tech Challenge designs.

## Task one - Four points of contact – Car

It is no accident that most cars have four wheels. Points of contact at all four corners with the center of gravity acting downwards in the middle provide a very stable vehicle.

A four wheel vehicle is inherently stable. An easy way to think of this is what would happen if you drop a car from a modest height? There is a good chance the vehicle will land and remain upright.

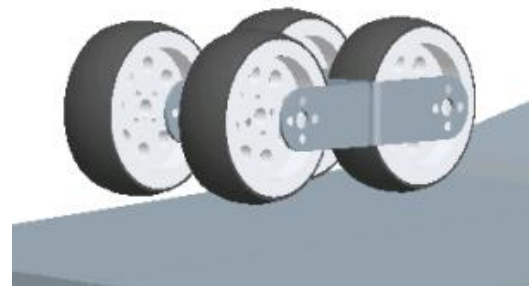


<http://livedesignonline.com/stagingrental/Vertigo.jpg>

### Drop-Test

A Pro|ENGINEER simulation has created with a four wheeled robot chassis a short distance above the floor. Four cam connections link the wheels to the floor. With gravity applied, the robot drops to the floor coming to rest after some bouncing.

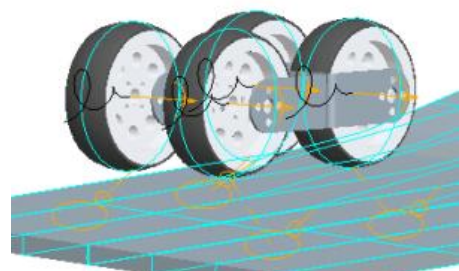
1. Start **Pro|ENGINEER**.
2. Set the working directory to **\FTC\_Robot\Stability\Fourcycle**.
3. Open the assembly **Stability-fourcycle.asm**
4. **Zoom** until the model looks like this.



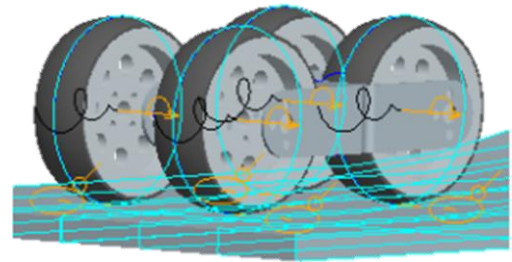
5. At the top of the **Pro|ENGINEER** screen open the **Applications** pull-down text menu and select **Mechanism**.



Look for the four cam connection symbols on the model plus the **light blue** lines and mesh to show cam surfaces. A mechanism tree will also appear below the normal model tree.



- In the mechanism tree, expand the **ANALYSES** sub heading then **Right-click** over **Drop\_test (DYNAMICS)** and from the floating menu select **Run**.



Pro|ENGINEER will mathematically apply gravity and friction to the model, and move the parts according to the forces acting on them.

You should see the model drop, bounce a little then come to rest like this.

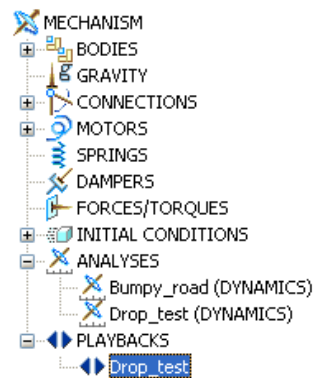


**Note:** The movement may not be smooth as the analysis is not carried out in real time. Running the analysis calculates and stores the position of the all the components for each time period ready for later viewing. Analyses of complex models can take hours for movement that might replay in a few minutes at real time.

### Playback

To see the results of the analysis we use playback controls.

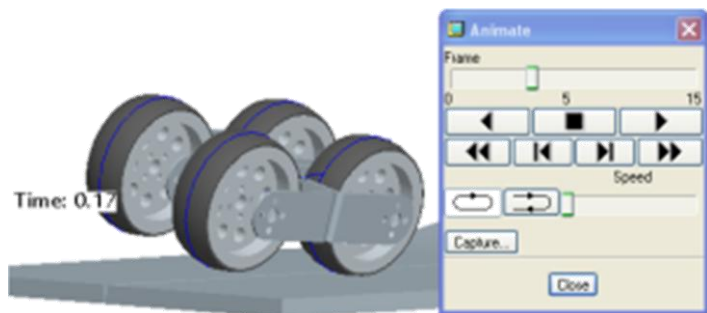
- Near the bottom of the mechanism tree expand the **PLAYBACK** entry.
- Right-click** over the entry for **Drop\_test** and, from the floating menu, select **Play**.



After a short pause the **Animate** dialog will open.

- In the Animate dialog select the **Repeat** button then click on **play**.

You will see the analysis being played back slowly on screen with readout of time.



- Experiment with the speed slider and other control buttons.
- With the animation running, try **rotating** and **zooming** in on the model.

## Video capture

If you want to share your animations with others who don't have **Pro|ENGINEER** installed you can capture the current view as a video file.

12. **Resize** the ProE window to reduce the size of the graphics window. This cuts down the size of the MPG file.
13. In the Animate dialog click on the **Capture...** button to open the Capture dialog.

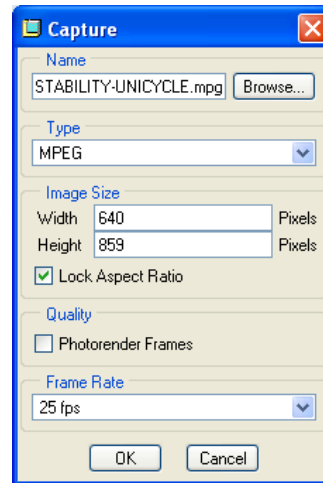
Pro|ENGINEER will default the image size to match the current graphics window.

14. Complete all the options then click **OK**.

The MPG file will be created but may take a few minutes.

15. **Close** the **Playback** dialog.

The same assembly will be needed for the next section.



## Bumpy floor

Although four wheeled vehicles are stable on bumpy surfaces one or more wheels can break contact with the floor unless some form of suspension or compliant chassis is designed.



[http://farm4.static.flickr.com/3258/2892692698\\_0b4160c271.jpg?v=0](http://farm4.static.flickr.com/3258/2892692698_0b4160c271.jpg?v=0)

For the next activity we have provided two different Pro|ENGINEER models to illustrate the benefits of suspension.

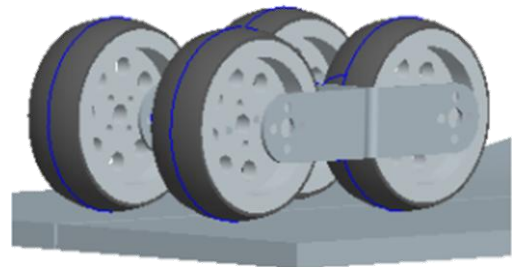
The first has a rigid chassis with no suspension while the second hinges the chassis in the middle similar to dumper trucks.




### Fixed chassis

Pro|ENGINEER should be running with the same assembly used in the previous activity open.

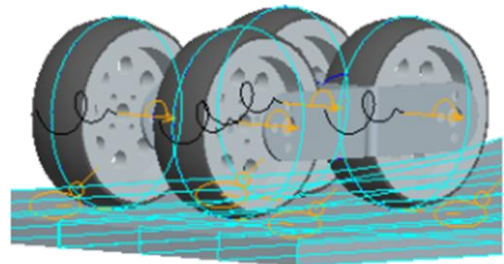
16. If not, set the working directory to `\FTC_Robot\Stability\Fourcycle` and open the file **STABILITY\_FOURCYCLE.ASM**
17. Zoom until the model looks like this.




18. If the model is not sitting on the floor look in the **Main** toolbar across the top of the screen and select the **Drag components tool** 
19. The drag dialog opens. Make sure the Snapshots tab is active then Double click on the **Bumpy\_road** entry.
20. Click on  to close the **Drag** dialog.
21. At the top of the Pro|ENGINEER screen open the **Applications** pull-down text menu and select **Mechanisms**.




Look for the four cam connection symbols on the model plus the **light blue** lines and mesh to show cam surfaces. A mechanism tree will also appear below the normal model tree.

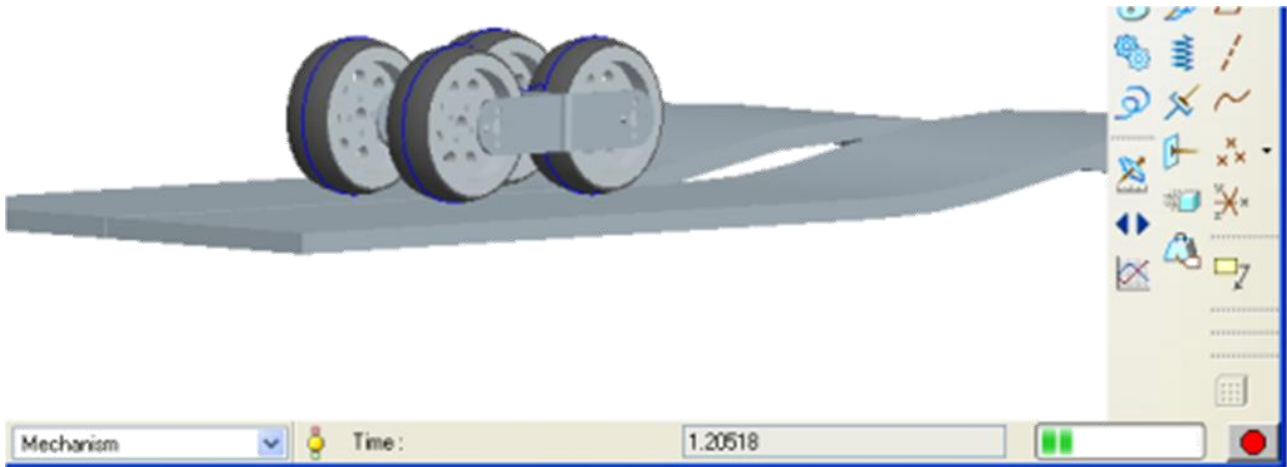


22. In the mechanism tree expand the **ANALYSES** sub heading then **Right-click**  over **Bumpy\_road (DYNAMICS)** and, from the floating menu, select **Run**.


Pro|ENGINEER will mathematically apply gravity and friction to the model, and move the parts according to the motors driving the wheels and the forces acting on the model.


 **Note:** *This simulation may take a long time! Pro|ENGINEER is doing a great deal of mathematical calculation.*

You should see the model drive across the floor and over the bumps.

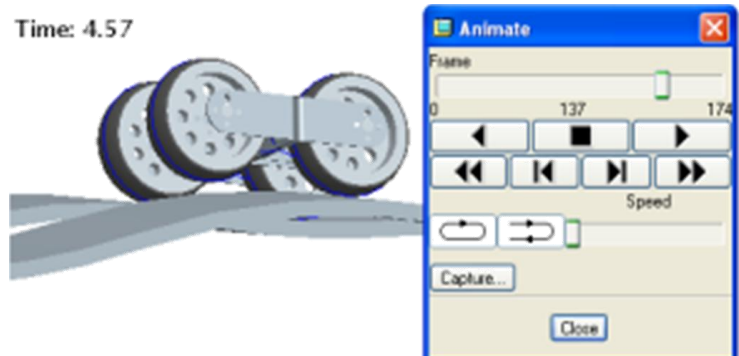


Notice the progress bar along the bottom of the screen showing elapsed time, a progress bar and a red record button. If Pro|ENGINEER seems to have frozen be patient, check the elapsed time which should still be increasing. The analysis should finish after a while.

 If the time indicator stops increasing, the red record button can be used to stop the simulation before the end.

 **Note:** Just like the previous analysis, movement on screen during recording may not be smooth as the analysis is not carried out in real time. The analysis calculates and stores the position of the all the components for each time period ready for later viewing. Analyses of complex models can take hours for movement that replay in seconds at real time.

23. Once the simulation has finished, open the **playback** in the mechanism tree
24. While it is playing, **zoom** and **rotate** the model to examine the way the wheels leave the floor as the robot drives over the bumps.



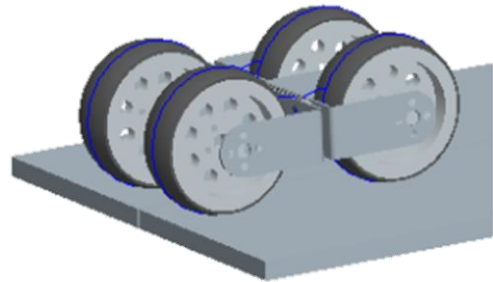
The next analysis will demonstrate a robot chassis with a simple form of suspension.


## Hinged chassis


25. Close all models then open the **File** menu and select **Erase**.
26. From the floating menu select **Not displayed** then **OK** to close the dialog.

This empties the memory of previous models.

27. The working directory should still be **\FTC\_Robot\Stability\Fourcycle**.
28. Open the assembly **Suspension\_fourcycle.asm**
29. **Zoom** until the model looks like this.



30. If the model is not sitting on the floor, look in the **Main** toolbar across the top of the screen and select the **Drag components tool** 

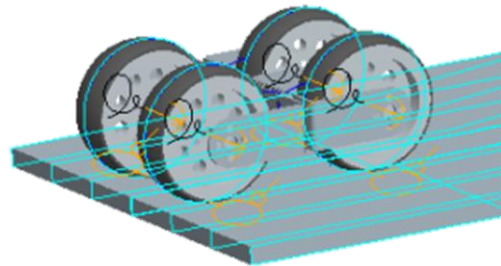
31. The drag dialog opens. Make sure the **Snapshots** tab is active then **Double left click**  on the **Bumpy\_start** entry.


32. Click on  to close the **Drag** dialog.

33. At the top of the **Pro|ENGINEER** screen open the **Applications** pull-down text menu and select **Mechanisms**.

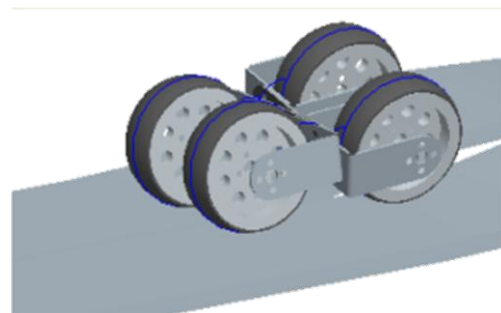


Look for the four cam connection symbols on the model plus the **light blue** lines and mesh to show cam surfaces. A mechanism tree will also appear below the normal model tree.



34. In the mechanism tree expand the **ANALYSES** sub heading then **Right-click**  over **Bumpy\_road (DYNAMICS)** and from the floating menu select **Run**.

**Pro|ENGINEER** will mathematically apply gravity and friction to the model, and move the parts according to the forces acting on them.

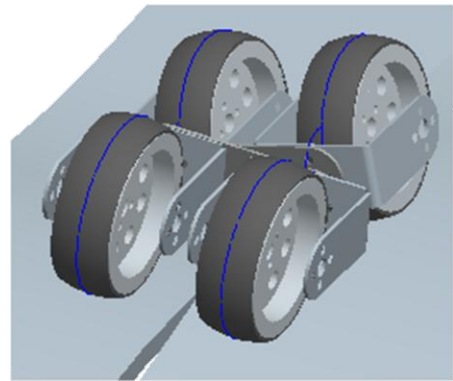


You should see the model drive across the floor and over the bumps.



**Note:** Just like the previous analyses, during recording the movement may not be smooth as the analysis is not carried out in real time.

35. Use the playback tools to replay the simulation and zoom in to see how all four wheels remain in contact with the floor while negotiating the bumps.
36. **Close** all models and **Erase not displayed**.



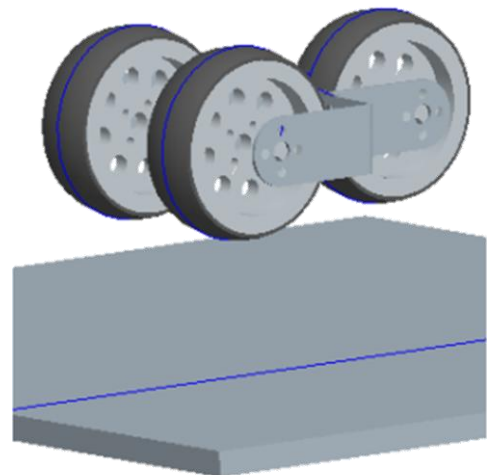
## Task two - Three points of contact – Trike




A tricycle is almost as stable as a four wheel vehicle and doesn't need suspension to negotiate uneven surfaces.

1. Close all models then open the **File** menu and select **Erase**.
2. From the floating menu select **Not displayed** then **OK** to close the dialog.

This empties the memory of previous models.


3. Change the working directory to **\\FTC\_Robot\Stability\Tricycle**.
4. Open the assembly – **Trike\_drop.asm**
5. **Zoom** until the model looks like this.

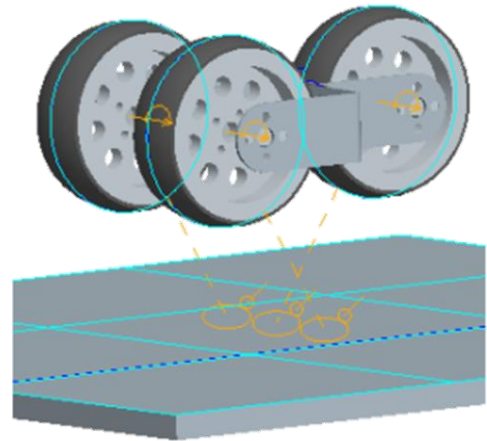



6. If the model is not floating above the floor, look in the **Main** toolbar across the top of the screen and select the **Drag components tool** .
7. The drag dialog opens. Make sure the **Snapshots** tab is active then **Double Left-click**  on the **Trike\_drop** entry.
8. Click on  to close the **Drag** dialog.



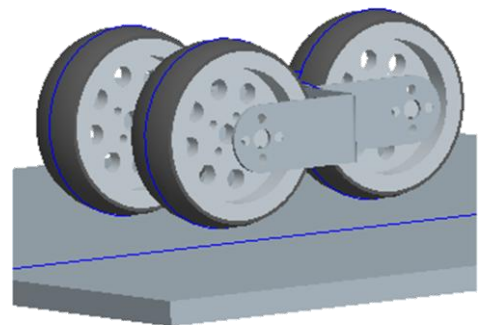
9. At the top of the **Pro|ENGINEER** screen open the **Applications** pull-down text menu and select **Mechanisms**.

 Look for the three cam connection symbols on the model plus the **light blue** lines and mesh to show cam surfaces. A mechanism tree will also appear below the normal model tree.




10. In the mechanism tree expand the **ANALYSES** sub heading then **Right-click**  over **Trike\_drop (DYNAMICS)** and from the floating menu select **Run**.

**Pro|ENGINEER** will mathematically apply gravity and friction to the model, and move the parts according to the forces acting on them.



You should see the model fall down on the floor, bounce a little then come to rest.

 **Note:** Just like the previous analyses, during recording the movement may not be smooth as the analysis is not carried out in real time.


11. Use the **playback** tools to replay the simulation and **zoom** in to examine how the tricycle settles on its wheels.
12. **Close** all models and **Erase not displayed**.




All of the stability models studied so far have been naturally stable.

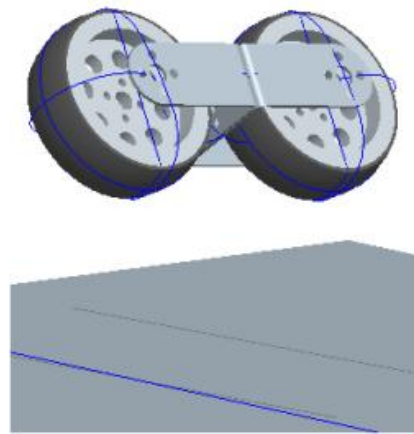
The next three models are inherently unstable and would need some form of active control if they were used as the basis for a robot.


### Task three - Two points of contact – Bicycle

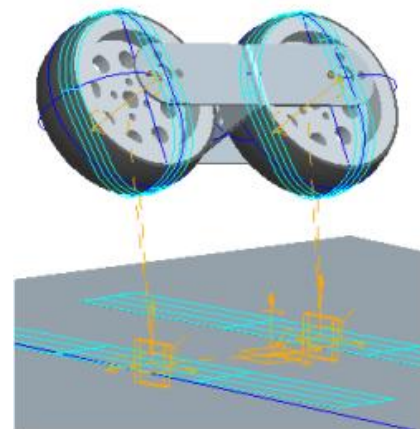
Bicycles will not remain upright without input from the rider. This can be demonstrated by trying to stand a bicycle up in **Pro|ENGINEER**.


 **Note:** because cam connections only work in a single plane, ‘dummy’ components hidden by the floor have been used to allow the bicycle frame to move in a natural way.

1. Change the working directory to **\\FTC\_Robot\Stability\Bicycle.**
2. **Open** the assembly – **Bicycle\_drop.asm**
3. **Zoom** until the model looks like this.
4. If the model is not sitting on the floor look in the Main toolbar across the top of the screen and select the **Drag components tool**  and select the **Drag components tool**
5. The drag dialog opens. Make sure the **Snapshots** tab is active then **Double Right-click**  on the **Bike\_drop** entry.
6. Click on  to close the **Drag** dialog.
7. At the top of the **Pro|ENGINEER** screen open the **Applications** pull-down text menu and select **Mechanisms**.

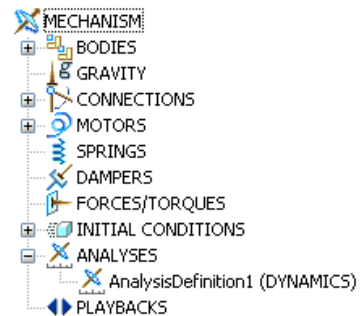


 Look for the two cam connection symbols on the model plus the **light blue** lines and mesh to show cam surfaces. A mechanism tree will also appear below the normal model tree.




8. In the mechanism tree expand the **ANALYSES** sub heading then **Right-click**  over **Bicycle\_drop (DYNAMICS)** and from the floating menu select **Run**.

**Pro|ENGINEER** will mathematically apply gravity and friction to the model, and move the parts according to the forces acting on them.



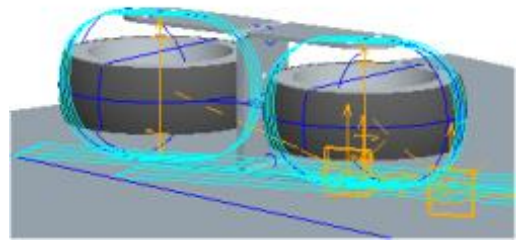
You should see the model fall down on the floor, bounce and then fall over.

 **Note:** Just like the previous analyses, during recording the movement may not be smooth as the analysis is not carried out in real time.

9. Use the playback tools to replay the simulation and zoom in to examine how the bicycle falls over.

This is the first model we have seen which is unstable. You may be interested in searching the web for research designing bicycle robots with dynamic stability control systems.

10. **Close** all models and **Erase not displayed**.



#### Task four - Two points of contact - Segway™

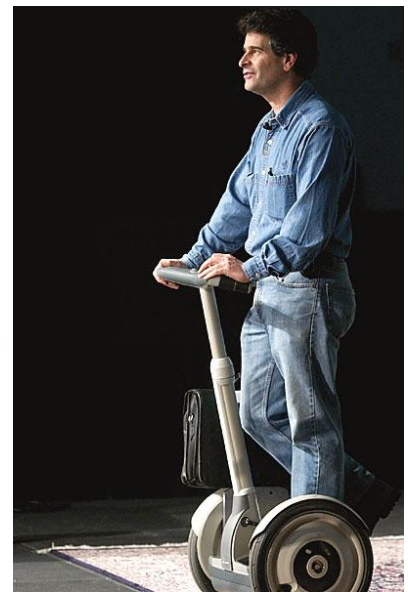
The founder of **FIRST**, Dean Kamen has many inventions to his name including medical devices that have helped lots of people. However, the invention Dean Kamen is most famous for is the Segway™ personal transporter where the rider stands between two wheels either side of the machine.

Instinct makes us think the Segway™ will fall forwards or backwards but in practice this doesn't happen.

What do you think stops it from falling over?

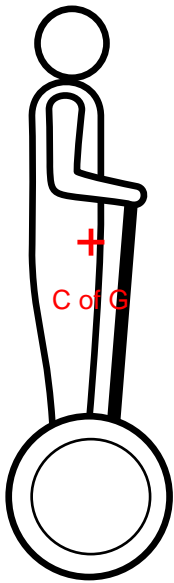
Sensors in the Segway™ detect when it is tilting from the horizontal and power the wheels in the direction of the tilt until the machine is back to vertical. This is called 'Active stability control'.

Does this remind you of the way a unicyclist avoids falling forwards or backwards?



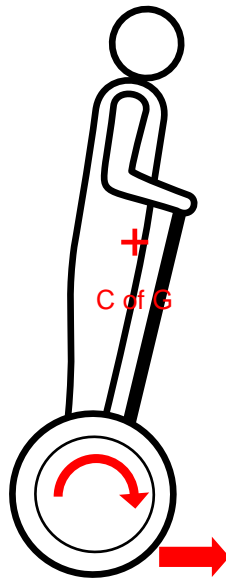
## How Segway™ works

Upright



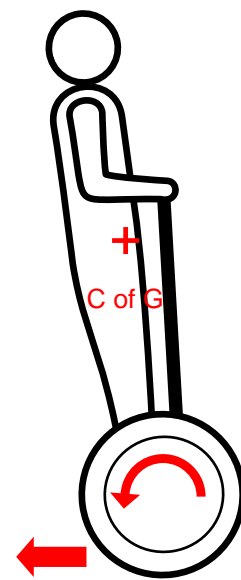
No drive needed

Falling forwards



Motors drive forward to move wheels under centre of gravity

Falling backwards



Motors reverse to move wheels under centre of gravity

This is the same process a bike rider uses to wheelie.

Accelerate to increase the height of the front wheel and move the rear wheel under the C of G

Decelerate/brake to lower the height of the front wheel and move the rear wheel under the C of G.



### Segway™ stability

Like a bicycle, the Segway™ will only remain upright with control input. This comes from sensors controlling the motors. We will demonstrate the inherent instability in the following Pro|ENGINEER assembly.



**Note:** because cam connections only work in a single plane, 'dummy' components hidden by the floor have been used to allow the Segway™ to move in a natural way.

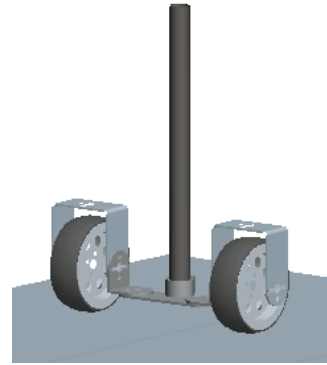
1. Close all models then open the **File** menu and




select **Erase**.

- From the floating menu select **Not displayed** then **OK** to close the dialog.

This empties the memory of previous models.

- Change the working directory to **\\FTC\_Robot\Stability\Segway**.
- Open the assembly **Segway\_stability.asm**
- Zoom** until the model looks like this.




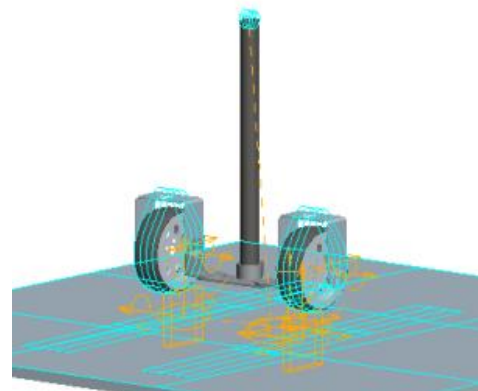
- If the model is not sitting on the floor look in the **Main** toolbar across the top of the screen and select the **Drag components tool** .
- The drag dialog opens. Make sure the **Snapshots** tab is active then **Double Left-click**  on the **Segway\_topple** entry.
- Click on  to close the **Drag** dialog.


- At the top of the **Pro|ENGINEER** screen open the **Applications** pull-down text menu and select **Mechanisms**.



Look for the dummy\_floor components and

three cam connection  symbols on the model plus the **light blue** lines and mesh to show cam surfaces. A mechanism tree will also appear below the normal model tree.



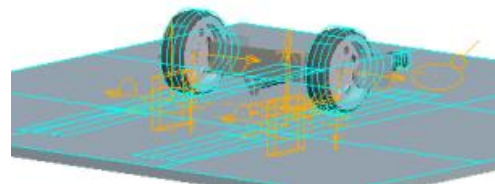
- In the mechanism tree expand the **ANALYSES** sub heading then **Right-click**  over **Segway\_topple (DYNAMICS)** and from the floating menu select **Run**.

**Pro|ENGINEER** will mathematically apply gravity and friction to the model, and move the parts according to the forces acting on them.

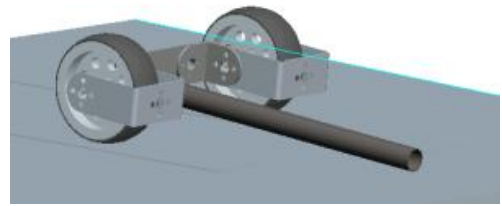
You should see the model fall, bounce, then fall over coming to rest lying on the floor.



**Note:** Just like the previous analyses, during recording the movement may not be smooth as the analysis is not carried out in real time.



- Use the playback tools to replay the simulation and zoom in to examine how the Segway™ settles on its wheels.



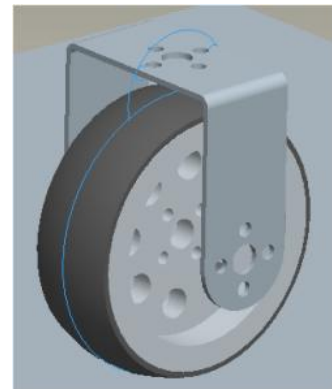
This is the second unstable model studied so far.

### Task five - Single point of contact – Unicycle

A Pro|ENGINEER simulation has been set with a unicycle standing on a flat surface. Three cam connections simulate contact points between the unicycle and the floor.

With gravity applied, the unicycle topples over coming to rest lying on the floor.

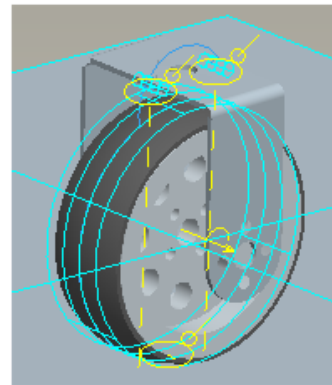
- Start Pro|ENGINEER.
- Set the working directory to `\FTC_Robot\Stability\Unicycle`.
- Open the assembly **Stability-unicycle.asm**
- Zoom** until the model looks like this.



- At the top of the Pro|ENGINEER screen open the **Applications** pull-down text menu and select **Mechanisms**.



Look for the three cam connection symbols on the model plus the light blue mesh to show cam surfaces. A mechanism panel will also appear at the bottom of the model tree.

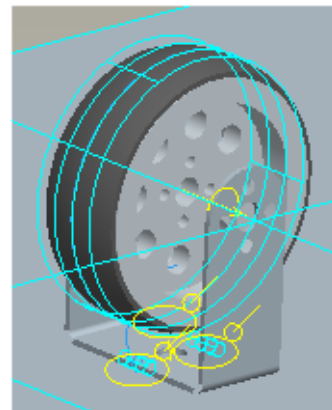


- In the mechanism model tree expand the **ANALYSES** sub heading then **Right-click** over **Unicycle\_stability (DYNAMICS)** and from the floating menu select **Run**.



Pro|ENGINEER will mathematically apply gravity and friction to the model, and move the parts according to the forces acting on them.

You should see the model topple over and come to rest, probably upside down like this.




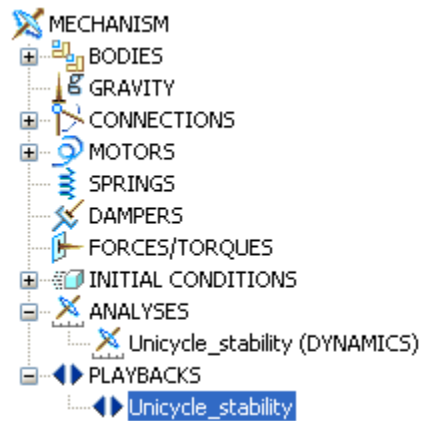


**Note:** The movement may not be smooth as the analysis is not carried out in real time. The analysis stores the position of the all the components for each time period set in the analysis definition ready for later viewing.



### Playback

To see the results of the analysis we use playback controls.

7. Near the bottom of the mechanism tree expand the **PLAYBACK** entry.
8. **Right-click**  over the entry for **Unicycle\_stability** and, from the floating menu, select **Play**.

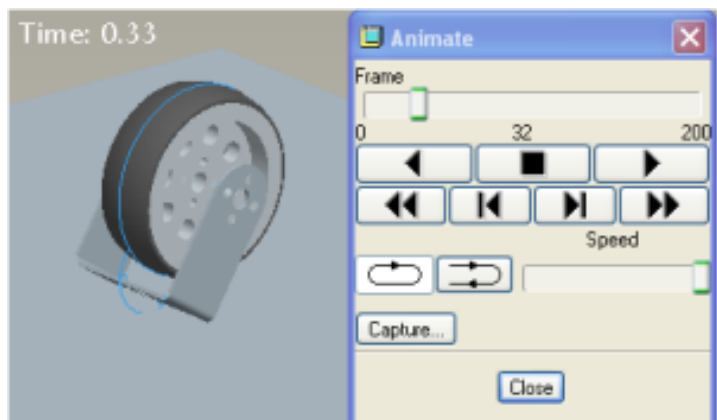


After a short pause the **Animate** dialog will open.

9. In the Animate dialog select the **Repeat** button  then click on **play** .

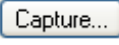

You will see the analysis being played back slowly on screen with a readout of Time.

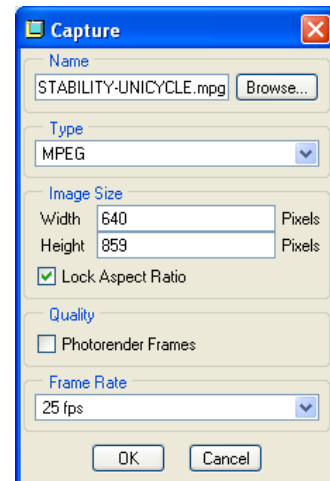
10. Experiment with the speed control slider and other buttons.
11. With the animation running, try rotating and zooming in on the model.



### Video capture

If you want to share your animations with others who don't have **Pro|ENGINEER** installed you can capture the current view as a video file.

12. **Resize** the ProE window to reduce the size of the graphics window. This cuts down the size of the MPG file.
13. In the Animate dialog click on the  button to open the Capture dialog.
14. ProE will default the image size to match the current graphics window.
15. Complete all the options then click .
16. The MPG file will be created which may take a few minutes.
17. **Close** all models and **erase** all files from memory.



### Unicycle theory

Understanding how a unicyclist balances on a single point of contact with the ground will help you understand the control systems for robotics.

The rider has two main actions to 'control' the movement of the unicycle;

1. Pedaling forwards and backwards
2. Steering by swinging their hips.

**Falling forwards** - Imagine the unicycle is beginning to fall forwards; the rider senses this and pedals forwards to bring the unicycle upright, i.e. with the centre of gravity directly over the wheel contact with the ground.

**Falling sideways** – This requires two inputs by the rider. The unicycle is falling sideways which the rider senses and swivels their hips turning the wheel to face in the falling direction. The rider can now pedal forwards to bring the unicycle back to the upright position.



In practice the rider is constantly sensing the vertical axis of the unicycle and steering/pedaling in the appropriate direction to keep the wheel contact below the centre of gravity.

This briefing is now complete.



## Briefing four – Electric motors

These are probably the most common ‘actuators’ in use today. Electric motors can be found in electric screwdrivers, digital cameras, cars, elevators, etc.



### Motors applied to FTC

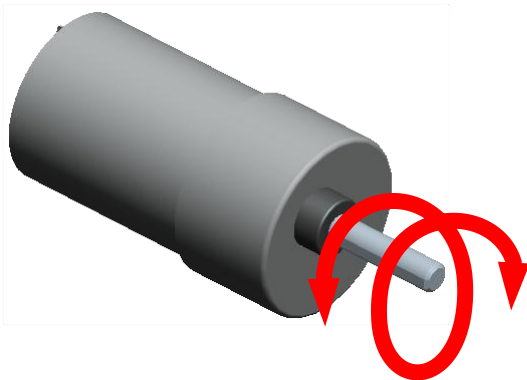
The example robot provided in this project guide uses most of the components in the Tetrax robot kit. The wheels and shoulder joint are driven using the geared DC motor and the elbow and gripper movements are driven by servo motors.

Things can become confusing when the following Pro|ENGINEER simulations use a ‘servo motor’ to drive the robot wheels and a force motor to operate the gripper.

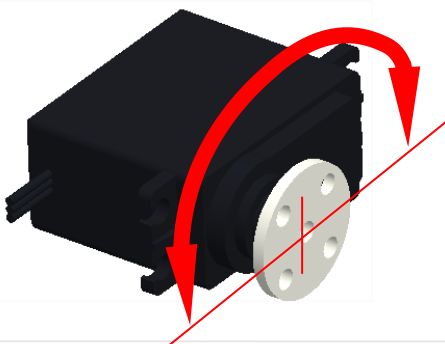
### DC motor and servo motor

Electric motors and servos are key output devices for robotic control systems. Two different types of electric motors are provided in the FTC Tetrax™ kit. Each provides a different type of rotational movement.

The **DC geared motor** will continue to rotate as long as it is switched on.



The **Servo motor** rotates through only 180 degrees and maintains a position depending on the square wave signal being sent to it. This will be explained in more detail later.

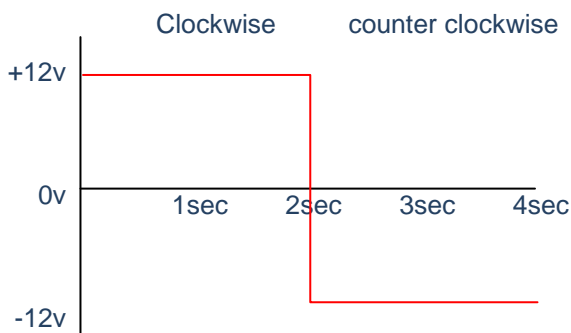


## DC geared motor

The DC motors have a built in gearbox that reduces the rotational speed and increases the turning force (torque) of the output shaft.



The input to the DC geared motor is a voltage up to 12v dc across the two terminals. While connected, the spindle revolves continuously. Reversing the voltage reverses the direction of the output shaft.



The graph above shows +12v being applied to the motor for two seconds and -12v for the next two seconds. We would expect the motor to turn clockwise then anticlockwise with the spindle ending up exactly back where it started

The DC geared motor in the Tetrax™ kit does not have feedback which can cause problems if the motor stalls. IN the context of an FTC robot this may not be a problem as the torque available from the DC geared motors is quite considerable.

A data sheet is available for this motor which includes the following information:

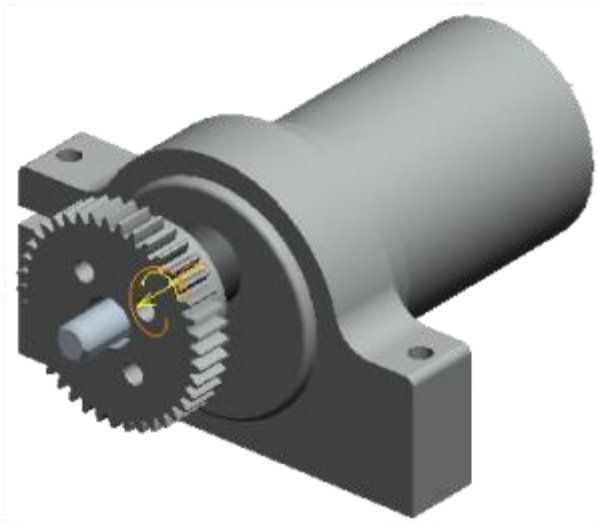
- Rotational speed: 154 revolutions per minute (rpm)

Stall torque: 23.40 Kg cm (ISO) 325 oz/in (ANSI)

This information was used to construct **Pro|ENGINEER** models including a 'servo' motor turning the output shaft from the gearbox. Later in this tutorial you be shown how these can be built into virtual robots.

### CAD model

1. Set the working directory to **IFTC\_Robot**
2. Open the model **dc\_turning\_demo.asm**
3. Open the **Applications** pull-down menu and select **Mechanisms**.



The model is a DC geared motor with a small gear fixed to the spindle. You should see two symbols;

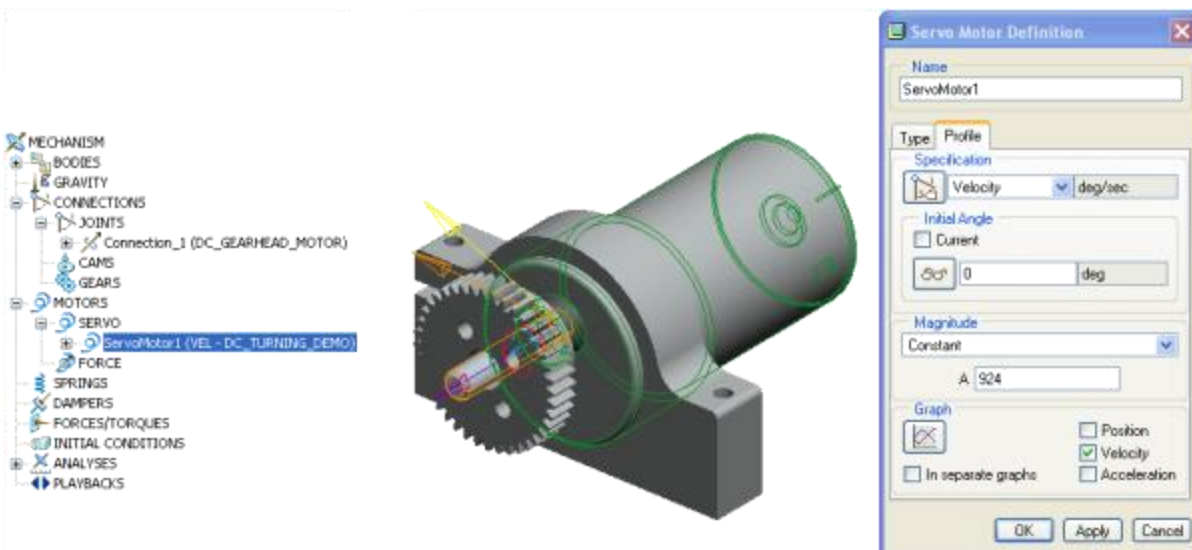


Denotes a 'Pin' mechanical connection exists, in this example allowing the spindle of the DC geared motor to rotate.



Denotes a 'servo' motor, in this case driving the output shaft of the DC geared motor.




4. At the bottom of the model tree, in the mechanisms section, expand **MOTORS** and **SERVO** then right click on the **ServoMotor 1** entry.
5. From the floating menu select **Edit Definition**



In the Servo Motor Definition dialog the **Type** tab is listing the motion axis for the pin, in this case the rotation axis of the servo motor.

6. Select the **Profile** tab.

This shows the 'servo' motor drives the shaft at a constant rotational '**velocity**' of **924** degrees per second. Where do you think this value came from?

7. **Left Click**   to close the **Servo Motor Definition** dialog.
8. In the mechanism tree expand the **ANALYSES** heading, **Right-click**  over **Motor\_turn (POSITION)** and from the floating menu select **Run**.
9. Once the analysis has finished, use the **PLAYBACK** to look at the motion in detail.
10. **Close** the model and **erase** models from memory.

### Calculating rotational velocity

The data sheet for this DC geared motor specifies a rotational speed of 154 revolutions per minute (rpm). To convert this to degrees per second:

Convert minutes to seconds

$$\frac{154}{60} \approx 2.57 \text{ revolutions per second}$$

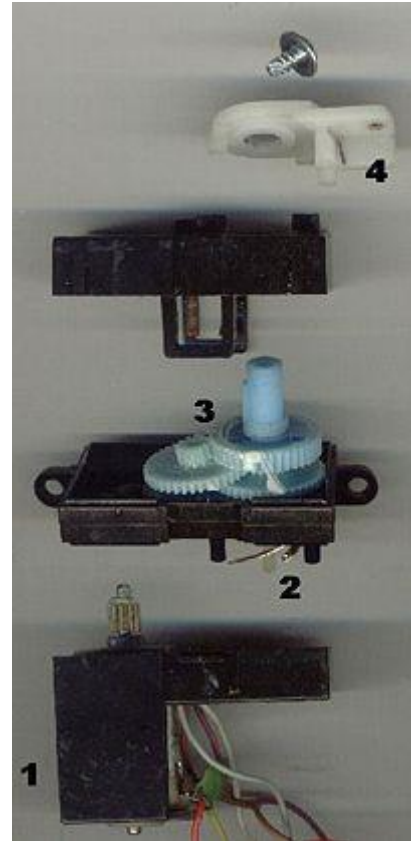
Convert revolutions to degrees

$$2.57 \times 360 = 925 \text{ degrees per second}$$

## RC Servo motor

These are based on a small DC motor packaged with a geared reduction and electronic control system. The output shaft provides low speed of rotation and high turning force (torque).

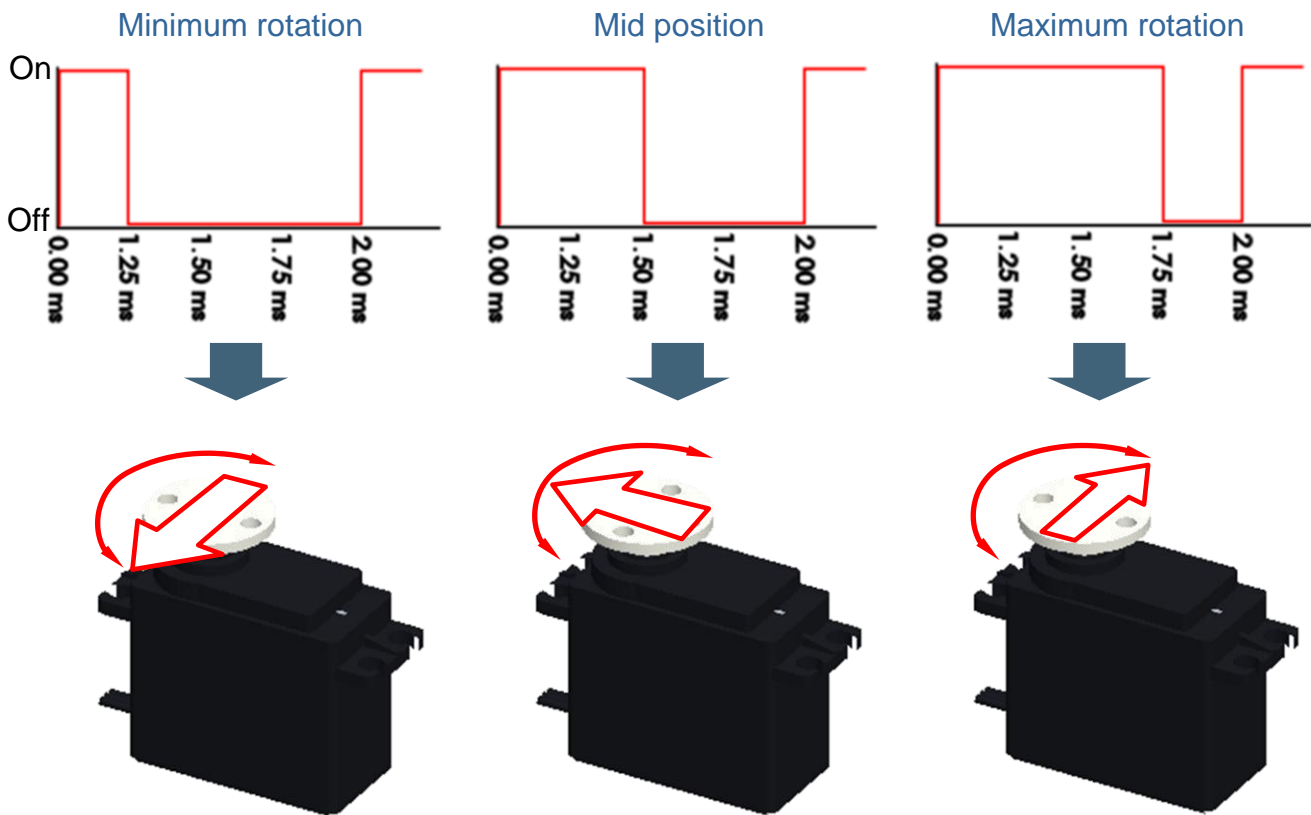
Built into RC servos are sensors to detect the rotational position of the output shaft and electronics to link motor control to the shaft positional information.



<http://en.wikipedia.org/wiki/Servomechanism>



A constant stream of high frequency pulses control the position of a servo motor.

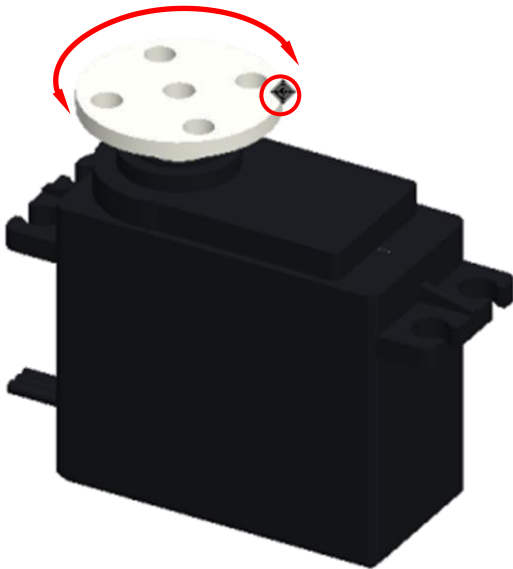
The difference between the on and off time sets the position. Here we show the pulse signal and the position of the servo for three different electronic signals.



### Limiting rotation

The RC servo motors in the Tetrix™ kit will only rotate through 180 degrees. This has been built into the Pro|ENGINEER model.

11. Set the working directory to **\FTC\_Robot**
12. Open the model **SERVO\_HS475.ASM**
13. Use  **Drag** or **Ctrl+Alt+Left Click**  to find out how the output horn moves in the assembly.



The horn should only move through half a revolution.

14. In the model tree find the entry for   **SERVO\_METAL-HORN.PRT** .

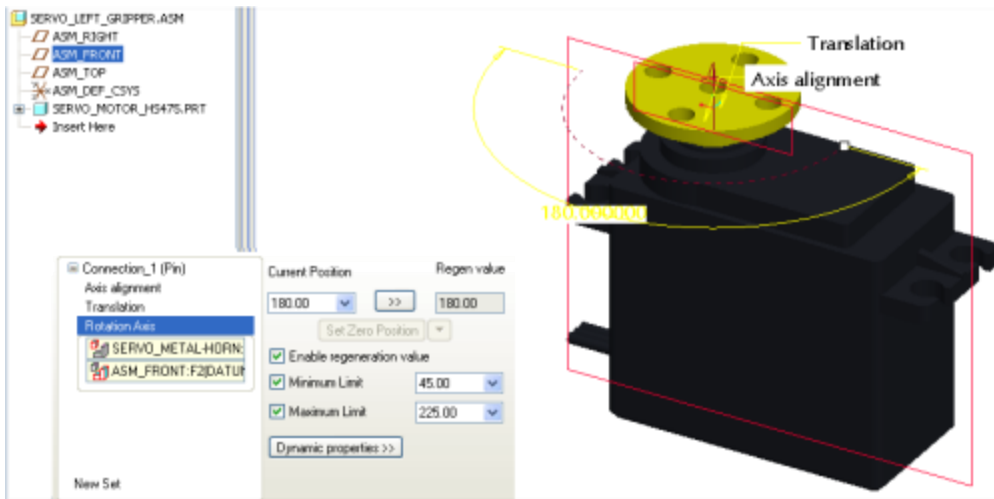
Between the icon and text is a very small square with a dot in the centre, highlighted above with a red circle. The square represents a component which is 'packaged' with at least one degree of movement within the assembly. The dot tells us this component is assembled using a mechanical connection rather than an assembly constraint. Let's look inside this mechanism connection.

15. In the model tree right click on **SERVO\_METAL-HORN.PRT** and from the floating menu choose **Edit Definition**.

The Assembly dashboard will open and the metal horn will change color to yellow to show this is the component we are investigating and the assembly connection is complete.

16. In the dashboard, open the **Placement** tab and click on the **Rotation Axis** entry.

The model will display the geometry used to constrain the rotational movement of the servo horn and a pop-out panel will list them including; **maximum** and **minimum** limits and a **Regen** value.



The purpose for the first two should be clear. The **Regen** value is useful for positioning the joint when the model is regenerated.

- 17. **Close** the dashboard and sub assemblies
- 18. **Erase** models from memory

### Servo feedback for DC motor

The Tetrax geared DC motor can be converted into a servo motor with the Tetrax encoder.



Tetrax encoder assembly instructions

This briefing is now complete.



## Briefing five - Mechanisms

Mechanisms are a logical next step to static assemblies. Mechanisms allow parts of your assembly to operate in a mechanical sense. Mechanical movement is usually divided into

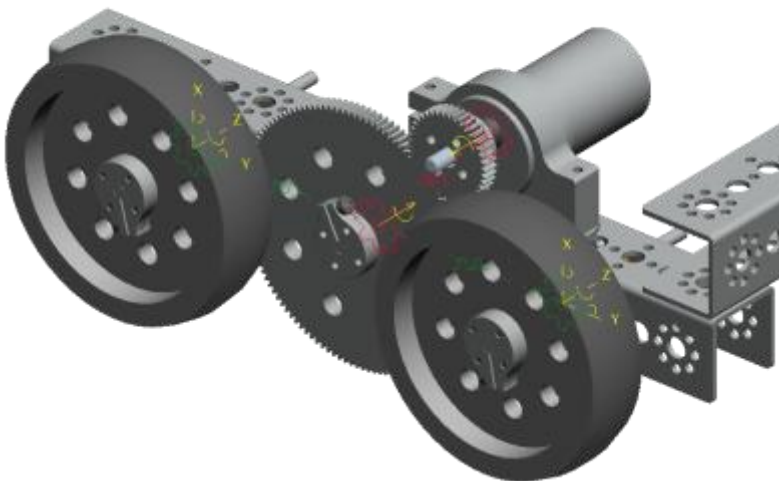
### Mechanisms applied to FTC

The example robot provided in this project guide uses most of the components in the Tetrix robot kit.

Before you look at the Pro|ENGINEER mechanisms simulations you are advised to learn about constraints and how to model mechanical assemblies. The following **Online Courses** in PTC University Coach will help: <http://schools.ptculms.com/>

- **Assemblies > Bottom Up Design**
- **Introduction to Mechanisms > Start with Overview and Creating Mechanisms then Advanced Connections > Gear Pair Connections.**

Understanding the basics will let you come back to take additional modules as you need them.



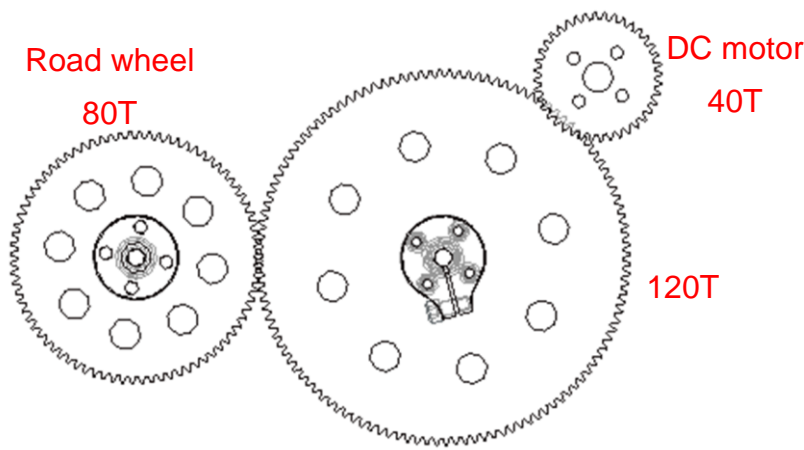
In addition, PTC's schools curriculum contains projects covering aspects of mechanisms including:

- **Starchaser Rocket**(assembly and force motors)
- **Scalextrc4Schools** (assembly, force motors, slot connections, gear ratios, efficiency)

### Task one - Gearing

There are three sizes of gears in the Tetrix™ kit which we describe by the number of teeth they have; 120T, 80T and 40T.

In the tutorial robot a set of three gears connect the DC motor to the large diameter wheel.



From Briefing five we know the DC motor rotates at 924 degrees per second so, using a little math we can work out the rotational velocities for each of the gear pairs.

**40T – 120T gear pair**

For this gear pair the 40T gear is the driver and the 120T gear is being driven. The formula for working out the rotational velocity is:

$$\text{Driver Teeth} \times \text{rotational speed} = \text{Driven teeth} \times \text{rotational velocity}$$

$$40 \times 924 = 120T \times ?$$

$$\frac{40 \times 924}{120} = ?$$

$$120T \text{ Rotational speed} = \mathbf{308} \text{ degrees per second}$$

**120T – 80T gear pair**

The 120T gear now becomes the driver and the 80T is the driven gear.

$$\text{Driver Teeth} \times \text{rotational speed} = \text{Driven teeth} \times \text{rotational speed}$$

$$120 \times 308 = 80T \times ?$$

$$\frac{120 \times 308}{80} = ?$$

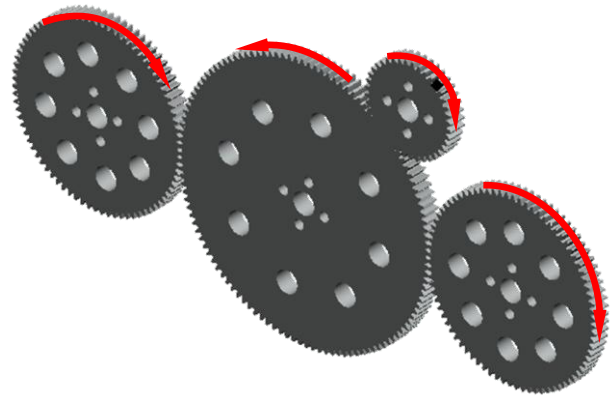
$$80T \text{ Rotational speed} = \mathbf{462} \text{ degrees per second}$$

This same equation could be used to work out the angular velocities of any gear pair.

### Interrogate gear pair connections.

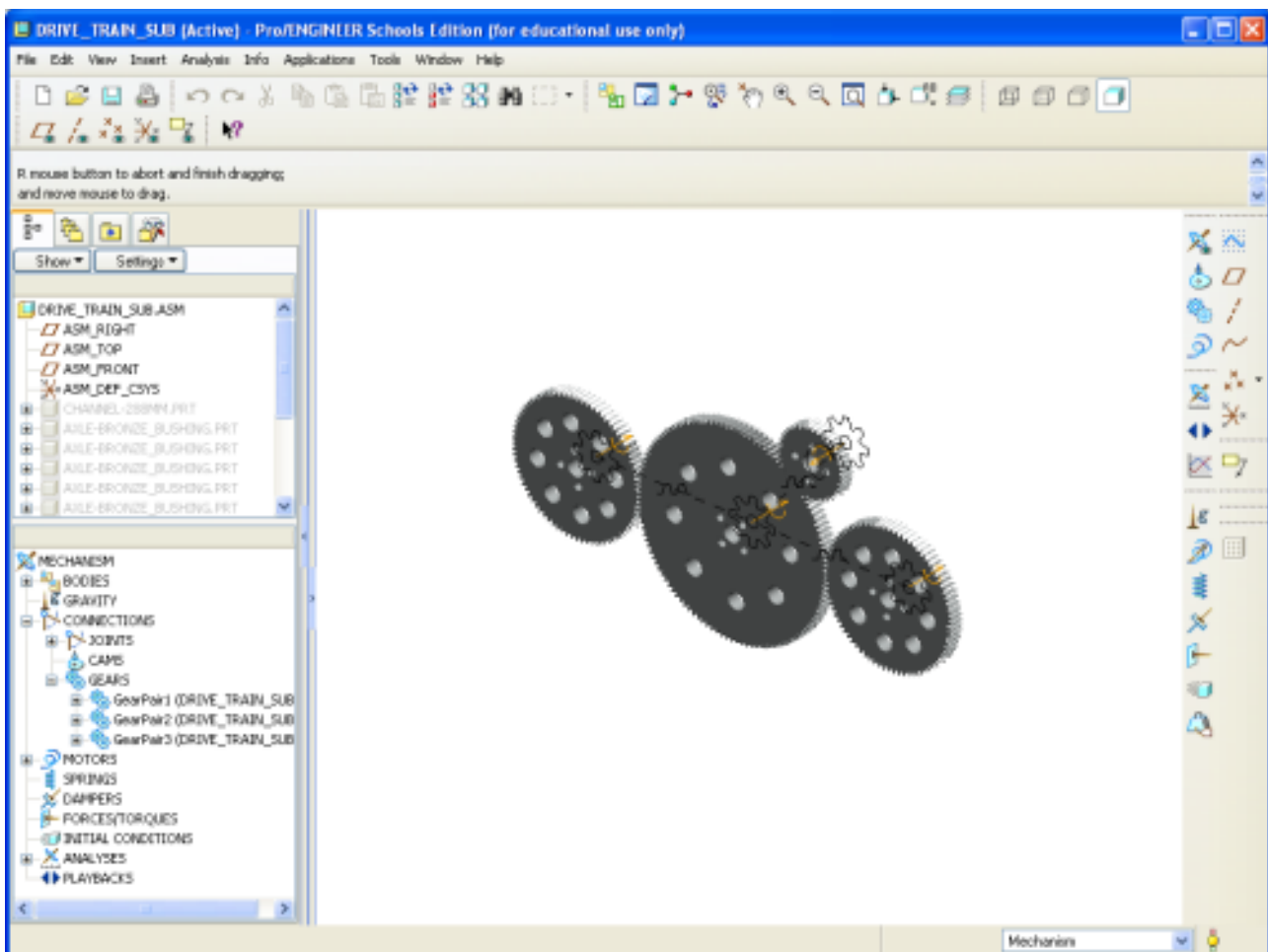
1. Pro|ENGINEER should be open with no models on screen and none in memory.
2. Set the working directory to **\\FTC\_Robot\Gearing**.
3. Open the assembly **DRIVE\_TRAIN\_SUB.ASM**


Many of the components and some sub assemblies have been hidden to make it easier to interrogate the mechanism.

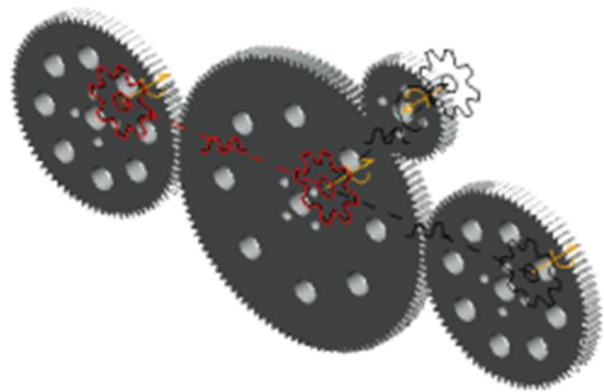


4. Hold down **Ctrl+Alt** then drag the perimeter of the small gear to test the kinematic movement.
5. In the main toolbar across the top of the screen click on **Applications** then select **Mechanisms**.

The model will display symbols showing mechanical connections, motors, etc and a new mechanisms panel will appear below the model tree.

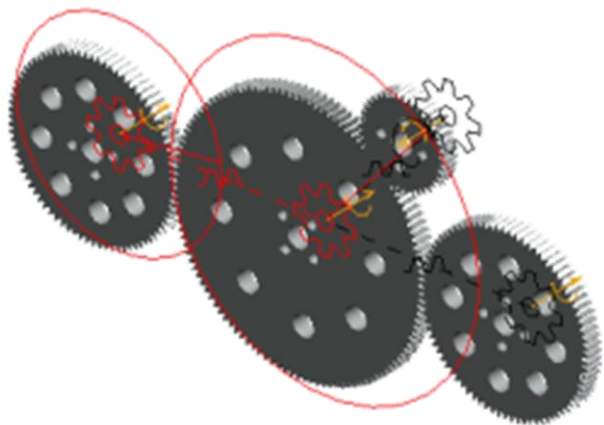


6. In the mechanisms panel, click on the **+** next to **CONNECTIONS** to expand it.
7. Expand the **GEARS** entry and you will see several gear definitions.
8. **Left Click**  on one of the gear entries and you will see the gear definition symbols highlighted on the model.



The illustration here shows the gear connection between the large idler and the gear driving the rear wheel.

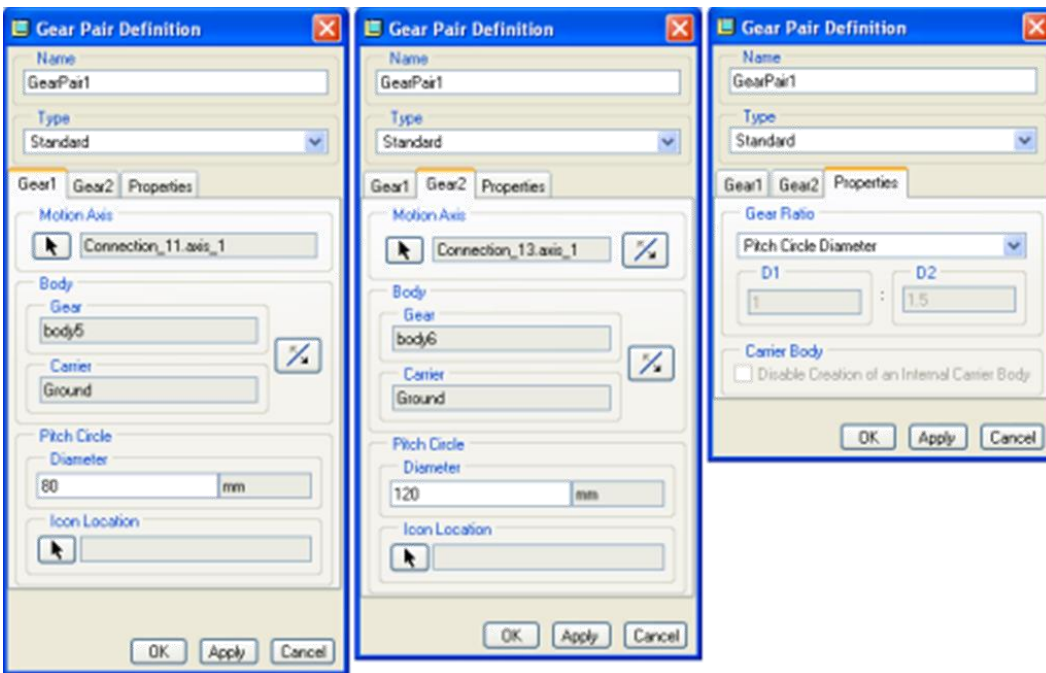
9. In the mechanisms panel, right click on the gear connection and, from the floating menu, select **Edit Definition**.
10. The **Gear Pair Definition** dialog will open and the model will display geometry used to define the gear pair.



Each gear definition has three tabs one for each of the gears in the connections and a Properties tab.

11. Click on each of these tabs and note what is highlighted on the model.

**Note:** In this diagram the three tabs are shown side by side. In practice you will only be able to view one panel at a time.



The gear ration is defined by the two numerical values; **60mm /3.15** in for the **Gear1** tab and **120mm / 4.72** in for the **Gear2** tab. Although these are labeled as mm/in, these are not absolute values. They represent the ratio between the two gears. In this example we used the number of teeth on each gear.

The third tab provides a ‘user defined’ method of defining gear pairs as a ratio. If this were used we can enter simple numbers like **1(40T) 2(80T)** and **3(120T)**.



**Note:** You may want to try the analysis which drives the gears for 1 second.

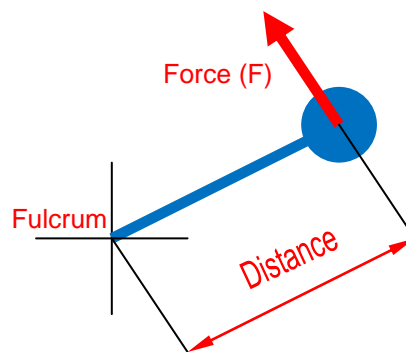
12. **Close** any dialogs and models then **Erase not displayed**.

### Task two - Torque

Torque or turning force is important in robots because you need a certain amount of torque to turn the wheels to make things move.

Another name for torque is moment and they are made up of a force and a distance from the axis or fulcrum of a rotating object.

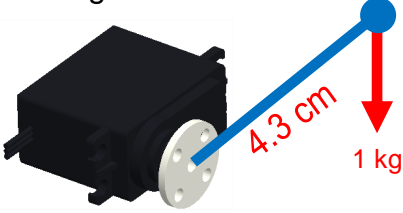
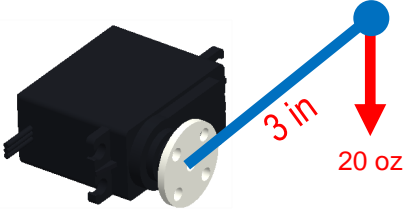
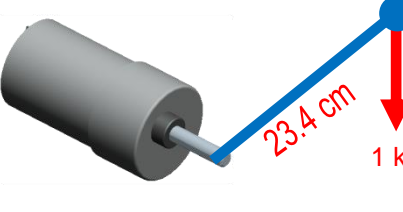
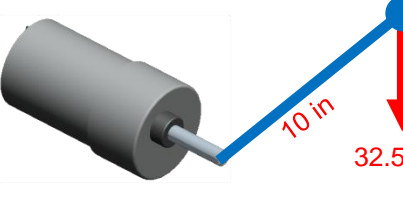
$$\text{Torque (T)} = \text{distance}(r) \times \text{Force (F)}$$




Metric (mm)	ANSI (inches)
Force in Newtons (N)	Force in pounds (lbf)
Distance in meters (m)	Distance in feet (ft)
Moments in Newton Meters (Nm)	Moments in foot pounds (ft lbf)

### Motor torque

Motors in the Tetrax™ kit have a published torque at the output shaft. Appendix two and three are copies of the manufacturer’s data sheets for the servo and DC geared motors respectively. Taken from the data sheets, here are the published torque values for the two motors with examples.

Device	Metric	ANSI
<b>Servo motor</b> Stall torque (4.8v)	$4.3 \text{ kg.cm} = 42.2 \text{ N cm}$ 	$60 \text{ in ozf} = 3.75 \text{ in lbf}$ 
<b>DC geared motor</b> Stall torque (12v)	$23.4 \text{ kg.cm} = 229.55 \text{ N cm}$ 	$325 \text{ in ozf} = 20.3 \text{ in lbf}$ 

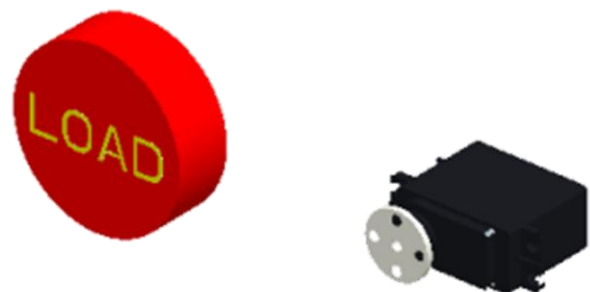
 **Note:** the data sheet for the DC motor only gives metric information. The torque figure was converted using <http://www.unitconversion.org/>


### Torque applied to FTC

We can use this information entered into Pro|ENGINEER models to help design FTC robots. To demonstrate this you will use a model of the servo motor rotating a mass against the force of gravity. The model you will use has been simplified. In practice there would be an arm or beam connecting the load to the servo.

#### Servo motor torque

1. Start Pro|ENGINEER.
2. Set the working directory to **\FTC\_RobotMoments**.
3. Open the assembly **Lifting\_a\_mass.asm**
4. **Rotate/zoom** until the model looks like this.



5. Hold down the **Ctrl + Alt** keys and, with the **Left mouse button**  held down, drag the **red** load to explore motion in the assembly. Take special note of the limits of movement.

A cylindrical object is assembled to the output shaft at a certain distance from the axis to act as a 'load' on the motor. The mass of this object has been carefully set just greater than the torque capability of the motor.


The servo motor has a 'Force' motor applied which will move the output shaft with a defined torque.


### Interrogate the model


Before running the analyses we will interrogate the model to find the mass of the load, how far it is from the pivot and the torque of the servo motor.

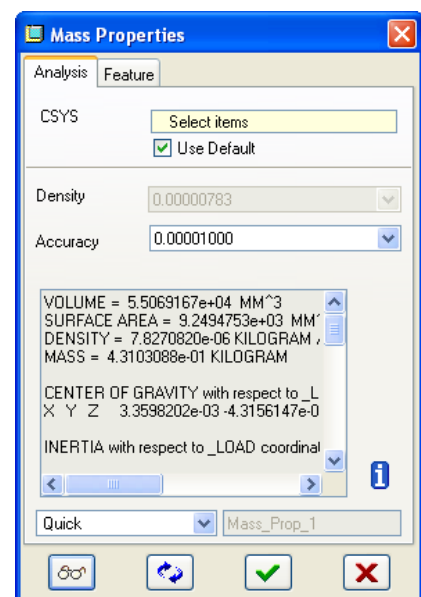
Knowing this information we will ask you to predict what will happen before the analysis is run.


6. In the model tree right click over the part **LOAD.PRT** and from the floating menu select **Activate**.
7. In the Main toolbar open the **Analysis** pull-down text menu, click on **Model** and then **Mass Properties**.


8. In the Mass Properties dialog, click on  to perform the analysis

 Note: The metric mass of the part is **0.431 kg**.


 Note: The ANSI mass of the part is **0.96 lb**.




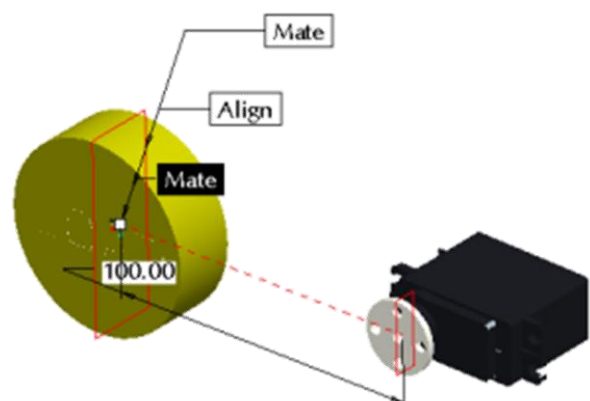
9. Click  to close the dialog.

10. In the model tree **Right-click**  over **LIFTING\_A\_MASS.ASM** and from the floating menu select **Activate**.

11. In the model tree **Right-click**  over the **LOAD.PRT** and from the floating menu select **Edit Definition**.

 The assembly constraints will preview and show the center of the load is located **100mm/3.95 inches** from the axis of the servo.

12. Click  to close the assembly




dashboard or middle mouse click.

A quick calculation shows the turning force or torque the load will place on the servo:

$$\text{Torque (T)} = \text{distance(r)} \times \text{Force (F)}$$

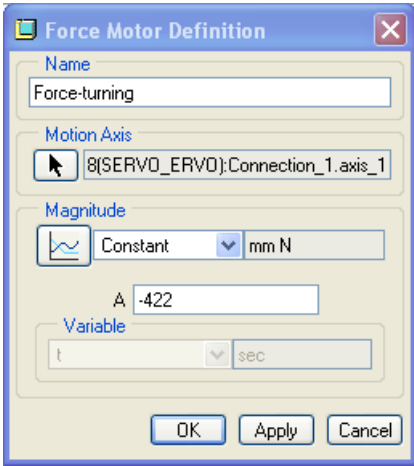
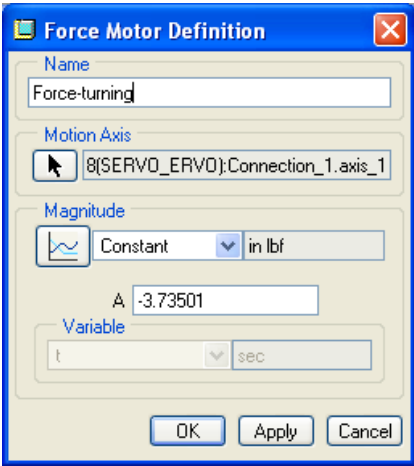


Metric	ANSI
r = 100 mm	r = 3.95 in
F = 0.431 kg x 9.81 = <b>4.23</b> Newtons	F = 0.96 lbf
T = r x F	T = r x F
T = 100 x 4.23	T = 3.95 x 0.96
T = <b>423</b> N mm	T = <b>3.8</b> in lbf = <b>60.7</b> in oz

If we now look at the torque programmed into Pro|ENGINEER for the servo motor:

13. In the **Main** toolbar open the **Applications** menu and select **Mechanism**.
14. In the mechanism tree, expand **MOTORS** and **FORCE** then **Right-click**  over the entry **Force-turning (LIFTING\_A\_MASS)**.
15. From the floating menu, select **Edit Definition**

The Force Motor Definition dialog opens.



<b>Metric</b>	<b>ANSI</b>
	
<p> <b>Note:</b> The value is the same as the data sheet (4.3 kg cm) but expressed as millimeter Newton (mm N).</p>	<p> <b>Note:</b> The value is the same as the data sheet (60 oz in) but expressed as inch pound feet (in lbf).</p>

16. **Left Click**  **Close** to close the dialog.



**Prediction**

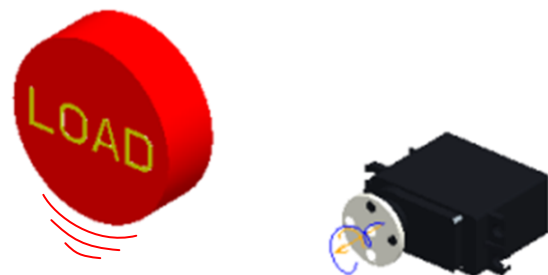
What would you expect to happen if we start the analysis in the following situations?

- Starting with the load horizontal?
- Starting with the load vertical?


Make a note of your prediction then carry out the analyses that follow.

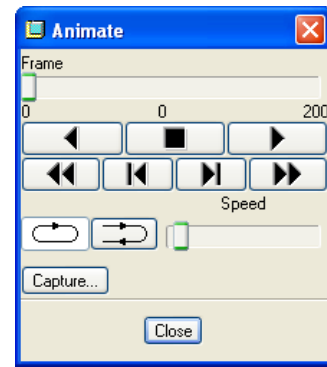
**Torque simulation**


17. Change the view to **Trimetric**.
18. In the Mechanism tree on the left of the screen expand the **ANALYSES** heading
19. **Right-click**  on the  **Horizontal\_start** entry and, from the floating menu, select **Run**



The mechanism will be analysed and a playback created.

20. In the mechanism tree PLAYBACK section, **Right-click**  over the **Horizontal\_start** entry and select **Play**
21. In the mechanism tree expand the **PLAYBACK** heading and **Play** the simulation several times until you can describe accurately how the mechanism moves.



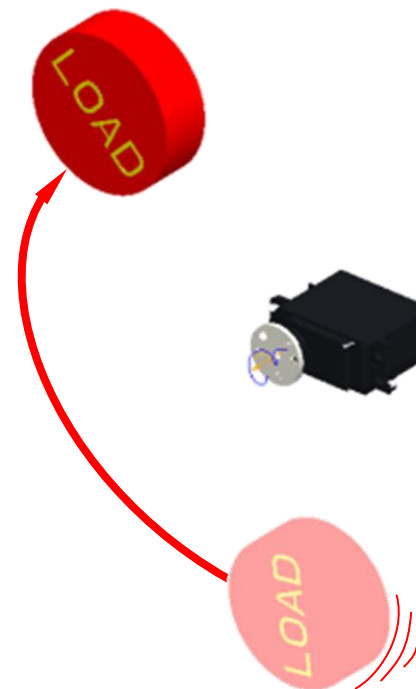
-  **Note:** You can rotate and zoom while the movement is playing back
22. Now try to explain **why** the mechanism moves in this way.

You should have recognised the torque from the servo motor acting clockwise is just less than needed to overcome the torque from the load. This assumes a static situation which is why the analysis started with the load stationary at the horizontal position.

The next analysis would seem to contradict what you have just seen and suggest there is enough torque in the motor to overcome the load.

23. In the mechanism tree **ANALYSES** section, **run** the entry **Vertical\_start**.
24. Once the analysis has finished, use the **PLAYBACK** controls to replay the movement watching how the mechanism moves.
25. Describe what you think is happening and why it is behaving like this.

Pro|ENGINEER is able to simulate the dynamic movement of the mechanism helping us understand what is happening.



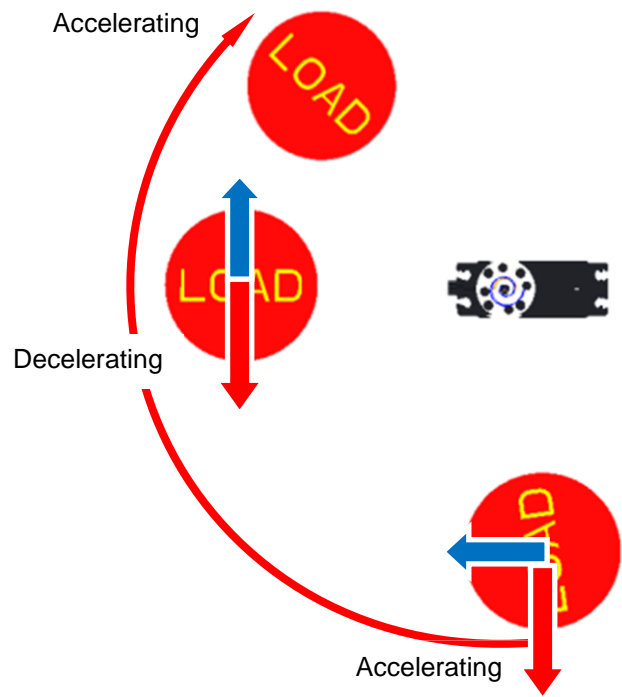
At the start of the analysis the servo is able to rotate the load as **gravity** is acting vertically down and servo torque is acting **sideways**.

The servo is able to accelerate the load adding kinetic energy or inertia due to its mass and velocity. This is sometimes called 'flywheel effect'.

As the load approaches the 9 o'clock position, the force of **gravity** on the load is now opposing the servo **torque**. The forces don't balance so there is a net downward force decelerating the load.

However, the kinetic energy in the load is enough to carry it past the 9 o'clock position where it begins to accelerate again.

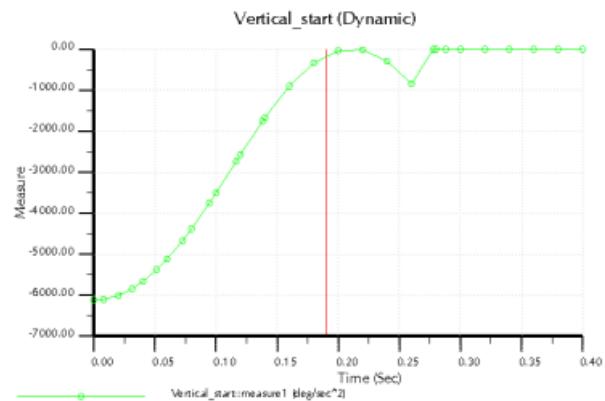
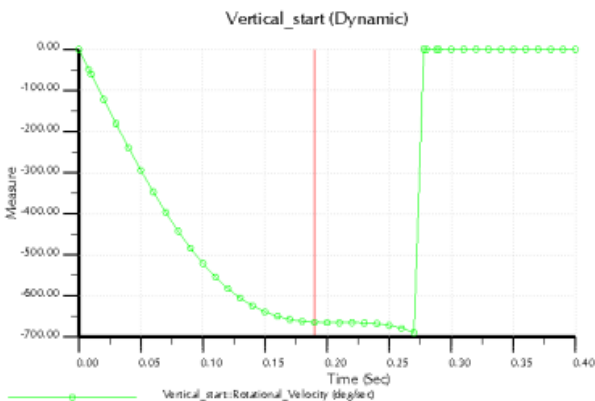
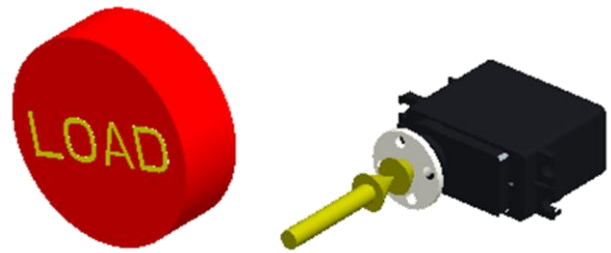
The load comes to a sudden stop because the servo only has 180 degrees of movement and this is the limit of rotation.



Pro|ENGINEER has additional analysis tools including the option to 'measure' parameters and graph them dynamically while the playback is being viewed.

This model has a force arrow showing angular velocity. During playback force arrows change size and direction.

Here we have playback with two graphs showing rotational velocity on the left and rotational acceleration on the right.



- Open the **File** pull-down menu **Left Click**  **Erase** and from the floating menu select **Current**.

This closes the current model and removes files from memory.

This has only touched on the powerful tools for mechanisms dynamic simulation and analysis in Pro|ENGINEER.

To learn more about these tools in Pro|ENGINEER go to PTC University Coach at:  
<http://schools.ptculms.com/cadtrain/coachlms/client/login>

Try out the online Pro|ENGINEER courses under the heading:

- **Introduction to Mechanism.**
- **Mechanism Simulation using Pro/ENGINEER Wildfire 4.0**

This briefing is now complete.

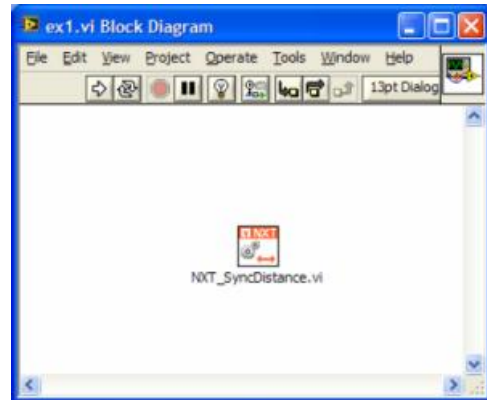
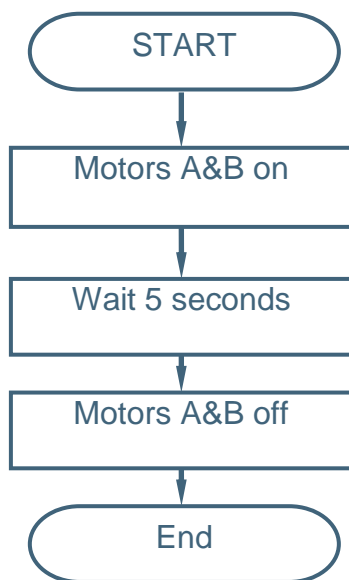
## Briefing six - Movement and control

You will now learn about the geometry of movement and how math can be really helpful when designing your robot and programming its movement.

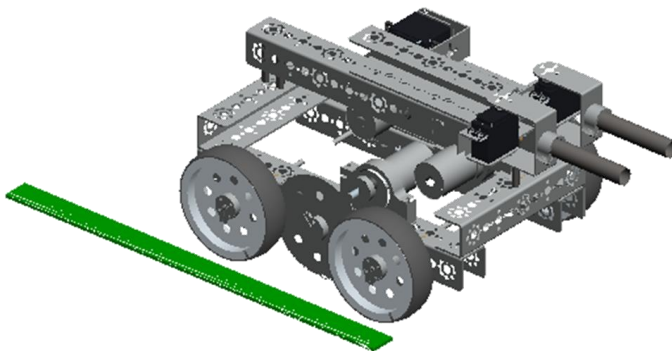
The simplest form of instructions used to program robots switches motors on for set periods of time.

For example:

Labview example



To help with programming we need to know how far our robot will travel in a set period of time. To find this out we could build the robot and drive it for a set time measuring how far it travels.



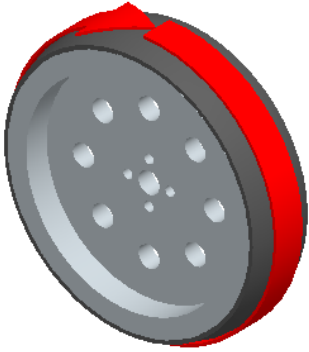
However, what if you haven't built the robot yet? **Pro|ENGINEER** has sophisticated measurement and simulation tools we can use with the component specifications to get an accurate prediction how the finished robot will perform.

## Task one - Wheels

Let's take a look at a single wheel and work out the relationship between diameter and distance travelled.

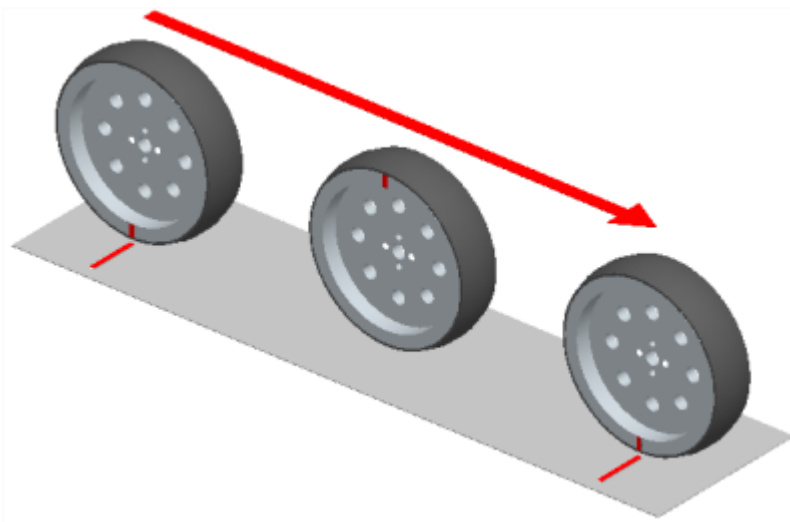
### Speed and distance

Find the large wheel in the Tetrax™ kit and guess how far it will travel in one revolution. The wheel traveled the distance around the circumference of the wheel.



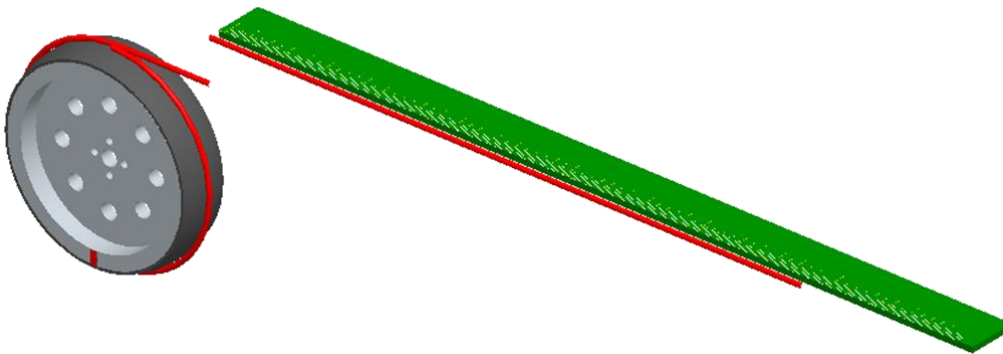
There are several ways we can find this out;

### Experiment



1. Make a mark on the rim of the wheel.
2. Place the mark over a line on a piece of paper.
3. Carefully roll the wheel in a straight line along the paper until the mark on the rim reaches the bottom again.
4. Make another mark on the paper.
5. Measure the distance between the two marks on the paper. This is the circumference of the wheel.

**Measuring**



- 6. Wrap a flexible tape measure around the outside of the wheel. If you don't have one of these use a piece of string then measure it with a normal ruler.

**Geometry**

- 7. Measure the diameter of the wheel using calipers then use the formula:

$$\text{Circumference} = \pi \times \text{diameter}$$

For the large Tetrax™ wheel the diameter is 4” or 100mm so:

Metric (mm)	ANSI (inches)
Circumference = 3.14 x 100 mm	Circumference = 3.14 x 4”
Circumference = <b>314 mm</b>	Circumference = <b>12.56”</b>

**Velocity across the floor**

We can now combine this information with the speed of rotation from briefing five to find out how far the robot will travel each second. To do this we will use the formula:

$$\text{Velocity along the floor} = \text{Circumference of tire} \times \text{Revolutions}$$

The wheel revolves 454.5 degrees per second which is:

$$\frac{454.5}{360} = \mathbf{1.26} \text{ revolutions per second}$$

So the distance travelled in one second is:

Metric (mm)	ANSI (inches)
V = 314 x 1.26 mm	V = 12.56 x 1.26 mm
V = <b>395.64 mm</b>	V = <b>15.83 inches</b>
<b>Velocity = 400 mm sec<sup>-1</sup></b>	<b>Velocity = 16 in sec<sup>-1</sup></b>

### Distance travelled

We can now create a formula to predict how long we need to run the motor for to travel a certain distance. This will help us plan accurate programs for the electronic controller.

Time (**T** sec) required to cover the required distance (**D** mm/inch)

Metric (mm)	ANSI (inches)
$T = \frac{D}{395.64}$	$T = \frac{D}{15.83}$
E.g. to travel a distance of 1500mm:	E.g. to travel a distance of 54”:
$T = \frac{1500}{395.64}$	$T = \frac{54}{15.83}$
<b>T = 3.79 seconds</b>	<b>T = 3.41 seconds</b>

### In practice!

Theory is all very well but there are several reasons why the robot may not behave exactly as planned including; friction, motor variations, inertia, floor surface variations, etc.

### Task two - Friction

**Definition:** ‘Friction is a force that opposes motion’.

- Friction is greater for rough surfaces
- Friction varies according to the materials e.g. low friction materials like PTFE are used for some bearings and high friction rubber is often used for tires.
- Friction increases as the pressure between surfaces increases – e.g. axles that are tight in a bearing; heavy vehicles.
- Friction is not affected by the area of contact between materials.
- Coefficient of friction is the unit of measurement and is different for each pair of materials and the surface finish of both.

Friction coefficients depend on the type of materials in contact. Tables of typical friction coefficients are found in physics and engineering texts.

There are two types of friction:

- The **coefficient of static friction ( $\mu_s$ )** describes the amount of energy that is needed to initiate movement in your model.
- The **coefficient of kinetic friction ( $\mu_k$ )** describes the amount of energy that is lost to friction while keeping your model in motion.





**Note:** The coefficient of static friction for two surfaces must be equal or larger than the coefficient of kinetic friction for the same two surfaces.

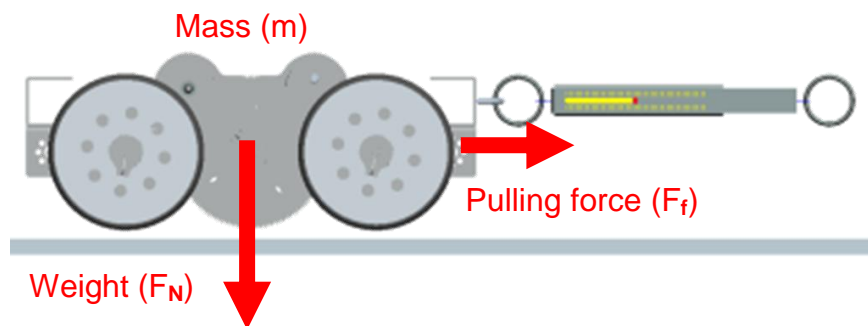
### Friction applied to FTC

Pro|ENGINEER can simulate the friction between the wheels and floor but first we need to know the coefficient of friction between the robot wheel and the ramp surface.

#### Coefficient of friction

Coefficient of friction is an empirical value and so can only be found by experiment.

This is easily found by measuring the force needed to pull the robot along a horizontal floor surface combined with the mass of your robot. This illustration shows a force meter being used to pull the robot.



**Force to keep the robot moving** – Readings should be taken when the robot is moving steadily across the surface.

**Force to start the robot moving** – This is harder to measure unless you have a force meter with a marker which records the maximum reading or data logging equipment with force sensors to measure the two values for coefficient of friction.

With these values we can now calculate the coefficients of friction for our robot.

**Example:** FTC robot pulled along a carpeted floor.

Metric (mm)	ANSI (inches)
Mass of FTC robot = 3 kg	Mass of FTC robot = 6.6 lbs
Weight = Mass x gravity	Weight = Mass x gravity
Weight = 3 x 9.81	Weight = 6.6 x 9.81
Weight ( $F_N$ ) = 29.43 Newtons	Weight ( $F_N$ ) = 64.75 lb f
Pulling force ( $F_f$ ) = 20 Newtons	Pulling force ( $F_f$ ) = 4.4 lb f

<p>Coefficient of Friction (<math>\mu</math>)</p> $F_f = \mu \times F_N$ $20 = \mu \times 29.43$ $\frac{20}{29.43} = \mu$ $\mathbf{0.68} = \mu$	<p>Coefficient of Friction (<math>\mu</math>)</p> $F_f = \mu \times F_N$ $43.164 = \mu \times 64.75$ $\frac{43.164}{64.75} = \mu$ $\mathbf{0.67} = \mu$
---	---

The coefficient of friction will be used next to program the Pro|ENGINEER model when analyzing movement.



**Note:** When simulating your FTC robot, it is very important you *measure the friction on the same material the arena floor is made of.*

The 2009 **FIRST** Robotics Challenge (FRC) arena has a low friction surface to simulate the low gravity moon environment. On low friction surfaces like this the wheels may slip and slide.

If your team enters the FRC you should find out the coefficient of friction between your robot and this low friction surface then use this to analyze the movement of your robot.

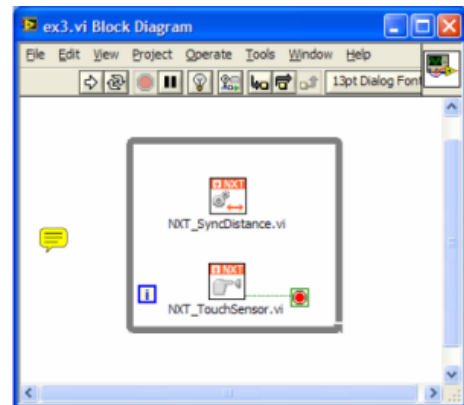
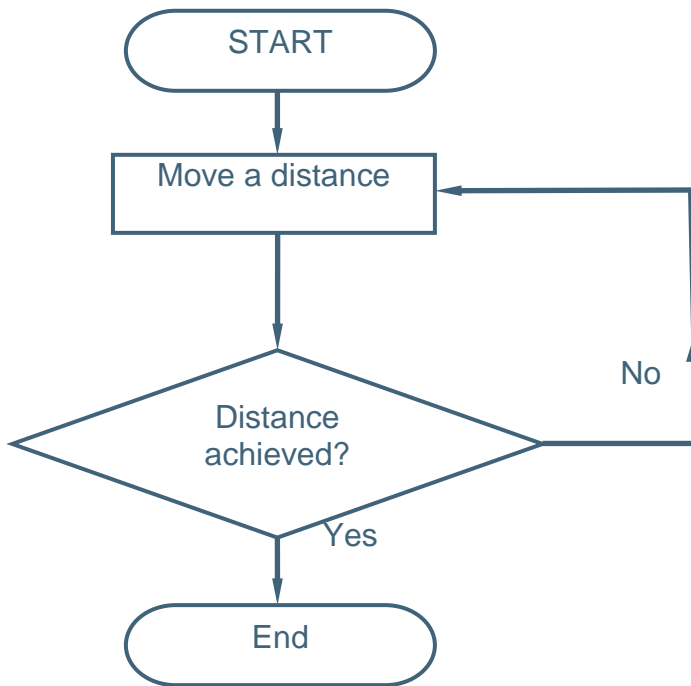
### Friction applied to control

If the wheels of a robot driving over this surface slip would this interfere with control program? Would it help to know how much slip there has been? Could you use feedback sensors on the wheel spindle?

Can we measure velocity across the floor surface? Could you put sensors somewhere else to ensure the correct distance is travelled and build this into the control program?

Example:

Labview



Would it help to know how much slip there has been?

Could you use feedback sensors on the wheel spindle?

Can we measure velocity across the floor surface? Could you put sensors somewhere else to ensure the correct distance is travelled and build this into the control program?

**Moving in a straight line**

No matter how carefully the motors are manufactured the chance that any two run at exactly the same rotational velocity are slim. The visible outcome of this is a robot with tank drive like ours travelling in a slight curve rather than a straight line.

An innovative solution to this can be found in the drive system of a toy sold by Milton Bradley in the 1980s called Big Trak™.

This had permanent magnets on the motor shafts which locked together when the motors were running at near the same speed. More information about these innovative toys can be found on the web.



<http://www.thebigtrak.com/gallery.htm>

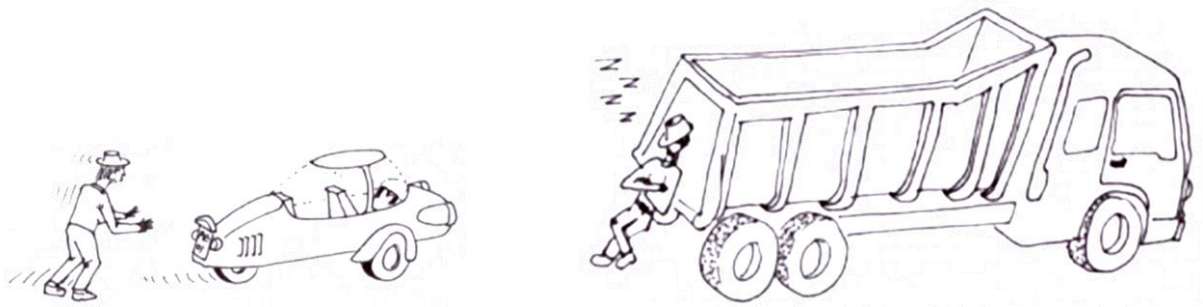
<http://www.thebigtrak.com/legacysites/robotprojects/> - disassembly instructions showing magnetic clutch after step 10.

## Task three - Inertia

**Definition** 'Inertia is an objects resistance to changes in velocity'.

Anything with mass resists being moved or accelerated, this is called inertia. The mass of an object determines how much inertia it has, i.e. the greater the mass of an object the more inertia it has.

Inertia is why engineers try to reduce the mass of objects that move such as pistons, cars, rockets and aircraft.



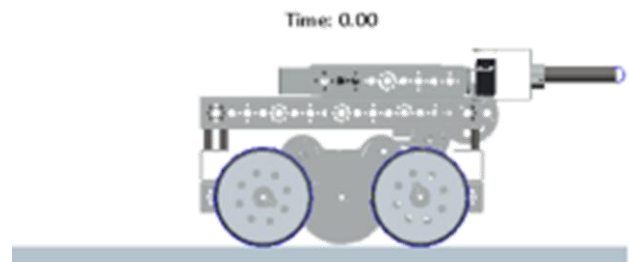
A small (light) car is easy to push... ...while a large (heavy) truck is impossible to push

## Inertia applied to FTC

The FTC robot weighs the same as three bags of sugar so inertia must be considered. After switching on the motors it will take a little while to accelerate to full speed.

We will simulate this using Pro|ENGINEER first using Earth gravity then using Moon gravity to see if this makes a difference to the motion.

1. Pro|ENGINEER should be running with no models on screen or in memory.
2. Set the working directory to **\FTC\_Robot\Inertia**
3. Open the model **INERTIA.ASM**
4. Change the view to **FRONT** and zoom to see this.



The next analysis is configured to last 0.25 seconds with a frame rate of 100 to capture the motion in detail. Cam connections between the wheels and the floor have liftoff (0.5) and friction (0.68) enabled.


5. In the main toolbar open the **Applications** pull-down menu and select **mechanism**.

- In the mechanism tree expand the ANALYSES heading and **Right-click**  over the **Inertia (DYNAMICS)** entry then select **Run**.



**Remember:** The analysis is mathematically complex and does not take place in real time. It may take longer than 0.25 seconds to finish.


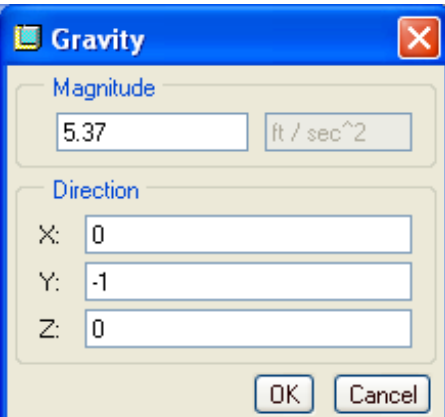
**Saving a playback**


- In the model tree, listed under **PLAYBACKS** you will now see a new entry.
- Right-click**  over the new entry and, from the floating menu, select **Save**.
- Type a **name** for the playback and click **Save**.
- Replay** the analysis and **zoom** in to look at the first part of the simulation. You will see the wheel spin for a short time until the robot is moving.

We will now change the force of gravity acting on the model.

Metric (mm)	ANSI (inches)
Earth (g) = 9.806.65 m sec <sup>2</sup>	Earth (g) = 32.2 ft sec <sup>2</sup> = 386 in sec <sup>2</sup>
Moon (g) = 1.635 m sec <sup>2</sup>	Moon (g) = 5.37 ft sec <sup>2</sup> = 64.44 in sec <sup>2</sup>

- In the mechanism tree **Right-click**  over the  **Gravity** entry and, from the floating menu, select **Edit Definition**.

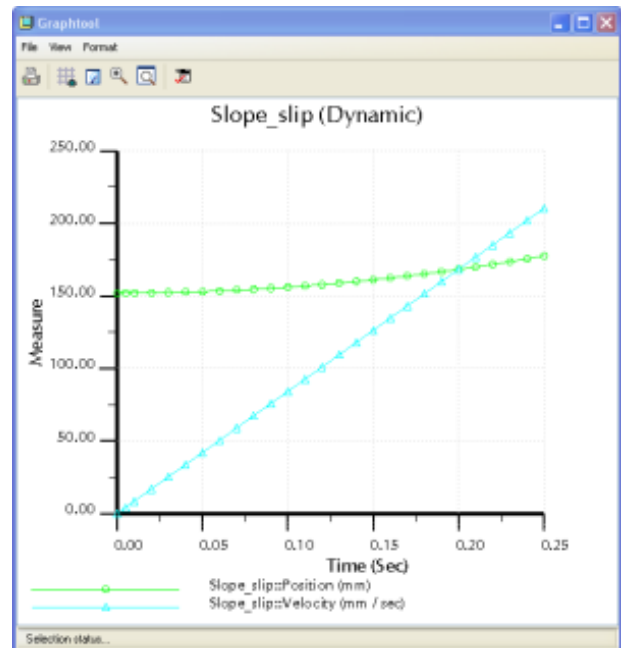
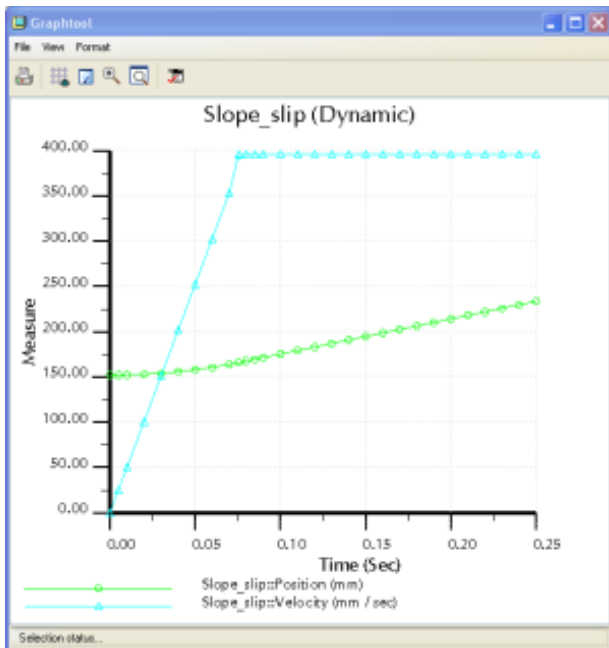
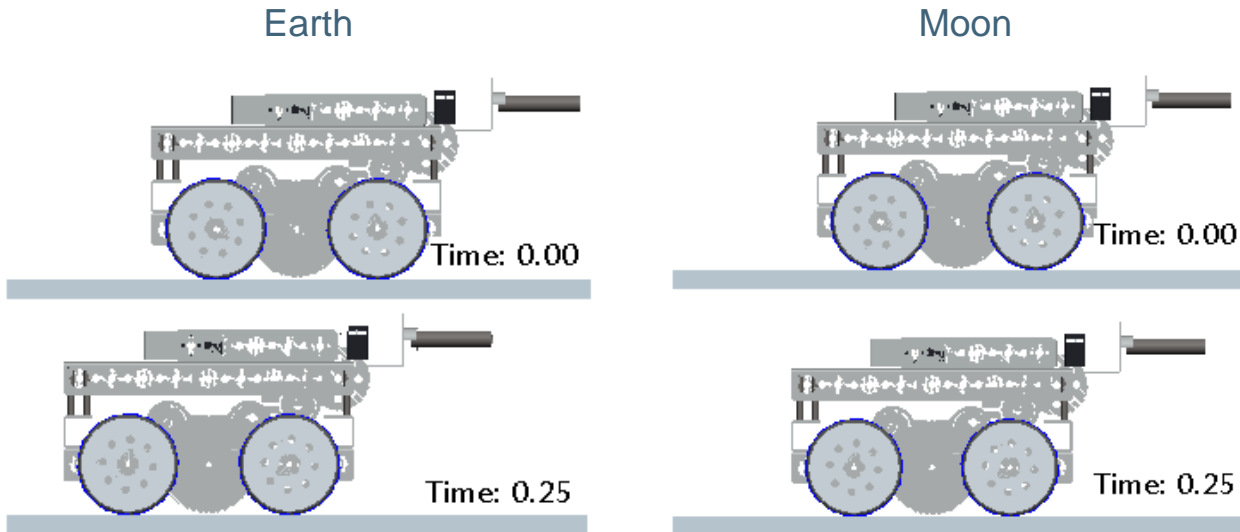
Metric (mm)	ANSI (inches)
<p>12. Change the <b>Magnitude</b> to 1635 then click <input type="button" value="OK"/>.</p> 	<p>8. Check which units gravity is being measured in then change the <b>Magnitude</b> to 5.37 or 64.44 then click <input type="button" value="OK"/>.</p> 

- In the mechanism tree **Right-click**  over the **Inertia (DYNAMICS)** entry and select **Run**.

14. **Save** the Playback.
15. **Replay** the analysis and **zoom** in to look at the first part of the simulation. You will see the wheel spin for all of the analysis with the robot moving a much shorter distance.



Below is a graphic representation of the total movement in 0.25 seconds plus graphs in **Pro|ENGINEER** measuring the position and velocity of the robot.



On the left we have Earth gravity where the robot accelerates for a little over 0.05 seconds reaching full velocity of 400 mm sec<sup>2</sup> for the rest of the analysis.

On the right, using Moon gravity, the wheels were still slipping and the robot still accelerating when the analysis ended at 0.25 seconds. The maximum velocity at the end of the Moon analysis was only  $160 \text{ mm sec}^2$  and still increasing.



**Note:** The vertical scale is different between the two graphs. The maximum value on the Earth (left) is  $400 \text{ mm sec}^2$  whereas on the Moon (right) the maximum velocity is  $250 \text{ mm sec}^2$ .

16. **Close** all dialogs and models then **Erase not displayed**.

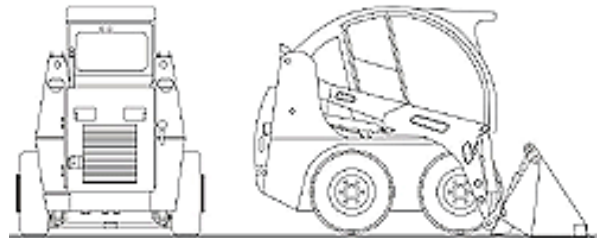
## Task four – Turning

Turning a robot of this design can be done in two ways.

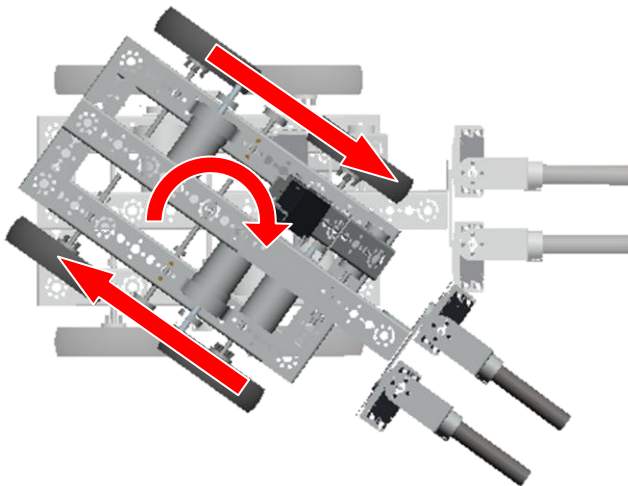
- **Spin** on the spot with wheels driving in opposite directions.
- **Turn** with the wheels on one side driving and the other side stationary.

### Spin

‘Skid steer’ is the term used to describe turning with a robot of this type. If the wheels on one side are turning forwards while the other side is in reverse the robot will spin on its axis.




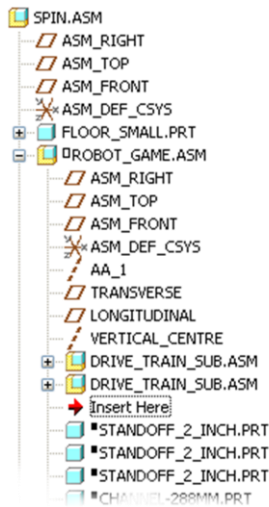
<http://www.terexvectra.com/>



We can use math to work out the geometry of the arrangement and how long we have to turn on the motors to achieve a specific angle spin. First we need to use the measurement tools in **Pro|ENGINEER** to find the track width of the robot.

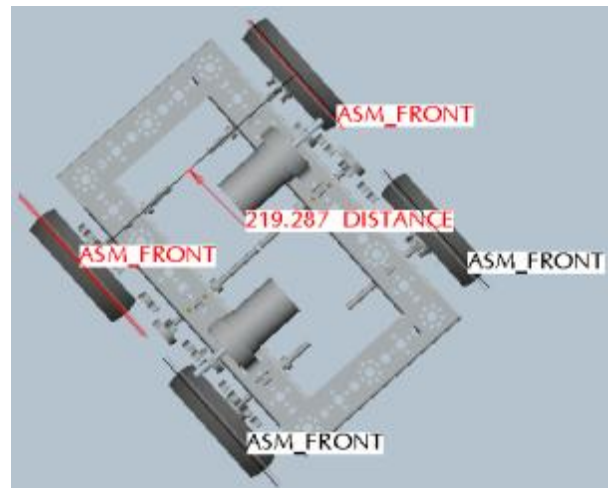
1. **Pro|ENGINEER** should be running with no models on screen or in memory.
2. Set the working directory to **\FTC\_Robot\Turn**

3. Open the model **Spin.asm**
4. Change the view to **TOP**
5. In the model tree expand the entry for **ROBOT\_GAME.ASM**
6. Drag the  **Insert Here** flag to below the two **DRIVE\_TRAIN\_SUB.ASM** entries



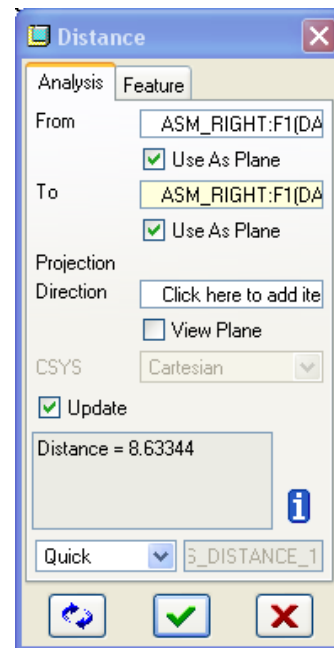
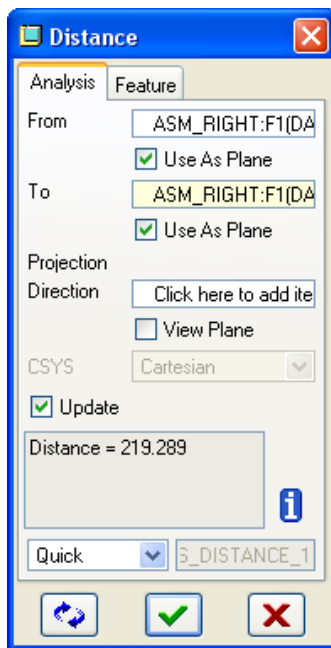
7. In the main toolbar across the top of the screen open the **Analysis** pull-down text menu and select **Measure > Distance**
8. Make datum planes visible and all other datum features invisible
9. Select the **ASM\_RIGHT** datum planes for one pair of wheels on opposite sides of the robot shown here in red.


The distance will be shown on the model and a dialog will open reporting the selected geometry and the track of the robot.



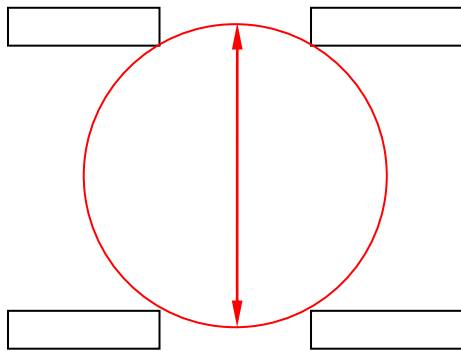
<b>Metric (mm)</b>	<b>ANSI (inches)</b>
--------------------	----------------------





10. Click  to close the **Distance** dialog.

With an accurate measurement of the track width we can now calculate the circumference of the circle traced by the wheels when the robot spins.



Circumference =  $\pi$  x diameter

Metric (mm)	ANSI (inches)
Circumference = 3.14 x 219.29	Circumference = 3.14 x 8.63
Circumference = <b>688.57</b> mm	Circumference = <b>27.1</b> inches

With this information we can work out how long the motors need to be on to turn the robot a set number of degrees. We know how far the wheels travel in one second so how many degrees turn does this represent?

Metric (mm)	ANSI (inches)
<b>Robot velocity= 395.64 mm sec<sup>-1</sup></b>	<b>Velocity = 15.83 in sec<sup>-1</sup></b>
Rotate = $\frac{395.64}{688.57}$	Rotate = $\frac{15.83}{27.1}$
Rotate = 0.57 of a turn	Rotate = 0.58 of a turn
Rotate = 0.57 x 360	Rotate = 0.58 x 360
<b>Robot rotate = 205 degrees</b>	<b>Robot rotate = 210 degrees</b>

Why do you think there is a difference in the number of degrees in each column when the robot is identical?

Look back to the diameter used for the wheels in each column. Is 4 inches exactly 100mm? Use an accurate measure like vernier calipers to check the actual diameter of the wheels. You may want to substitute this in the previous formulae to get a more precise outcome for your exact design.

### Degrees of turn

We can now create a formula to predict how long we need to run the motors in opposite directions to spin the robot a set number of degrees. This will help us plan accurate programs for the electronic controller.

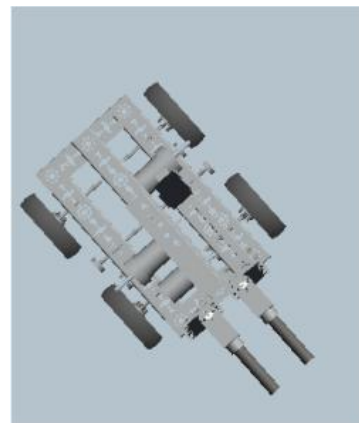
Time (T secs) required to spin the required R (degrees)

Metric (mm)	ANSI (inches)
$T = \frac{R}{205}$	$T = \frac{R}{210}$
E.g. to spin 180 degrees:	E.g. to spin 180 degrees:
$T = \frac{180}{205}$	$T = \frac{180}{210}$
$T = \mathbf{0.88}$ seconds	$T = \mathbf{0.86}$ seconds

Spin with Pro|ENGINEER

This model has cam connections between the wheels and the floor with liftoff enabled (0.5) and static/kinetic friction enabled (1). The analysis runs the wheels at a constant 454 degrees per second for 0.44 which according to the calculations above should spin the robot through 90 degrees.

11. Pro|ENGINEER should still be running the model **SPIN.ASM** open on screen.
12. Change the view to **Top** to see this.
13. In the main toolbar open the **Applications** drop-down menu and select **Mechanism**.

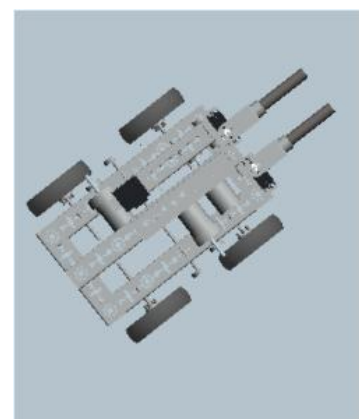


14. In the mechanism tree, expand **ANALYSES** and run the analysis **Spin\_on\_spot (DYNAMIC)**


The robot will turn to this position.

If we were to measure the angle before and after the analysis we would find the angle of spin is 86 degrees, very close to the 90 degrees calculated previously.

The cam connections in the assembly use a value of 1 for coefficient of friction so it should be no surprise the simulation so closely matches theory. Next you will alter the value to 0.68.



What do you think will happen when the simulation is re-run?

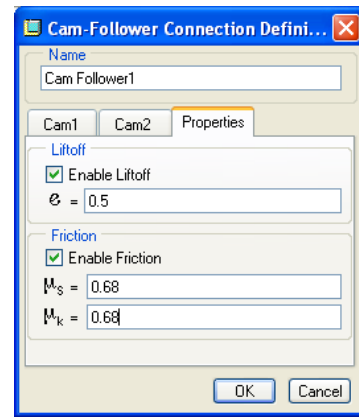
15. In the mechanism tree, expand the **CAMS** heading, then **Right-click**  over **Follower1 (Spin)** and, from the floating menu, select **Edit Definition**.

16. Activate the **Properties** tab and change both Friction values to **0.68**. then  to close.



**Note:** This was the coefficient of friction found previously by experiment.

17. Repeat this to **change** the coefficient of friction for the **other three cam connections**.

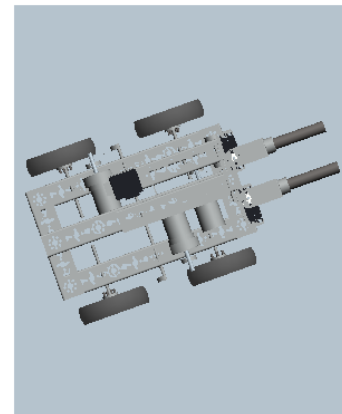


18. Re run the analysis **Follower1 (Spin)**.

The robot model has not turned by the same amount.

Measuring the rotation would show the finishing position is 67 degrees from the start.

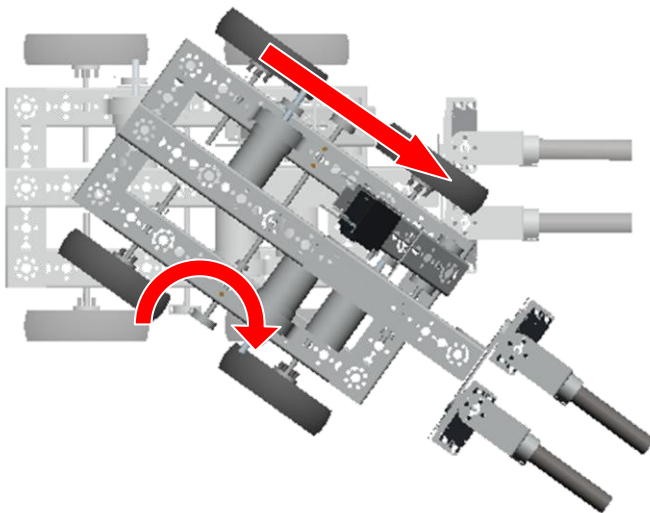
Why do you think there is a difference?



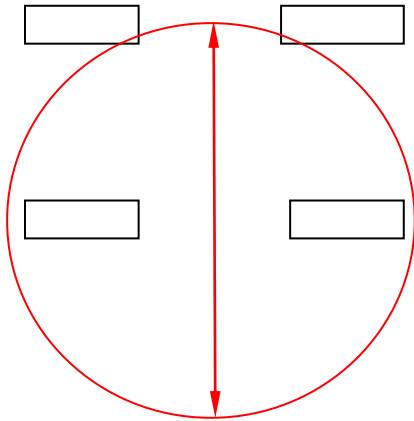
This shows how accurately Pro|ENGINEER simulates mechanical systems and underlines why Pro|ENGINEER is the leading 3D parametric design software in the world.

### Turn with Pro|ENGINEER

If the wheels on one side are turning while the other side is switched off the robot will turn around the fixed set of wheels rather than spin.



The first thing we need to know is the circumference on the circle traced by the wheels when the robot spins.



Circumference =  $\pi$  x diameter

Metric (mm)	ANSI (inches)
Circumference = 3.14 x 438.58	Circumference = 3.14 x 17.26
Circumference = <b>1377.14 mm</b>	Circumference = <b>54.2 inches</b>

With this information we can work out how long the motors need to be on to turn the robot a set number of degrees. We know how far the wheels travel in one second so how many degrees turn does this represent?

Metric (mm)	ANSI (inches)
Rotate = $\frac{395.64}{1377.14}$	Rotate = $\frac{15.83}{54.2}$
Rotate = 0.29 of a turn	Rotate = 0.29 of a turn
Rotate = 0.29 x 360	Rotate = 0.29 x 360
<b>Rotate = 104 degrees per second</b>	<b>Rotate =104 degrees per second</b>

Unlike the previous spin calculations, the results for each of these columns are identical. Why do you think this is the case? We rounded the values to two decimal places and by coincidence this has brought the two results together.

This highlights the question common in all forms of design and engineering of how precise do we need to be. Increased precision usually costs time and/or money.

The information provided by these calculations will be used to program the robot but when it is running there may be bumps in the floor, slippery patches and the motors may be operating slightly faster or slower. With this in mind we only need calculations within a few percent of the required value. Fine tuning can be done once the robot is operating.

### Degrees of turn

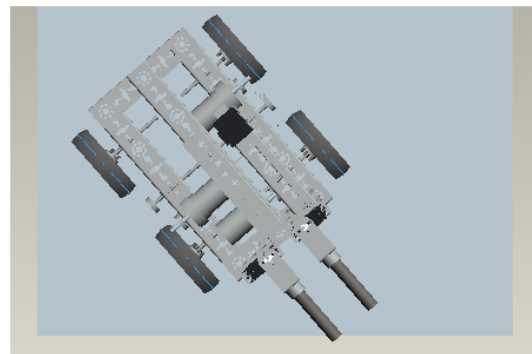
From this information we can now create a formula to predict how long we need to run the motors on one side to turn the robot a set number of degrees. This will help us plan accurate programs for the electronic controller.

Time (T secs) required to turn through A (degrees)

Metric (mm)	ANSI (inches)
$T = \frac{A}{104}$	$T = \frac{A}{104}$
E.g. to spin 180 degrees:	E.g. to spin 180 degrees:
$T = \frac{180}{104}$	$T = \frac{180}{104}$
T = <b>1.73</b> seconds	T = <b>1.73</b> seconds

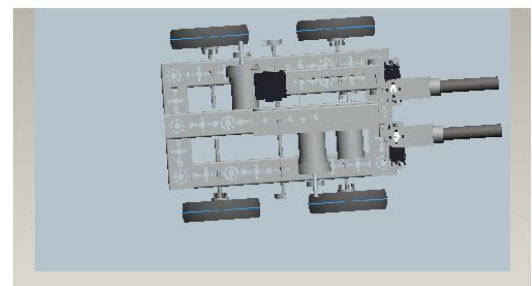
### Turn with Pro|ENGINEER

19. Pro|ENGINEER should be running with no models on screen or in memory.
20. Set the working directory to **\\FTC\_Robot\Turn**
21. Open the model **TURN.ASM**
22. Change the view to **Top** to see this.
23. In the main toolbar open the **Applications** drop-down menu and select **Mechanism**.
  
24. In the mechanism tree, expand **ANALYSES** and run the analysis **Turn (DYNAMICS)**



The robot will turn to this position.

If we were to measure the angle before and after the analysis we would find the angle of turn is 52 degrees which is more than the calculated value of 45 degrees!



The cam connections in the assembly use a value of 0.68 for coefficient of friction so we should expect there to be some wheel spin and the turn to be less.

We are assuming the axis of turn is perfectly central between the stationary wheels. Due to the center of gravity not being central the axis of turn is likely to be offset.

25. **Close** all models and dialogs then **Erase not displayed** to empty memory.

## Task five - Inclines

So far we have assumed a level playing field. The arena has a range of surfaces your robot must navigate and one of the features your robot must negotiate is a ramp.

### Ramp

Two important things will determine how well your robot copes with a ramp; friction between the wheels and the ramp surface, and centre of gravity (c of g). Both of these we can simulate using **Pro|ENGINEER**.

### Slope climb - Friction

Earlier in this tutorial you conducted an experiment to measure the force needed to pull the robot along a surface and used this to find the coefficient of friction for those two surfaces. You will now use this to analyze the slope climbing ability of the FTC robot.

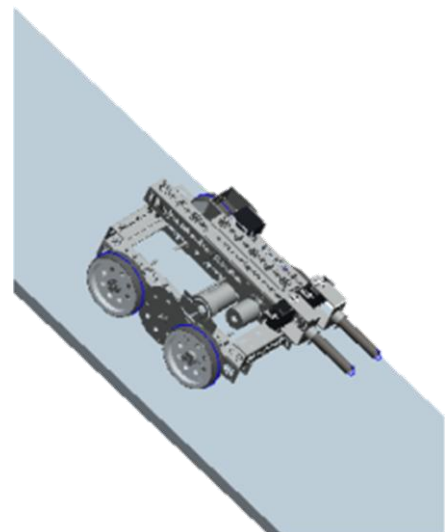
The next **Pro|ENGINEER** model has flat floor inclined at an angle of 25 degrees with the FTC robot half way up.

Cam connections exist between the wheels and the surface of the slope. Liftoff and friction are enabled to simulate the point when the wheels lose grip and the robot should slide backwards down the slope.

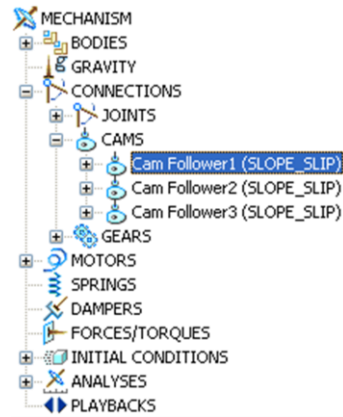
1. **Pro|ENGINEER** should be running with no models on screen or in memory.
2. Set the working directory to **\FTC\_Robot\Slope climb\Slope\_slip**
3. Open the model **SLOPE\_SLIP.ASM**
4. Change the view to **Trimetric** and **zoom** to see this.

Initially the slope is set at 25 degrees.

We don't know whether it will climb or slip backwards so the robot is placed half way up the slope.



5. In the main toolbar open the **Applications** menu and select **Mechanisms**.
6. Expand **CONNECTIONS** and the **CAMS** group then **Right-click** over the first **Cam Follower** entry and from the floating menu choose **Edit definition**.
7. The Cam-Follower... dialog will open.



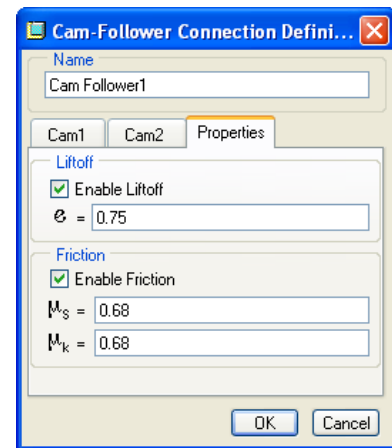
The first two tabs store the geometry selections for the two halves of the cam connection.

The third tab, shown here, stores the values for Liftoff, and coefficient of friction.

### Liftoff

The official term for this is 'coefficient of restitution' or how much bounce there is between surfaces. Typical values are:

- Range = 0 - 1
- Rubber ball = 0.9
- Soft clay = 0.1



The coefficient of restitution for the FTC tire is:

**FTC Rubber tire = 0.75**

### Coefficient of friction

How easy it is to slide two surfaces across each other. From the previous section you will know there are two values for coefficient of friction; Static ( $\mu_s$ ) and Kinetic ( $\mu_k$ ).

Remember, coefficient of friction is an empirical value and so can only be found by experiment.

Typical values plus the value for the FTC tire found earlier in this tutorial are:

- Range = 0 - 1<sup>1</sup>
- Silicone rubber = 0.9
- PTFE (Teflon) = 0.05


<sup>1</sup> In theory, coefficient of friction cannot be greater than 1. In practice, certain materials like some silicon rubbers, dragster tires and adhesive tape can have a coefficient of friction higher than 1.



The coefficient of friction for the FTC tire is:

**FTC Rubber tire = 0.68**

8. Click **OK** to close the Cam-Follower... dialog

9. In the mechanism tree expand  **ANALYSES** then **Right-click** the **Slope\_slip (Dynamics)** entry.

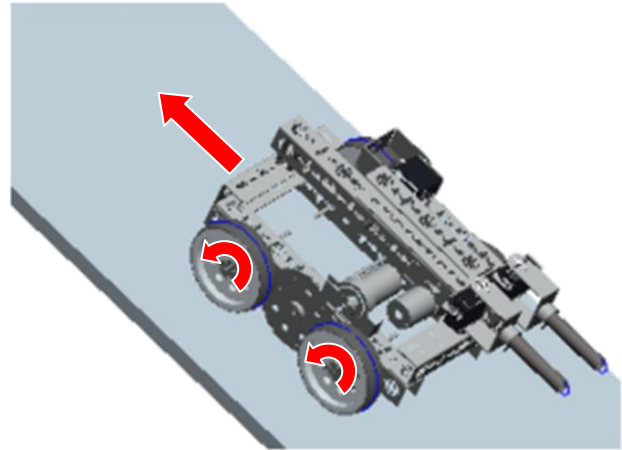
10. From the floating menu, select **Run**.


Pro|ENGINEER will use mathematics to calculate the vectors and forces acting on the model and store the movements for later playback. You should see the progress of the one-second analysis reported along the bottom of the screen **but not in real-time**.



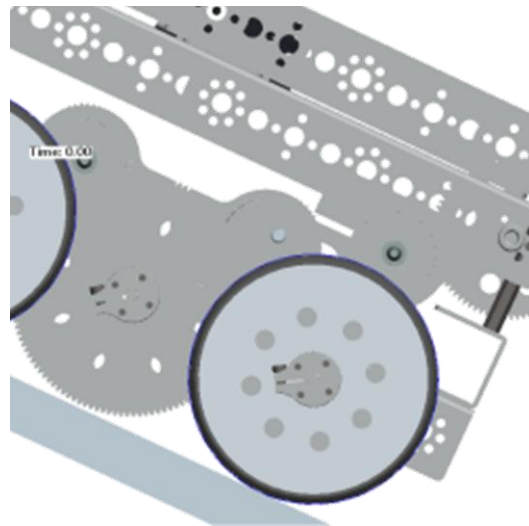
**Note:** The analysis may take quite a time to complete.

The robot will move forward up the incline driven by the wheels.




11. In the mechanism tree expand the **PLAYBACK** heading, **Right-click**  over the **Slope\_slip** entry and, from the floating menu, select **Play**.

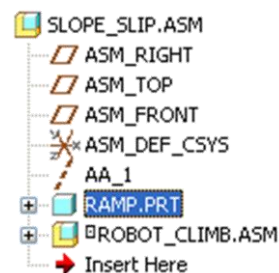
12. **Re-play** the analysis zooming in on the start to see if there is any wheel spin or wheelie.




### Increase slope angle

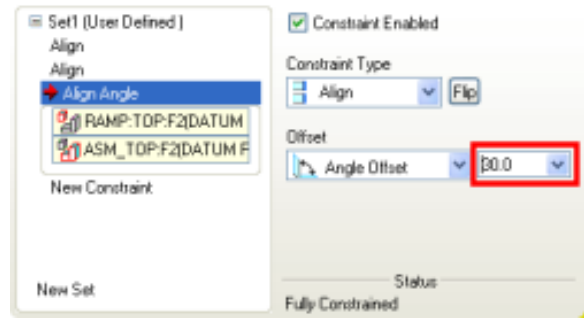
We can now increase the slope angle in the original assembly and then re-run the analysis.

13. In the model tree, **Right-click**  over **RAMP.PRT** and from the floating menu select **Edit Definition**.




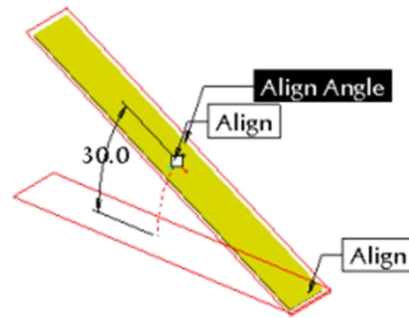
14. In the Assembly dashboard, open the **Placement** tab and **Left Click**  to select **Align Angle**.

15. Change the **Offset** value to **30** degrees.




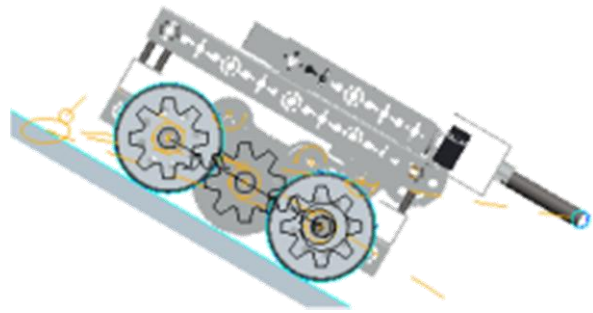
The angle of the amp will change in the assembly.

16. Click on  to close the dashboard.



17. Change the view to **FRONT**.

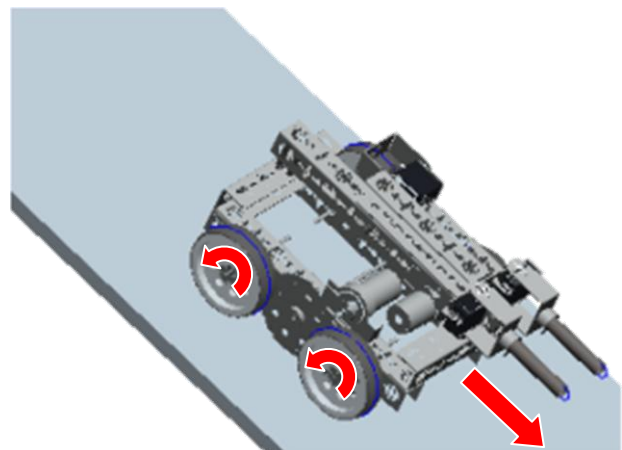
18. Hold down **Ctrl + Alt** and hold **Middle mouse**  and **drag** the large center gear to reposition the robot on the ramp near the middle.



19. In the mechanism **Right-click**  the **Slope\_slip (Dynamics)** entry.

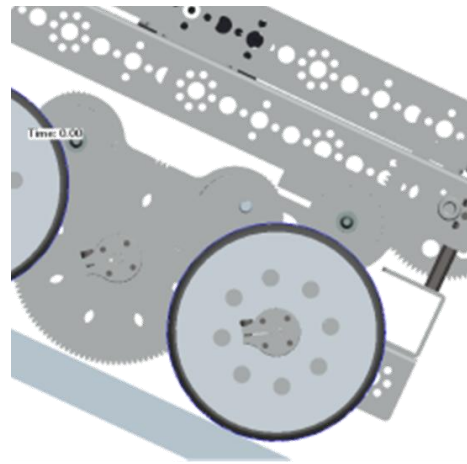
20. From the floating menu, select Run.

This time the **wheels on the robot** will rotate trying to move up the ramp but the robot will slide backwards.




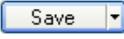
21. Use the **PLAYBACK** again to **re-play** the analysis.

This time closely examine the start of the analysis to see the robot accelerate down the slope despite the wheels trying to climb.






### Saving playbacks

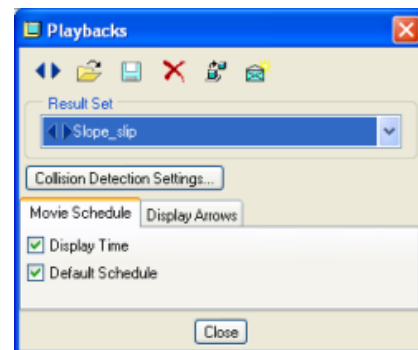
You may have noticed Playbacks are lost when a model is closed and reopened.

22. Move the mouse cursor over the PLAYBACK entry **Slope\_slip** and **Right-click** .
23. From the floating menu select **Save**.
24. Complete the dialog and click on . By default the playback will be saved in the working directory.

### Opening playbacks

Next time you open the assembly;



25. Change to **Mechanisms**
26. Right click over PLAYBACKS then select **Play...**
27. At the top of the Playbacks dialog click on  then select the saved playback file and click .
28. In the Playbacks dialog click on  to open the playback controls panel.

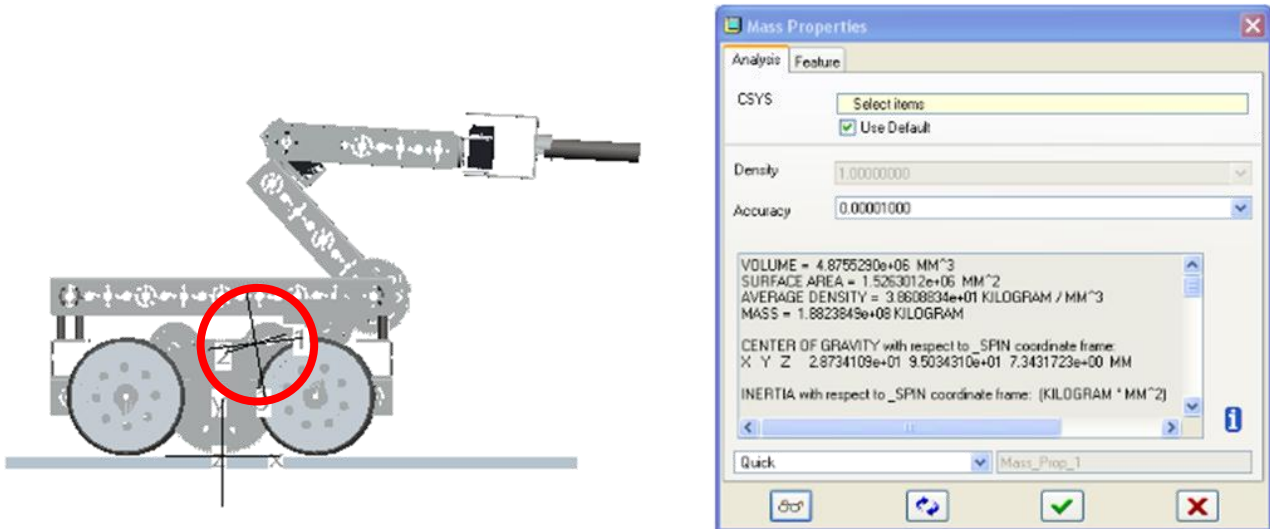


### Task six - Centre of gravity

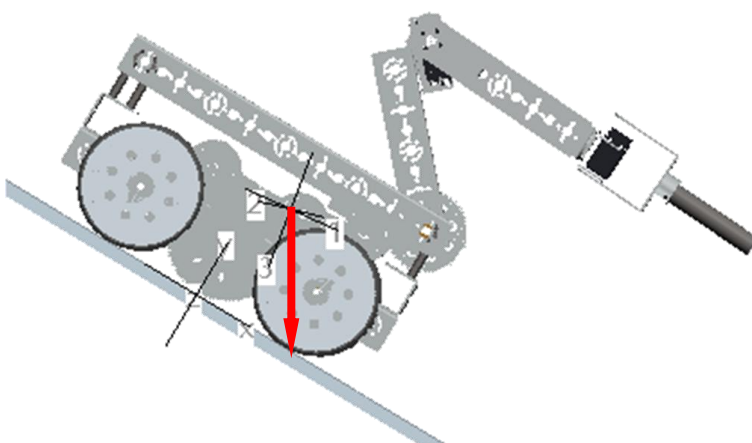
Briefing three mentioned c of g when talking about stability. The c of g is an imaginary point where the mass of the object can be considered to act. Pro|ENGINEER can tell us exactly where the c of g is in a model.

1. Pro|ENGINEER should be running with no models on screen or in memory.
2. Set the working directory to **\\FTC\_Robot\Slope climb\Slope\_tumble**

3. Open the model **SLOPE\_TUMBLE.ASM**
4. Change the view to **FRONT**.
5. In the model tree, move the mouse cursor over the entry **ROBOT\_CLIMB.ASM** then **Right-click**  and, from the floating menu, select **Activate**.
6. In the main toolbar open the **Analysis** pull-down text menu choose **Model** then **Mass Properties**. The Mass Properties dialog opens.
7. **Select**  to carry out an analysis. The dialog now contains a range of information and the model displays a coordinate system showing visually the c of g position here outlined in red.



Notice the c of g is not in the centre between the two wheels? Why do you think this is?  
 Gravity acts downwards so if the c of g moves outside the base, the robot will topple over. This is shown in the diagram below with the robot on a slope and about to topple over.



Gravity acting downwards from the c of g is in line with where the lower wheel contacts the ramp.

If the robot were facing up the ramp would the model be more or less stable? The position of the robot arm will affect the c of g and needs to be considered when tackling this obstacle.

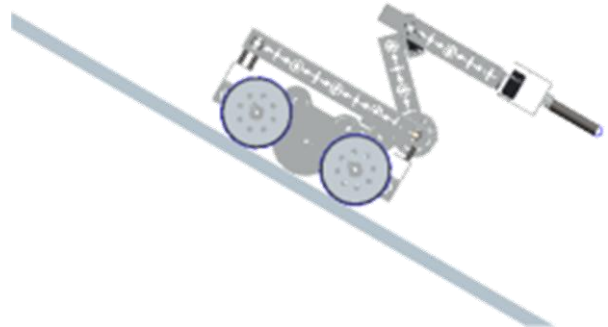
### Slope climb - Tumble


The mechanism dynamics tools in **Pro|ENGINEER** will now be used to demonstrate stability and find the steepest slope the FTC robot will climb.

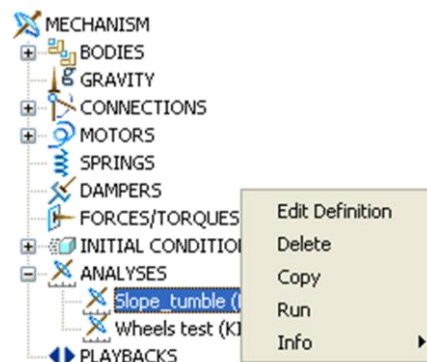
The slope has been set at an angle of 30 degrees.

The robot model is very similar to the previous example with cam connections between the tires and the slope surface with liftoff and friction enabled. The coefficient of friction for the cam connections has been set to 1.

8. What effect do you think this will have on the simulation?
9. Change the view to **FRONT** and zoom in like this.

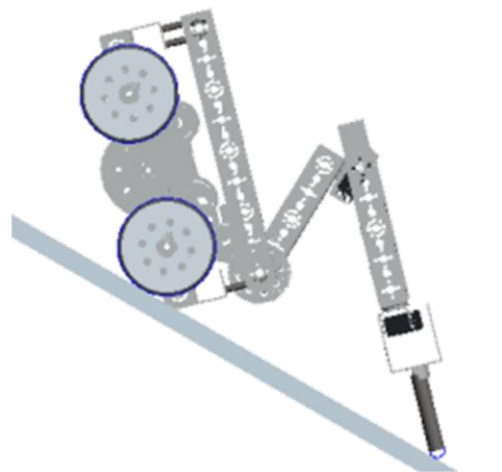


10. In the main toolbar open the **Applications** menu and select **Mechanisms**.
11. In the mechanism tree expand  **ANALYSES** then **Right-click** the **Slope\_tumble (DYNAMICS)** entry.
12. From the floating menu, select **Run**.



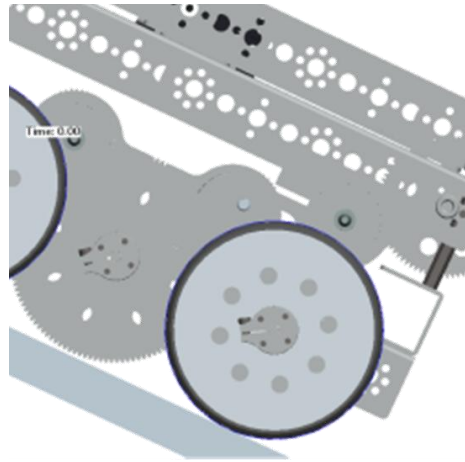
**Pro|ENGINEER** will use mathematics to calculate the vectors and forces acting on the model and store the movements for later playback. You should see the progress of the two-second analysis reported along the bottom of the screen **but not in real-time**.

The robot will begin to move forwards up the incline driven by the wheels. The robot will wheelie until the gripper contacts the floor and, after some bouncing, the robot will climb the slope on its rear wheels with the gripper dragging behind.



13. Use the **PLAYBACK** again to **re-play** the analysis.

Closely examine how the gripper contacts the slope.



#### Optional extension

Explore the coefficient of restitution (liftoff) between the gripper and the slope.

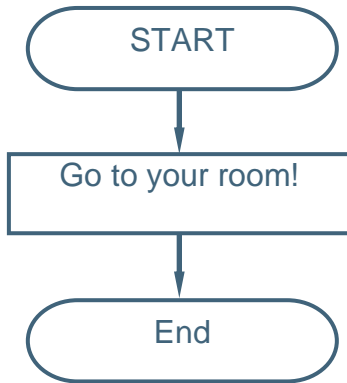
- Change Cam Follower 3 connection (gripper end to slope) Properties > Liftoff ( ) to 0.25
- Re run the analysis
- Look at the playback to observe how the gripper now bounces on the floor.

This briefing is now complete.

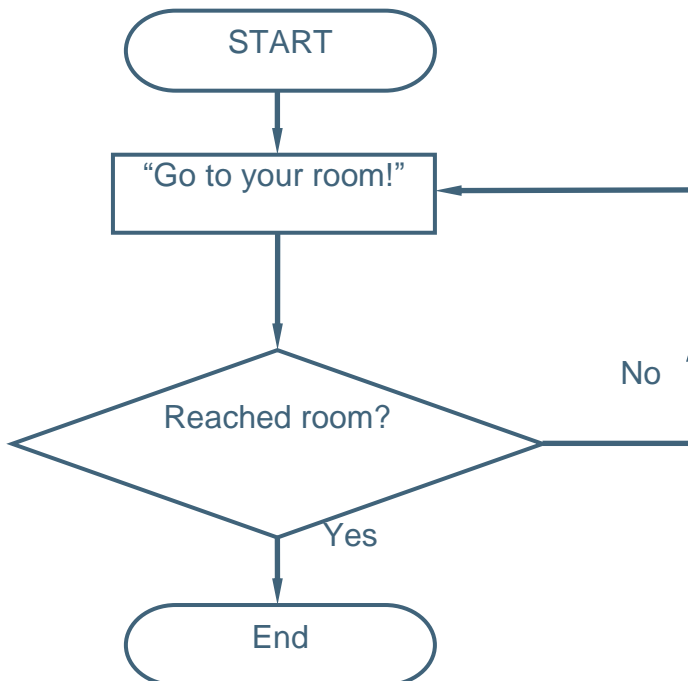
## Briefing seven - Systems and control

Control systems can be divided into two main types; open loop and closed loop.

Imagine your parents have told you to go to your bedroom. You leave the living room but they cannot be sure you have reached your room; instead you may have gone to the bathroom. This is open loop control.



Imagine there was a video camera in your bedroom giving your parents feedback on whether you had reached there? They might keep issuing instructions until you comply!

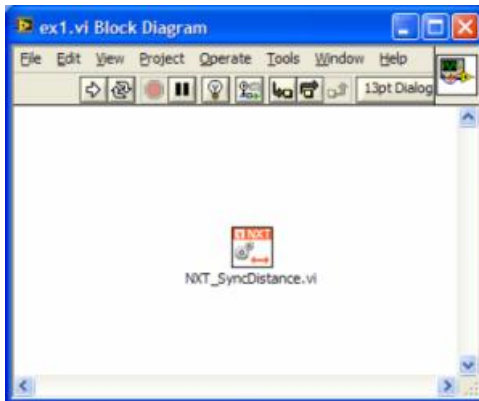


Aren't you glad parents don't use feedback all the time?

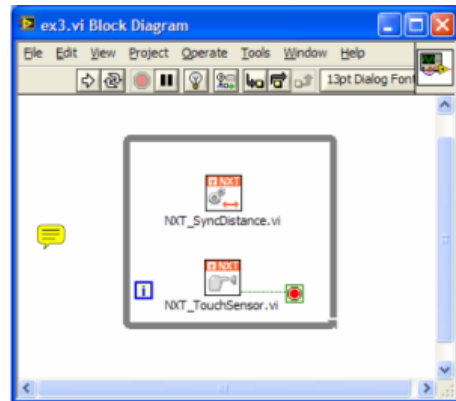
## Labview

The labview control programs to achieve the examples above look like this.

Open loop



Closed loop



These examples are reproduced with the kind permission of Enable Training and Consulting, Inc. Their LV Mastery courses are available online to all FIRST Tech Challenge Teams and cover all aspects of the LabVIEW control systems.

Teams can visit [www.LVMastery.com/Curriculum](http://www.LVMastery.com/Curriculum) to register and find out more.

This briefing is now complete.



## Briefing eight - Engineering drawings

If you would like to learn about engineering drawings, often referred to as 'detailing' take the following **Online Courses** in PTC University Coach: <http://schools.ptculms.com/>

- Detailing
- Advanced detailing

In addition, PTC's schools curriculum contains projects covering aspects of detailing/ engineering drawing including:

- Sports drink bottle (Engineering drawings)
- F1 in schools D Type (Engineering drawings)

### Focus:

An effective method of communicating designs to other people is via the use of drawings. **Pro|ENGINEER** allows designers and engineers to quickly produce engineering production drawings directly from the solid model.

To help engineers interpret drawings anywhere in the world standards for presentation have been created. Historically every country had its own set of standards such as British Standards but these have been confined to the major continents. Examples include ANSI from North America and ISO, widely used in Europe and the Far East. In future we may end up with one set of global drawing standards. **Pro|ENGINEER** can format drawings for any international standard.

In the past, paper drawings have been the traditional, manual method of communicating product design information but the use of solid modelling has allowed a more direct and automated link.

Using drawings requires the engineers to interpret 2D orthogonal views whereas the 3D solid model contains more information and is easier to visualise. The use of Computer Numerically Controlled (CNC) machines now allows engineers to produce components directly from the solid model.



This level of automation means that orthographic drawings are now mainly used to provide overall dimensions, assembly details and inspection information.

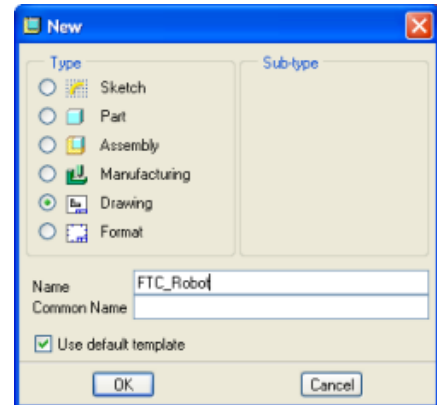
During this briefing you will have access to a PC running **Pro|ENGINEER** Wildfire 4.0 and learn how to create engineering drawings from an FTC robot including dimensions and annotations.

### Drawings and FTC


A key role for drawings in FTC is to communicate the design and operation of your robot to the judges. We will focus on this to produce a general assembly drawing with orthographic views containing key dimensions and a pictorial representation.

## Task one - Creating a drawing

1. Start Pro|ENGINEER.
2. Set the working directory to where your own FTC robot design is located.
3. Open your FTC robot assembly in Pro|ENGINEER.
4. From the Pro|ENGINEER top toolbar left-click **Create New File** . In the dialog box that appears the default **Type** is **Part**, left-click **Drawing**.
5. Enter the name of your robot. We will use **FTC\_Robot**.
6. Make sure **Use default template** is checked then click .



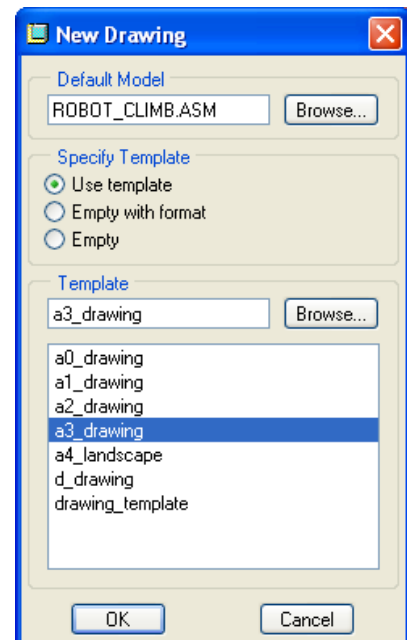
The New Drawing dialog opens.

7. Your car assembly should be listed in the **Default Model** field.
8. Make sure the **Use template** option is checked.
9. **Select** the paper size/template you wish to use for the drawing then **click** .

Pro|ENGINEER will create a drawing with drawing border/format and three orthographic views




**Note:** The toolbar on the right of the screen has changed to display the commands and options relevant to drawing creation.

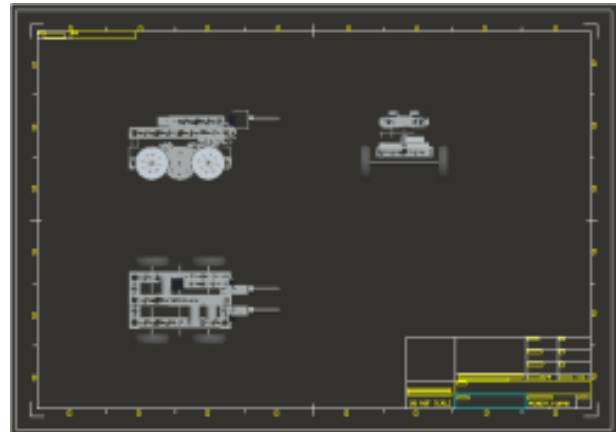


The views are too small so the scale of the drawing will be changed. Look in the bottom left corner of the drawing area and you will see the current scale is set to a scale such as **0.167**. This will be changed to 0.25

SCALE : 0,167    TYPE : ASSEM    NAME : ROBOT\_CLIMB    SIZE : A3

### Changing the scale



10. Double click on the scale text at the bottom of the drawing area.
11. A small dashboard will open in the command prompt line (usually across the top of the Pro|ENGINEER screen).
12. Alter the value to **0.25** then click on .

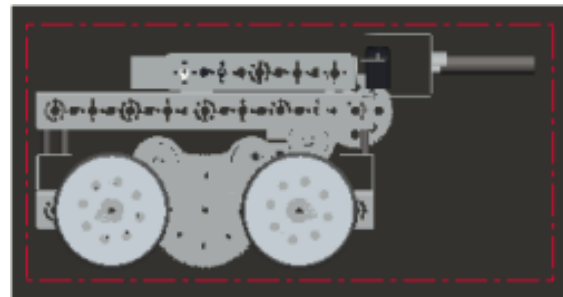


The views are now larger and the scale text shows the new value.


The three orthographic views are currently shaded and need to be changed to hidden line removal to comply with most international drawing standards.

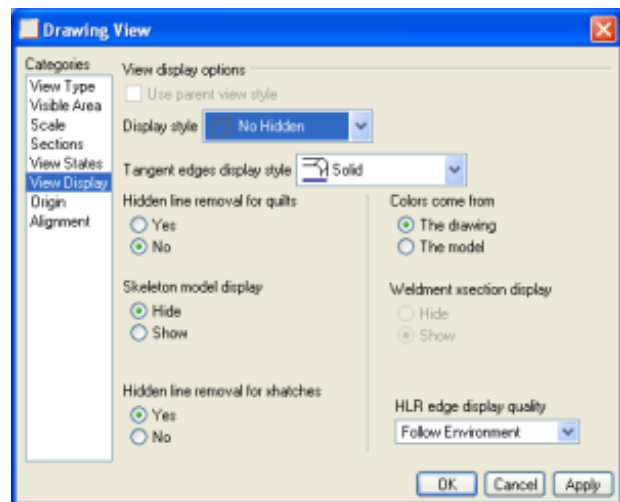
### Changing view properties

13. **Left Click**  to select one of the views. A red dashed outline will appear to show it is selected.
14. Keep the mouse cursor over the selected view then **hold** down **Right-click**  until a floating menu appears.
15. From the floating menu select **Properties**.



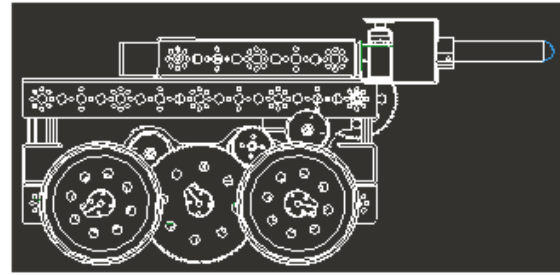
The **Drawing View** dialog opens.

16. In the Categories list select View Display.
17. Change the **Display Style** to **No Hidden** then click on  to close the dialog.





The selected view will now display without shading.

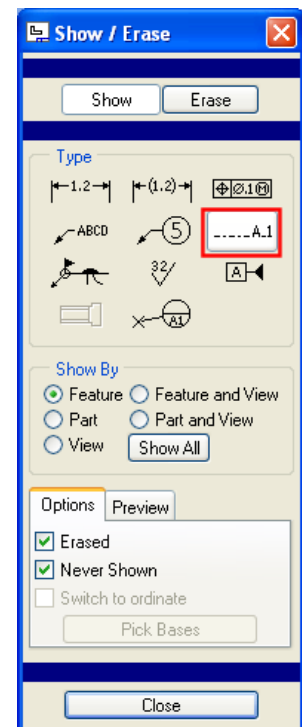
- Use the same method to change the properties of the other three views.



### Adding centre lines

With three views in place we can add dimensions and annotations to the drawing. The first step is to show the centre lines for the wheels.

- From the Drawing toolbar across the top of the Pro|ENGINEER screen, select **Show/Erase** . This will open the **Show/Erase** dialog box.
- In the **Type** section of the Show/Erase dialog left-click **Axis** .

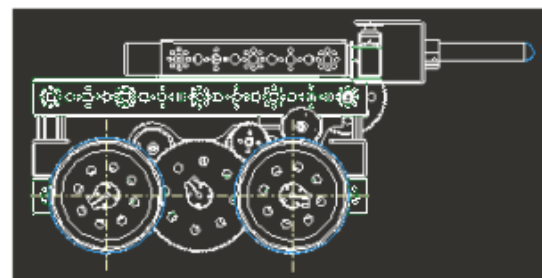


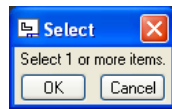
**Note:** Clicking on buttons in the **Type** section toggles them on/off.

- In the **Show By** section make sure **Feature** is selected.
- Zoom** in on one of the views.
- Move the mouse over one of the tires and, when the cylindrical surface making up the tire pre highlights, click to **select** the wheel.

Pro|ENGINEER will create centre-lines indicating the axes of the wheel revolve feature.

- Select** each wheel in turn until all wheels have centre lines.

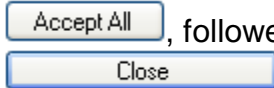




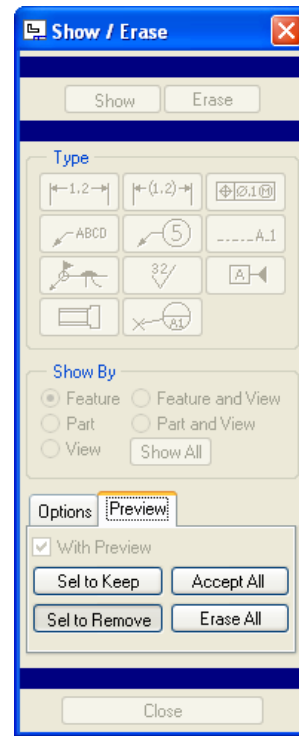
25. In the smaller **Select** dialog box **select** .

This informs **Pro|ENGINEER** you have finished your selections and will change the options in the **Show/Erase** dialog.

26. In the **Show/Erase** dialog select



**Pro|ENGINEER** will have created centre-lines in all 3 views.

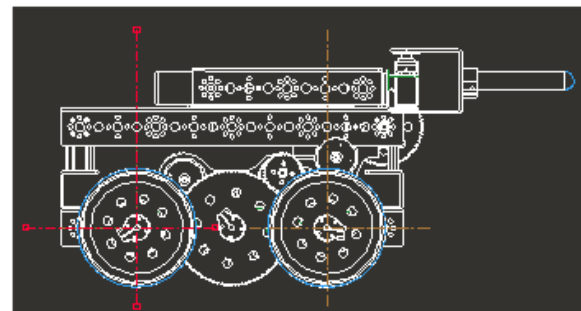


To improve the aesthetics of the newly created centre-lines the length of the centre-lines can be manually adjusted.

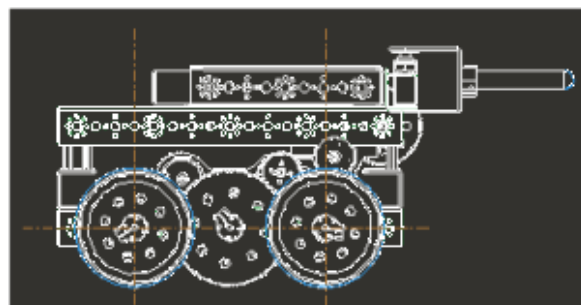
27. In the lower view select one of the newly created centre-lines.

The centre-lines will now have drag handles at each end.

28. Using these drag handles drag the centre-lines to the required lengths, and repeat this process for the other centre-lines in this and the other views.



Centre lines normally extend beyond the model or main feature






## Task two - Adding dimensions

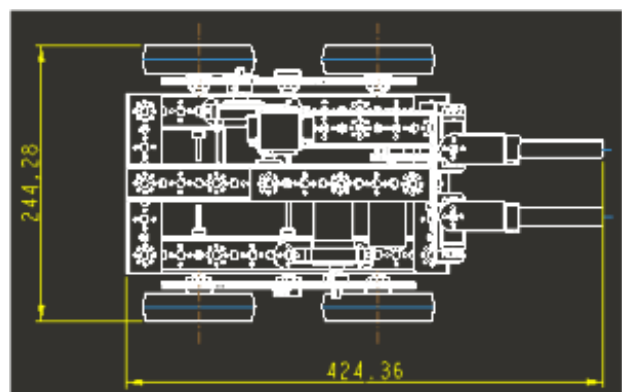
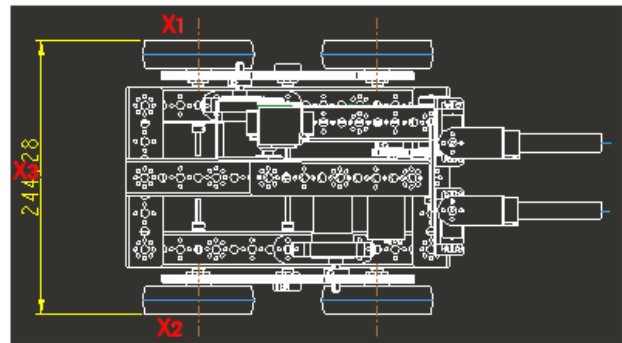
The next step is to create dimensions. It is important to know the purpose of the drawing which could be to manufacture a component, describe how to assemble the design or check dimensional accuracy for quality control. For each of these the dimensioning scheme would be different.

Pro|ENGINEER has the facility to import dimensions from the 3D model but for this exercise we will create dimensions individually.



First you will dimension the width of the front wing.

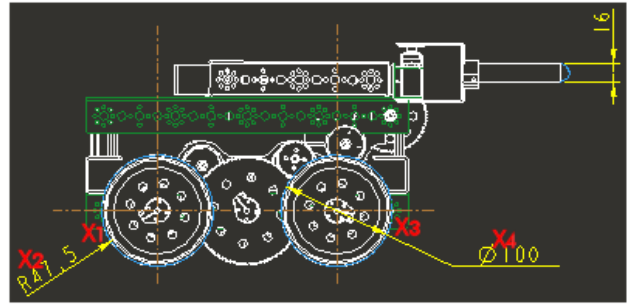
### Linear dimensions

1. In the Drawing toolbar across the top of the screen select , the **New references...** tool.
2. **Zoom** in on the plan view of the robot.
3. Move the mouse cursor over the outside edge of one wheel **X<sub>1</sub>**. When the edge pre-highlights in cyan, **Left Click**  to select and the edge will turn red.
4. Move the mouse cursor and click to select the outside edge **X<sub>2</sub>** of the opposite wheel to select it.
5. Move the mouse cursor into a clear space away from the end of the robot **X<sub>3</sub>** and **Middle mouse click**  to locate the dimension text.
6. The dimension will be created.
7. Add another dimension for the overall length of the robot.
8. Add any other linear dimensions you need to communicate to the intended audience.





## Circle dimensions

9. Zoom in on the front view of the robot.
10. Move the mouse over one of the wheels at  $X_1$  and, when the outer edge pre-highlights **light blue**, **Left Click**  to select it.
11. Move the mouse to where you want the dimension text to appear  $X_2$  and **Left click** .



A radius dimension will be created.

12. Move the mouse over the other wheel at  $X_3$  and, when the outer edge pre-highlights **light blue**, **Double left Click**  to select it.
13. Move the mouse to where you want the dimension text to appear  $X_4$  and Left click .

A diameter dimension will be created.

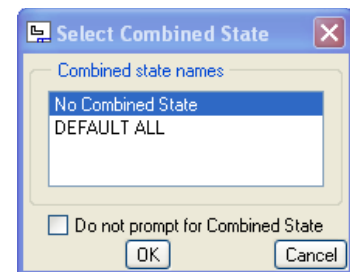
## Task three – Pictorial view

Additional views can be added at any time. In the steps that follow we will insert an Isometric shaded view to the drawing.

1. Open the **Insert** pull-down menu, select **Drawing View** then click on **General**.


The **Select Combined State** dialog opens.

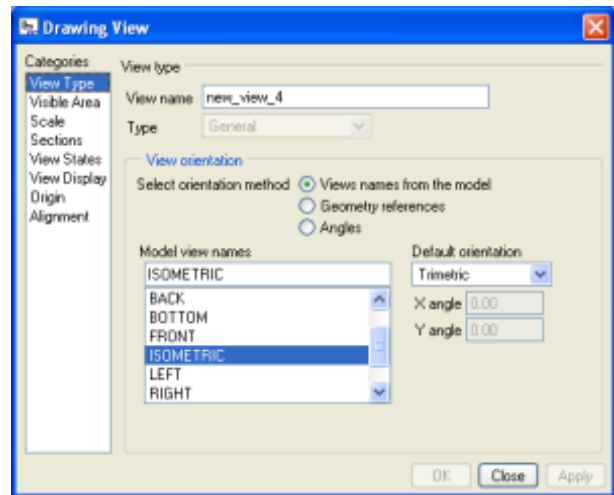
2. Make sure **No Combined State** is selected then click .



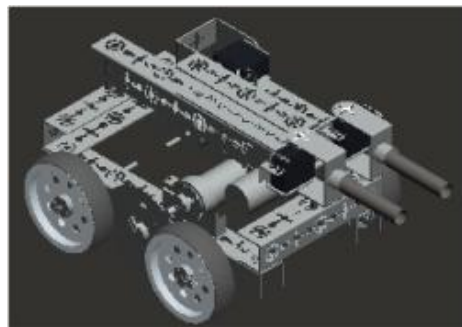
The prompt area at the top of the screen is prompting you to **Select CENTER POINT for drawing**

3. On the drawing, click where you want the view. For example in the blank area above the title block.

4. The **Drawing View** dialog opens and the view previews in the drawing.
5. In the **Categories** dialog **View Type** should be selected.
6. In the list **Model view names** scroll and select **ISOMETRIC**.
7. Click **Apply** to preview the changes in the drawing.
8. Click  close the dialog.



The Isometric view will appear in the drawing.



### Task four - Adding notes

The note tool in **Pro|ENGINEER** allows you to add text to the drawing. We will use it to fill-in the title block.

1. **Zoom** in on the title block in the bottom right hand corner of the drawing.

Notice some information has been entered for you automatically. This includes the drawing number from the filename and the drawing scale. **Pro|ENGINEER** can automate far more.

We will edit an existing, empty note and then add a note manually.

### Edit an existing note

2. Move the mouse inside the **TITLE** rectangle and when it pre-highlights click to select it.

The outline will turn red.

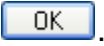


- Keep the mouse cursor inside the selected rectangle then click and hold the right mouse button.

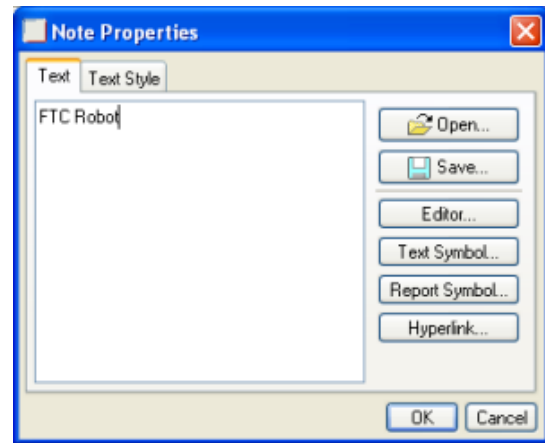
From the floating menu select **Properties**.

The **Note Properties** dialog opens.

- Under the **Text** tab, delete any text and type in **FTC Robot**.

- Click on .

The title block will now contain your text.



### Adding a note manually

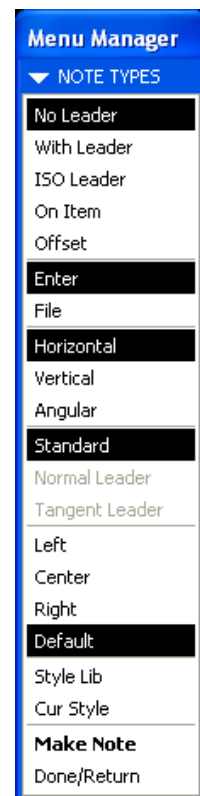
We will create two lines of text inside the large rectangle in the toolbar.

- Open the **Insert** pull-down text menu and select **Note...**


A **NOTE TYPES** menu manager dialog will open on the right of the screen.

Selections are made in each section working from top to bottom.

- We will use the defaults so at the bottom click on **Make Note**.





The prompt at the top of the screen is telling us to:

 Select LOCATION for note.

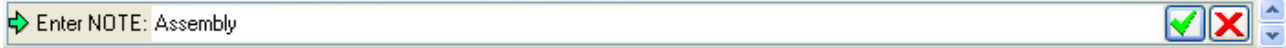
- Locate the large rectangle in the title block.
- Click near the top left corner of this rectangle  $X_1$  to establish the start point for the text.


The prompt area at the bottom of the screen is waiting for you to type text for the note.



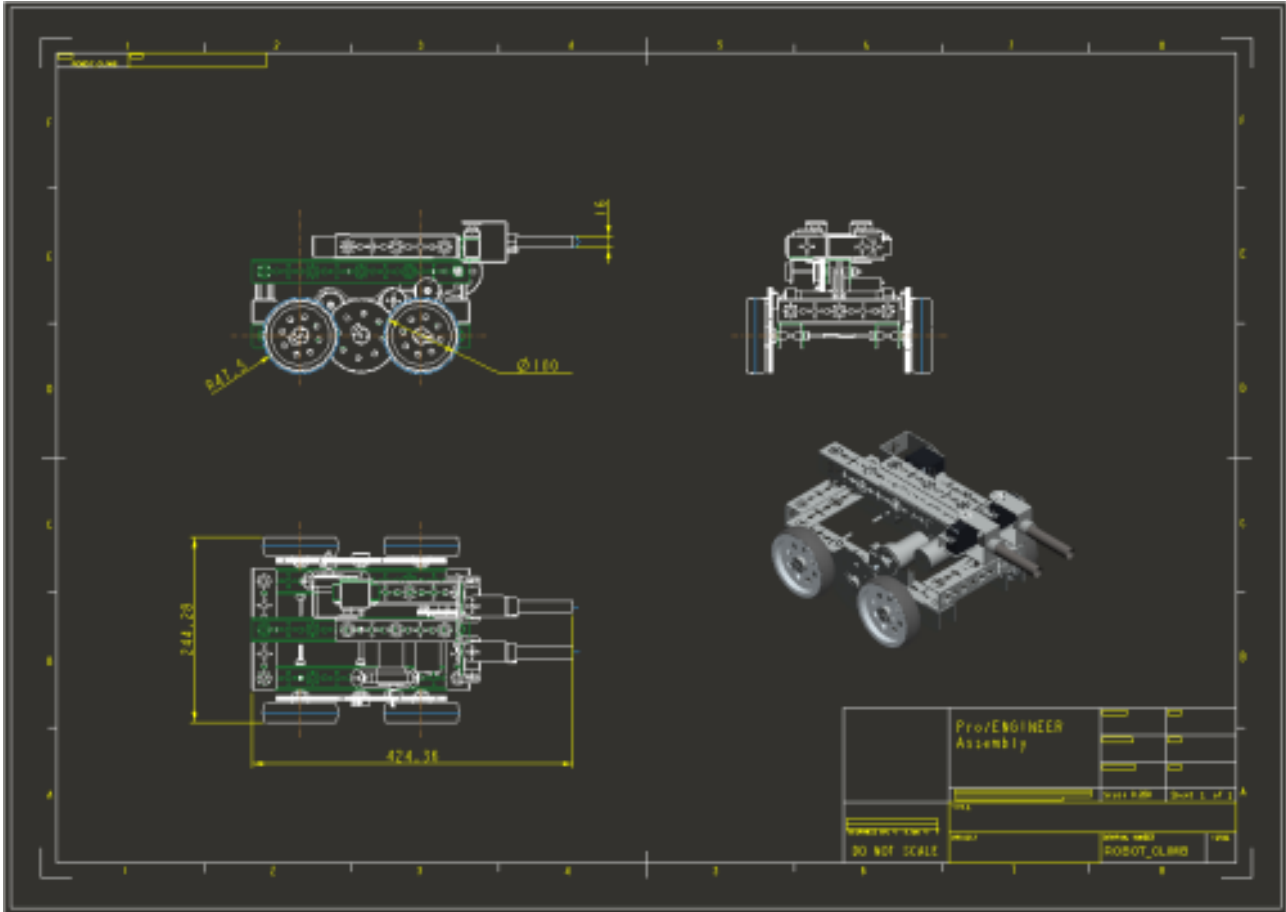
- Type** in the name of your team, we have used **Pro/ENGINEER**.
- Left Click**  **once** on the  check mark in the prompt line to close this line and open a second line for text.

12. In the prompt line, type the second line of text. We have used **Assembly**.



13. **Left Click**  **twice** on the  check mark to finish entering text.

14. The text will appear in the title block.



This briefing is now complete.

## Briefing nine – presentation images

If you would like to learn about rendering, PTC's schools curriculum contains projects covering rendering including:

- Periscope
- F1 in Schools D Type
- Scalextric4Schools

### Key principles

The **Advanced Rendering Extension** (ARX) module in **Pro|ENGINEER** is very powerful with many adjustments possible to the room environment such as materials, lighting, reflections, etc possible.

This session will provide a brief overview and hands-on experience of just a few features, enough to produce a final rendered image of the CO<sub>2</sub> car assembly.

The sequence you will work through is:

- Open an assembly of the FTC robot assembly
- Define initial render settings
- Load a scene definition
- Position the model in the room
- Change view of model
- Try an initial render
- Perform the final render
- Save the rendered image

### Task one - Getting started


1. Log-on and start **Pro|ENGINEER**
2. Set the working directory to: **\FTC\_Robot\Slope climb**
3. Open the model **ROBOT\_CLIMB.ASM**

The robot parts should already have an appearance similar to real materials.

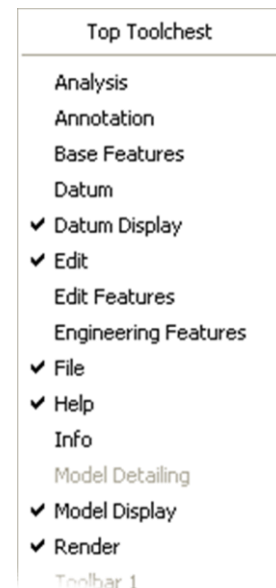
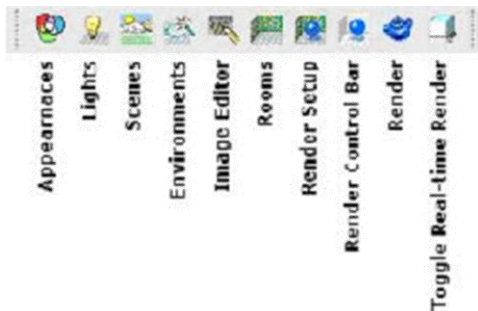
As you work through the rest of this section you will notice objects look even more like the material they would actually be made of.


## Task two - Initial render

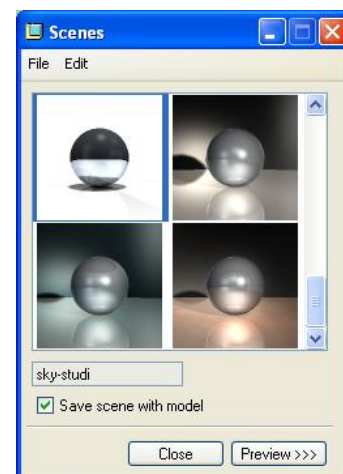
### Open the render toolbar

1. In the Main toolbar across the top of the screen, open the **View** pull-down menu and select **Realtime Rendering**.
2. In a blank area of the main toolbar, hold down the **Right mouse button**  and from the floating menu, select **Render**.

The render toolbar appears in the main toolbar across the top left of the screen.

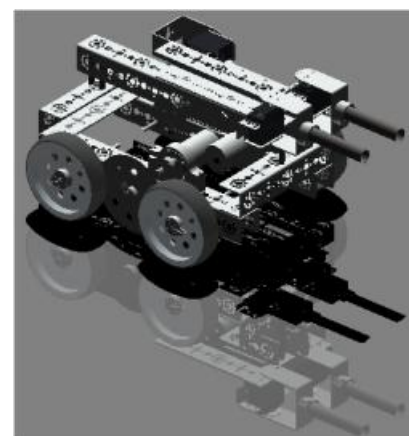


3. In the Render toolbar click on the **Scenes** button .
4. In the Scenes dialog scroll to select the **sky-studi** entry and double click to apply it to the model.
5. Check the option to **Save scene with model**.
6. Click  to close the dialog
7. A scene applies several parameters to the model in step including room size and shape, light settings, etc.






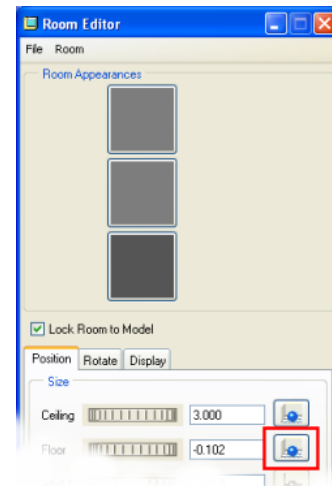
The graphics window will display a simple representation of the rendered view including shadows and reflections.

To see a higher quality view a number of options need to be changed.






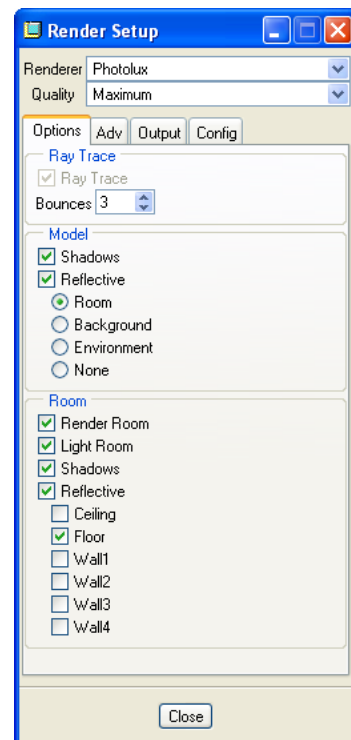
## Floor position

8. In the Render toolbar, select the **Room Editor** button .
9. In the Room Editor dialog, click  to **snap** the floor to the model.
10. **Click** on  to close the Room Editor dialog.




## Render settings

11. In the Render toolbar, click on  to open the floating **Render Control** toolbar.
12. In the Render Control toolbar, click on  to open the **Render Setup** dialog.
13. Makes sure the **Renderer** is set to **Photolux**.
14. Change the **Quality** option to **Maximum**.
15. Change the other options to those shown here.
16. Click on  to close the Render Setup dialog.

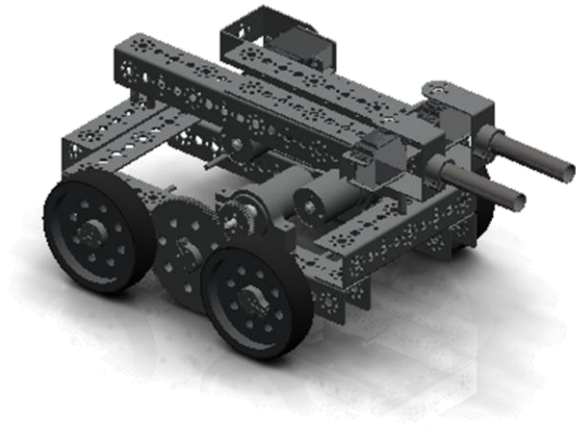


### Task three – Final render

1. Select the **Render** icon  from the Rendering Toolbar.

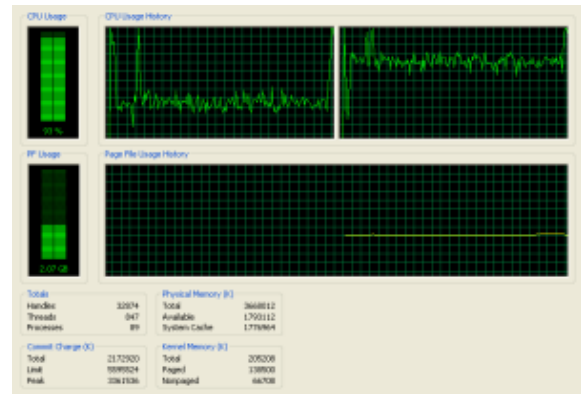
You will see the Render Abort dialog displaying progress. After a few moments, or perhaps a few minutes on slower computers, you will see the rendered image of your robot.

Congratulations, you have just created your first rendering!



On a fairly quick Dell PC with Dual Core processor and 2GB RAM, this draft render took around 3 minutes and used over 50% of the two processors resources for that time.

This shows the level of math computation needed to calculate the light paths, surface reflections and effects.




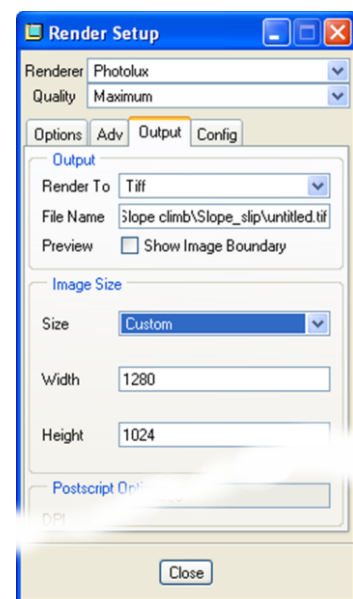
Experiment with the position and render each time until you are happy with the final image. When you are happy with all the settings, change the quality to **Maximum** and render the model. You will probably have time for a leisurely drink...


### Save rendered image

If you want a quick copy of the rendered mode you could capture and crop the screen display.

Pro|ENGINEER provides the option to create rendered images in several different formats and at higher resolutions than the screen.

2. Click  in the Render toolbar or click **View > Model Setup > Render Setup** to open the **Render Setup** dialog.
3. Click the **Output** tab.
4. Change the **Render To** option to **Tiff** or an image format of your choice.
5. A file name with the appropriate extension appears in the **File Name** box. Edit this giving the file a name you will remember.



6. Change the image **Size** to **Custom**.
7. Alter the **Width & Height** until the border surrounds your model.
8. Hold down **shift** and drag with the **MMB** to position the model in the frame.
9. **Close** the Render Setup dialog.
10. In the Render toolbar click on  to render the model.
11. The image will be saved with the required file name.

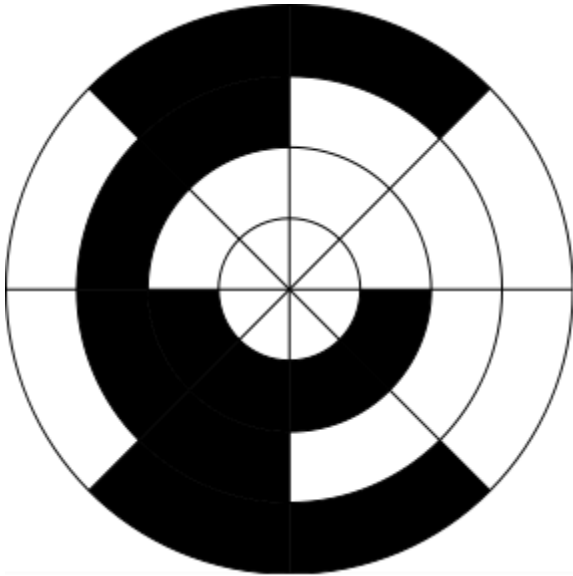
A Tiff image file has been created in the working directory of the rendered model

12. Save your model, exit **Pro|ENGINEER**.

This briefing is now complete.

## Extension one – Position encoding

Knowing the rotational position of an axel is very important in robotics and sensors are available that will do this. One of the principles used is a version of the binary counting system called 'Gray' scale where the binary 'bits' only change one at a time.



For more information on rotary encoding go to: [http://en.wikipedia.org/wiki/Rotary\\_encoder](http://en.wikipedia.org/wiki/Rotary_encoder)



## Extension two – Structural analysis

Schools Edition/Schools Advanced Edition - Loads, stress-strain, Young's modulus, deflection, yield, failure

## Appendix one - Web links

[Carnegie Mellon University - Robotics Academy](#)

[Self Assembly robots](#)

[Caterpillar – robotic mine trucks](#)

[Prosthetic arm – Dean Kamen](#)

[Robotic arm](#)

[Mechatronics](#)

[Mars rover – BBC article](#)

[Exoskeleton](#)

[FIRST team – Trobotics - 1346](#)

[Lunar habitat hauler](#)

[Pro|ENGINEER designed Twendy robot](#)

[Reliable mars rover – Futures channel](#)

[Carnegie Mellon Robotics Institute](#)

[Nuvo robot](#)

[Twendy-one](#)

[Technogames](#)

D:\PTCData\AA ProE\AA Curriculum\02 High School\02 09 Robotics\FTC\FTC\_Robotics.doc

## Appendix two – Servo motor specification



### Detailed Specifications

**Control System:** +Pulse Width Control 1500usec Neutral

**Required Pulse:** 3-5 Volt Peak to Peak Square Wave

**Operating Voltage:** 4.8-6.0 Volts

**Operating Temperature Range:** -20 to +60 Degree C

**Operating Speed (4.8V):** 0.23sec/60° at no load

**Operating Speed (6.0V):** 0.18sec/60° at no load

**Stall Torque (4.8V):** 61 oz/in. (4.3kg.cm)

**Stall Torque (6.0V):** 76 oz/in. (5.5kg.cm)

**Operating Angle:** 45 Deg. one side pulse traveling 400usec

**360 Modifiable:** Yes

**Direction:** Clockwise/Pulse Traveling 1500 to 1900usec

**Current Drain (4.8V):** 8mA/idle and 150mA no load operating

**Current Drain (6.0V):** 8.8mA/idle and 180mA no load operating

**Dead Band Width:** 8usec

**Motor Type:** 3 Pole Ferrite Motor

**Potentiometer Drive:** Indirect Drive

**Bearing Type:** Top Ball Bearing, Lower Bushing

**Gear Type:** Karbonite Gears

**Connector Wire Length:** 11.81" (300mm)

**Dimensions:** See Schematic

**Weight:** 1.4oz (39g)

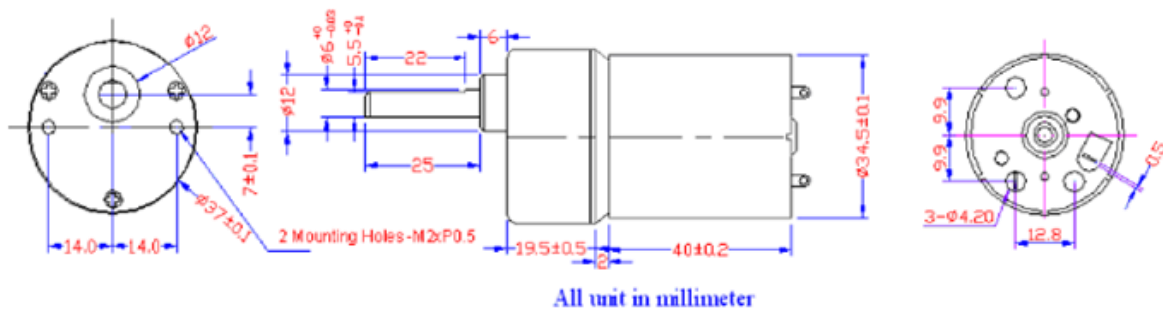
# Appendix three – DC geared motor specification

## TETRIX DRIVE MOTOR Dynamic Test Report

Product: DC Gear Motor  
 Customer: Pitsco  
 Mode NO.: HG372F280-052F  
 Test Date: 2008/05/12

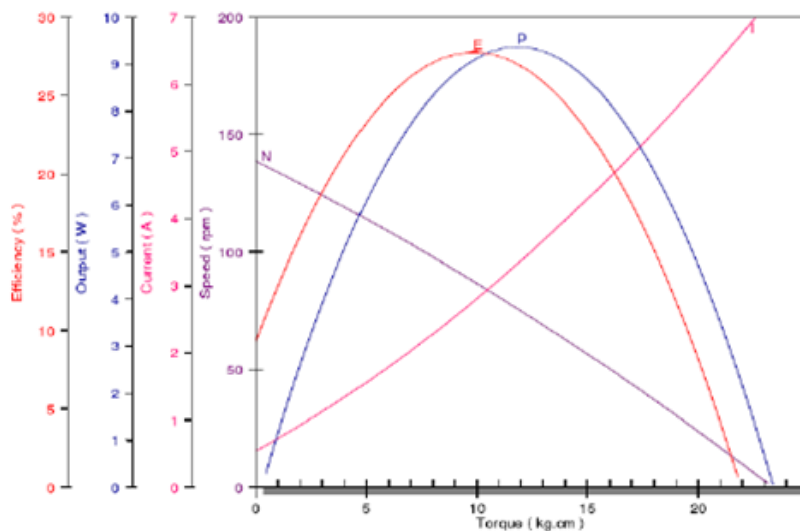
Rated Voltage: 12 VDC  
 No Load Speed: 154 RPM  
 No Load Current: 0.428 A  
 Gear Ratio: 1: 52

(A) Dimensional Figures:



(B) MOTOR PERFORMANCE CURVES AND CHARACTERISTICS:

Supply Voltage = 12 VDC



AT NO LOAD

SPEED = 138.7 RPM  
 CURRENT = 0.54 AMP

AT STALL EXTRAPOLATION

TORQUE = 23.40 kg-cm  
 CURRENT = 7.30 AMP

AT MAXIMUM EFFICIENCY

EFFICIENCY = 27.74 %  
 SPEED = 88.0 RPM  
 TORQUE = 9.77 kg-cm  
 CURRENT = 2.75 AMP  
 OUTPUT = 9.06 WATTS

AT MAXIMUM OUTPUT

SPEED = 76.1 RPM  
 TORQUE = 11.81 kg-cm  
 CURRENT = 3.32 AMP  
 OUTPUT = 9.36 WATTS  
 EFFICIENCY = 27.01 %

Torque Conversions:

1.0 kgf-cm = 0.098 Nm = 98 mNm  
 = 13.887 oz-in = 0.867 lb-in

PREPARED BY: \_\_\_\_\_

APPROVED BY: \_\_\_\_\_