Mechanism Design

with Creo Elements/Pro 5.0

(Pro/ENGINEER Wildfire 5.0)
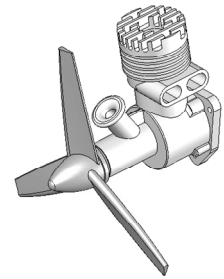

Kuang-Hua Chang, Ph.D.
School of Aerospace and Mechanical Engineering
The University of Oklahoma
Norman, OK

Lesson 1: Introduction to
Mechanism Design

1.1 Overview of the Lesson

The purpose of this lesson is to provide you with a brief overview of *Mechanism Design*. *Mechanism Design* is a virtual prototyping tool that supports mechanism analysis and design. Instead of building and testing physical prototypes of the mechanism, you may use *Mechanism Design* to evaluate and refine the mechanism before finalizing the design and entering the functional prototyping stage. *Mechanism Design* will help you analyze and eventually design better engineering products. More specifically, the software enables you to size motors and actuators, determine power consumption, layout linkages, develop cams, understand gear trains, size springs and dampers, and determine interference between parts, which would usually require tests of physical prototypes. With such information, you will gain insight on how the mechanism works and why it behaves in certain ways. You will be able to modify the design and often achieve better design alternatives using the more convenient and less expensive virtual prototypes. In the long run, using virtual prototyping tools, such as *Mechanism Design*, will help you become a more experienced and competent design engineer.

In this lesson, we will start with a brief introduction to *Mechanism Design* and the various types of physical problems that *Mechanism Design* is capable of solving. We will then discuss capabilities offered by *Mechanism Design* for creating motion models, conducting motion analyses, and viewing motion analysis results. In the final section, we will mention examples employed in this book and topics to learn from these examples.

Note that materials presented in this lesson will be kept brief. More details on various aspects of mechanism design and analysis using *Mechanism Design* will be given in later lessons.

1.2 What is *Mechanism Design*?

Mechanism Design is a computer software tool that supports engineers in analyzing and designing mechanisms. *Mechanism Design* is a module of the *Pro/ENGINEER* product family developed by *Parametric Technology Corporation*. This software supports users in creating virtual mechanisms that answer general questions in product design such as those described next. An internal combustion engine shown in Figures 1-1 and 1-2 will be used to illustrate some typical questions.

1. Will the components of the mechanism collide in operation? For example, will the connecting rod collide with the inner surface of the piston or the inner surface of the engine case during operation?

2. Will the components in the mechanism you design move according to your intent? For example, will the piston stay entirely in the piston sleeve? Will the system lock up when the firing force aligns vertically with the connecting rod?

3.   How much torque or force does it take to drive the mechanism? For example, what will be the minimum firing load to move the piston? Note that in this case, proper friction forces must be added to simulate the resistance of the mechanism before a realistic firing force can be calculated.

4.   How fast will the components move; e.g., the longitudinal motion of the piston?

5.   What is the reaction force or torque generated at a connection (also called *joint* or *constraint*) between components (or bodies) during motion? For example, what is the reaction force at the joint between the connecting rod and the piston pin? This reaction force is critical since the structural integrity of the piston pin and the connecting rod must be ensured; i.e., they must be strong and durable enough to sustain the load in operation.

The modeling and analysis capabilities in *Mechanism Design* will help you answer these common questions accurately and realistically, as long as the motion model is properly defined.

Piston

Piston pin

Engine case

Connecting rod

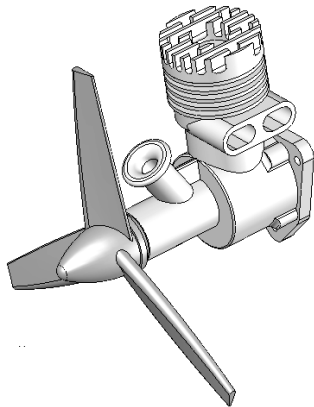Crankshaft

Propeller

Figure 1-1 An Internal Combustion
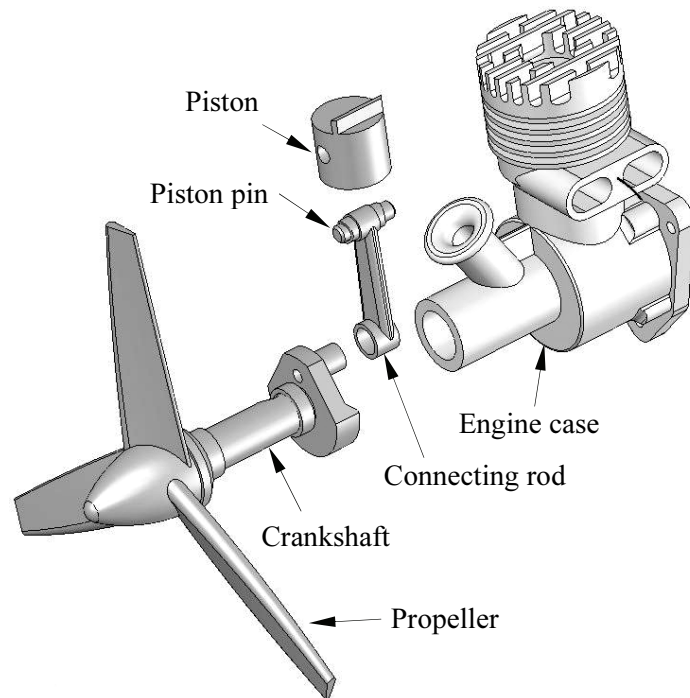   Engine (Unexploded View)

Figure 1-2 Internal Combustion Engine (Exploded View)

The capabilities available in *Mechanism Design* also help you search for better design alternatives. A better design alternative is very much problem-dependent. It is critical that a design problem be clearly defined by the designer up front before searching for better design alternatives. For the engine example, a better design alternative can be a design that reveals:

1.   A smaller reaction force applied to the connecting rod, and
2.   No collisions or interference between components.

In order to vary component sizes for exploring better design alternatives, the parts and assembly must be adequately parameterized to capture design intents. At the parts level, design parameterization implies creating solid features and relating dimensions properly. At the assembly level, design parameterization involves defining assembly mates and relating dimensions across parts. When a solid model is fully parameterized, a change in dimension value can be propagated to all parts affected automatically. Parts affected must be rebuilt successfully, and at the same time, they will have to maintain proper position and orientation with respect to one another without violating any assembly mates or revealing part penetration or excessive gaps. For example, in this engine example, a change in the bore

diameter of the engine case will alter not only the geometry of the case itself, but all other parts affected, such as the piston, piston sleeve, and even the crankshaft, as illustrated in Figure 1-3. Moreover, they all have to be rebuilt properly and the entire assembly must stay intact through assembly mates.
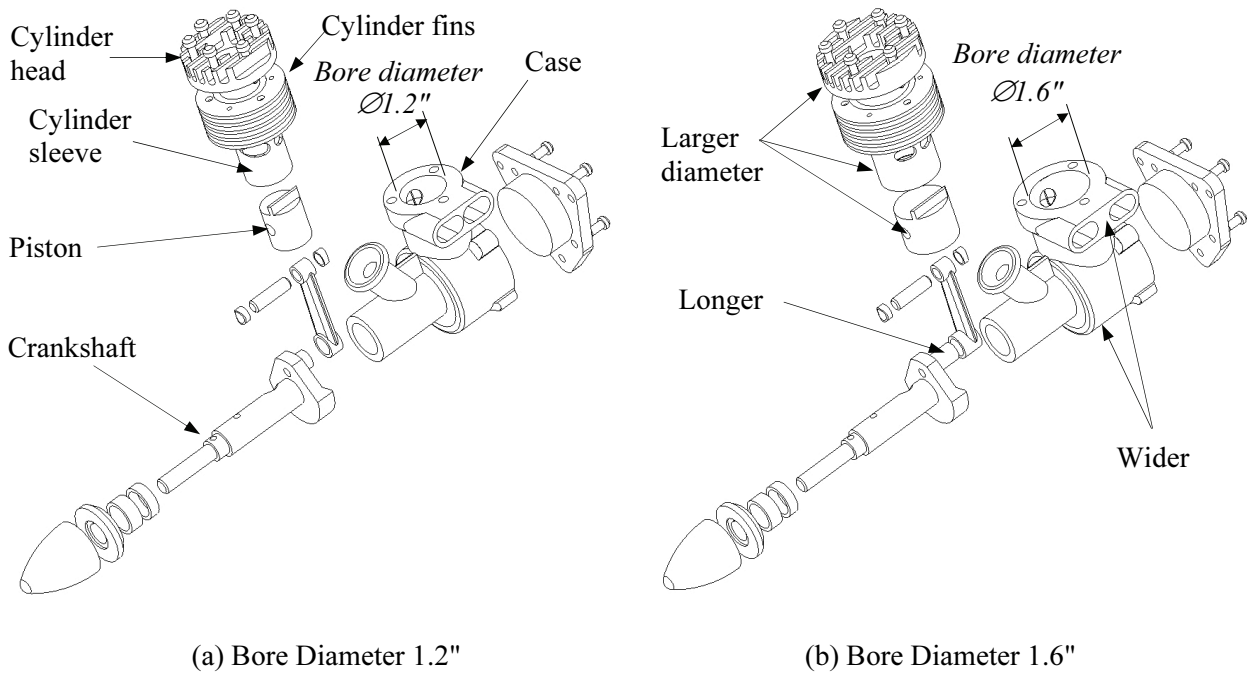

(a) Bore Diameter 1.2"   (b) Bore Diameter 1.6"

Figure 1-3 Internal Combustion Engine—Exploded View

1.3   Mechanism and Motion Analysis

A mechanism is a mechanical device that transfers motion and/or force from a source to an output. It can be an abstraction (simplified model) of a mechanical system. A linkage consists of links (or bodies), which are connected by connections (or joints), such as a pin joint, to form open or closed chains (or loops, see Figure 1-4). Such kinematic chains, with at least one link fixed, become mechanisms. In this book, all links are assumed rigid. In general, a mechanism can be represented by its corresponding schematic drawing for analysis and design purposes. For example, a slider-crank mechanism represents the engine motion, as shown in Figure 1-5, which is a closed loop mechanism.

In general, there are two types of motion problems that you will solve in order to answer general questions regarding mechanism analysis and design: kinematics and dynamics.

Kinematics is the study of motion without regard for the forces that cause the motion. A kinematic mechanism must be driven by a servo motor (or driver) so that the position, velocity, and acceleration of each link of the mechanism can be analyzed at any given time. Typically, a kinematic analysis must be conducted before dynamic behavior of the mechanism can be simulated properly.

Dynamics is the study of motion in response to externally applied loads. The dynamic behavior of a mechanism is governed by Newton's laws of motion. The simplest dynamic problem is the particle dynamics covered in Sophomore Dynamics – for example, a spring-mass-damper system shown in Figure 1-6. In this case, motion of the mass is governed by the following equation derived from Newton's second law,
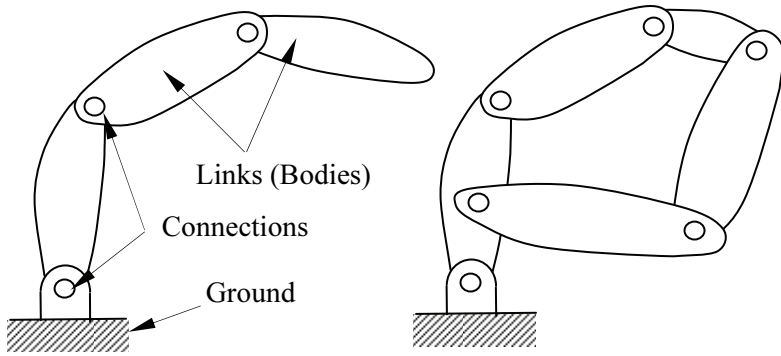
(a) Open Loop Mechanism   (b) Closed Loop Mechanism
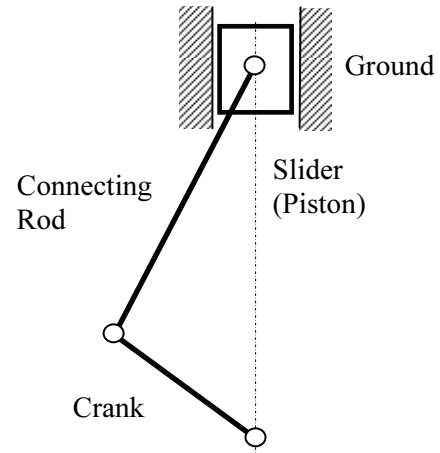
Figure 1-4 General Mechanisms

Figure 1-5 Schematic View of the Engine Motion Model

$$\sum F = p(t) - kx - c\,\dot{x} = m\,\ddot{x} \tag{1.1}$$

where ($\cdot$) appearing on top of the physical quantity represents time derivative of the quantity, m is the total mass of the block, k is the spring constant, and c is the damping coefficient.

For a rigid body, mass properties (such as the total mass, center of mass, moment of inertia, etc.) are taken into account for dynamic analysis. For example, motion of a pendulum shown in Figure 1-7 is governed by the following equation of motion,

$$\sum M = -mg\ell \sin\theta = J\ddot{\theta} = m\ell^2\ddot{\theta} \tag{1.2}$$

where M is the external moment (or torque), J is the polar moment of inertia of the pendulum, m is the pendulum mass, g is the gravitational acceleration, and $\ddot{\theta}$ is the angular acceleration of the pendulum.
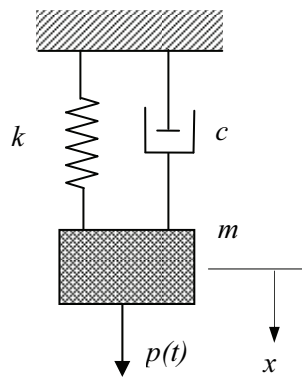

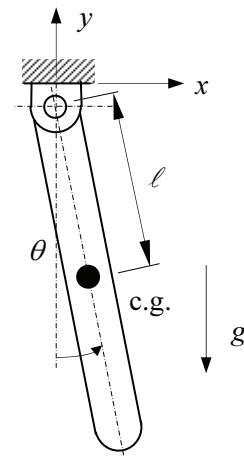Figure 1-6 The Spring-Mass-Damper System

Figure 1-7 A Simple Pendulum

Dynamics of a rigid body system, such as those illustrated in Figure 1-4, is a lot more complicated than the single body problems. Usually, a system of differential and algebraic equations governs the motion and the dynamic behavior of the system. Newton's law must be obeyed by every single body in the system at all times. The motion of the system will be determined by the loads acting on the bodies or joint axes (e.g., a torque driving the system). Reaction loads at the joint connections hold the bodies together.

Note that in *Mechanism Design*, you may create a kinematic analysis model; for example, using a servo motor to drive the mechanism before carrying out a dynamic analysis. In this case, position, velocity, and acceleration results may be similar to those of kinematic analysis; however, the inertia of the

bodies will be taken into account for dynamic analysis; therefore, reaction forces will be calculated between bodies.

1.4 *Mechanism Design* Capabilities

Overall Process

The overall process of using *Mechanism Design* for analyzing a mechanism consists of three main steps: model creation, analysis, and result visualization, as illustrated in Figure 1-8. Key entities that constitute a motion model include a ground body that is always fixed, bodies that are movable, connections (or joints) that connect bodies, servo motors (drivers) that drive the mechanism for kinematic analysis, loads, and the initial conditions. More details about these entities will be discussed later in this lesson.

The analysis capabilities in *Mechanism Design* include position (initial assembly), static (equilibrium configuration), motion (kinematic and dynamic), and force balance (to retain the system in a prescribed configuration).

```
┌────────────────┐        ─── Ground Body
│ Motion Model   │       /─── Bodies
│  Generation    │──────<──── Connections
└────────────────┘       \─── Drivers
        │                 \── Loads
        │                  ── Initial Conditions
        ▼
┌────────────────┐        ─── Position
│    Motion      │       /─── Static
│   Analysis     │──────<──── Motion (Kinematics
└────────────────┘       \     and Dynamics)
        │                 \── Force Balance
        │
        ▼
┌────────────────┐        ─── Animation
│    Results     │       /─── Graph
│ Visualization  │──────<──── Query
└────────────────┘        ── Report
```

Figure 1-8 General Process of Using
Mechanism Design

For example, the position analysis brings bodies closer within a prescribed tolerance at each connection to create an initial assembled configuration of the mechanism. More details about the analysis capabilities in *Mechanism Design* will be discussed later in this lesson.

The analysis results can be visualized in various forms. You may animate motion of the mechanism, or generate graphs for more specific information, such as the reaction force of a joint in the time domain. You may also query results at specific locations for a given time. Furthermore, you may ask for a report on results that you specified, such as the acceleration of a moving body in the time domain.

Operation Mode

Mechanism Design is embedded into *Pro/ENGINEER*. It is indeed an integrated module of *Pro/ENGINEER*, and transition from *Pro/ENGINEER* to *Mechanism Design* is seamless. All the solid models, placement constraints, etc. defined in *Pro/ENGINEER* are automatically carried over into *Mechanism Design*. *Mechanism Design* can be accessed through menus and windows inside *Pro/ENGINEER*. The same assembly is used in both *Pro/ENGINEER* and *Mechanism Design*.

Body geometry is essential for mass property computations in motion analysis. In *Mechanism Design,* all mass properties are ready for use. In addition, the detailed part geometry for interference checking is also available.

User Interface

User interface of the *Mechanism Design* is identical to that of *Pro/ENGINEER*, as shown in Figure 1-9. *Pro/ENGINEER* users should find it is straightforward to maneuver in *Mechanism Design*.
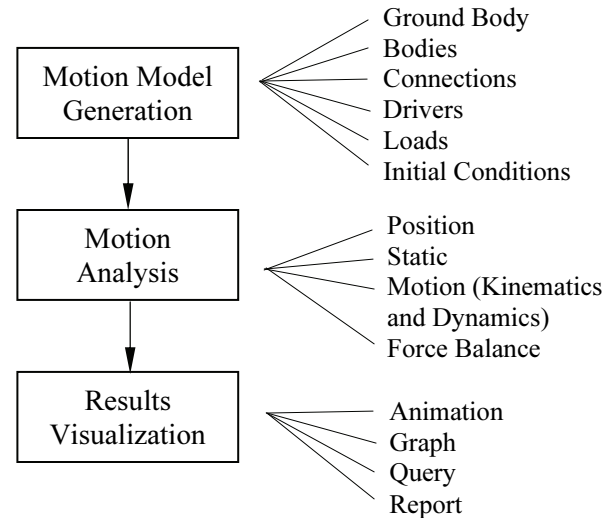
Title Bar   Pull-Down Menus   *Quit* Button

Create New Model

Shortcut Buttons

Prompt/Message Window   *Graphics* Window

Model Tree: Parts and Assemblies

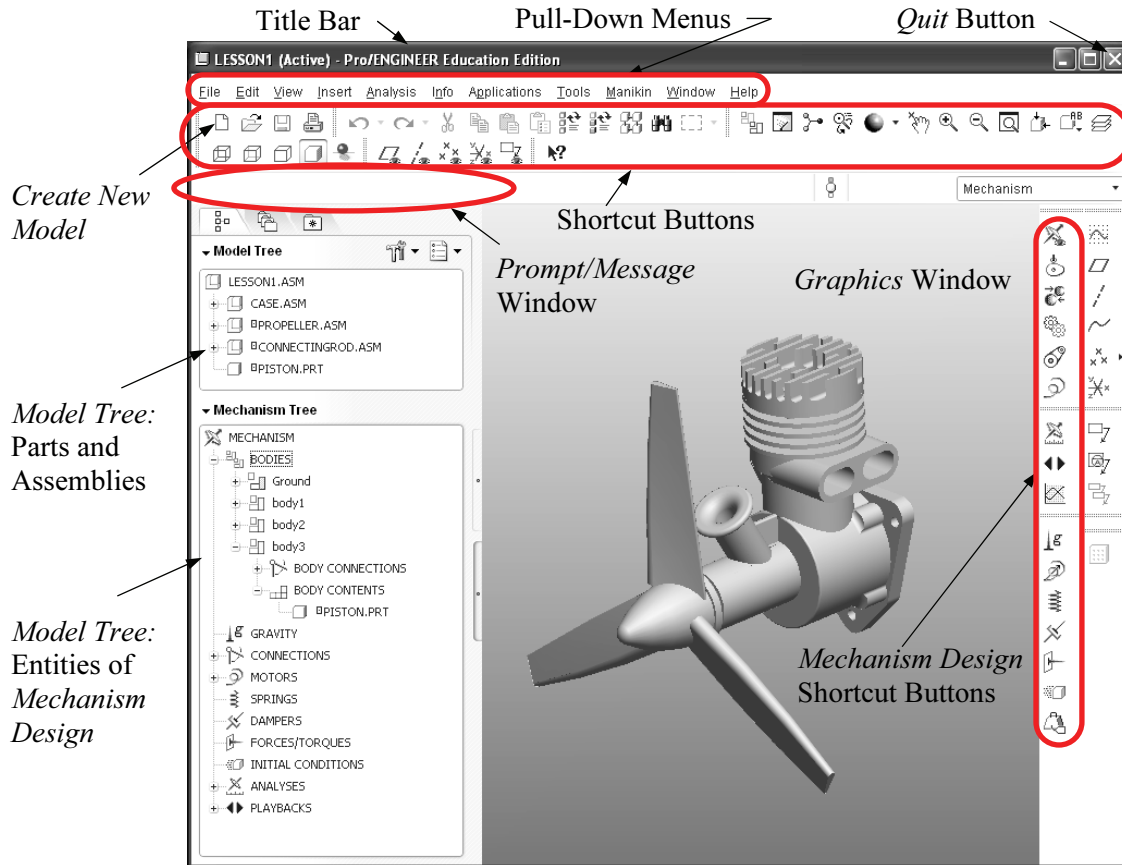*Model Tree:* Entities of *Mechanism Design*

Mechanism Design Shortcut Buttons

Figure 1-9 User Interface of *Mechanism Design*

As shown in Figure 1-9, the user interface window of *Mechanism Design* consists of pull-down menus, shortcut buttons, prompt/message window, scroll-down menu, graphics window, and model tree window.,

The *Graphics* window displays the motion model with which you are working. The pull-down menus and the shortcut buttons at the top of the screen provide typical *Pro/ENGINEER* functions. The *Mechanism Design* shortcut buttons to the right provide all the functions required to create and modify motion models, create and run analyses, and visualize results. When you click the menu options, the *Prompt/Message* window shows brief messages describing the menu commands. It also shows system messages following command execution. The shortcut buttons in *Mechanism Design* and their functions are summarized in Table 1-1.

Defining Motion (or Mechanism Design) Entities

The basic entities of a motion (or simulation) model created in *Mechanical Design* consist of ground, bodies, connections, initial conditions, drivers, and loads. Each of the basic entities will be briefly introduced. More details can be found in later lessons.

Table 1-1 The Shortcut Buttons in *Mechanism Design*

Button Symbol	Name	Function
	Display Entities	Turn icon visibility on or off in your assembly.
	Cam-Follower Connection Definition	Create a new cam-follower.
	Gear Pairs	Create a new gear pair.
	Servo Motors	Define a servo motor (driver).
	Analysis Definition	Define and run an analysis.
	Playbacks	Play back the results of your analysis run. You can also save or export the results or restore previously saved results.
	Measure Results	Create measures, and select measures and result sets to display. You can also graph the results or save them to a table.
	Gravity	Define gravity.
	Force Motors	Define a new force motor.
	Springs	Define a new spring.
	Dampers	Define a new damper.
	Forces/Torques	Define a force or a torque.
	Initial Condition Definition	Specify initial position snapshots, and define the velocity initial conditions for a point, motion axis or body.
	Mass Properties	Specify mass properties for a part or specify density for an assembly.

Ground Body

A ground (or ground body) represents a fixed location in space. The root assembly is always fixed; therefore, becoming the ground body (or part of the ground body). Also, the datum coordinate system of the root assembly is assigned as the *WCS* (World Coordinate System) by default. All datum features and parts fixed to the root assembly are part of the ground body.

Bodies

A body represents a single rigid component (or link) that moves relative to the other body (or bodies in some cases). A body may consist of several *Pro/ENGINEER* parts fully constrained using placement constraints. A body must contain a local coordinate system (*LCS*), body points (created as datum points), and mass properties. Note that body points are created for defining connections, force applications, etc.

A spatial body consists of three translational and three rotational degrees of freedom (dof's). That is, a rigid body can translate and rotate along the *X*-, *Y*-, and *Z*-axes of a coordinate system. Rotation of a rigid body is measured by referring the orientation of its *LCS* to *WCS*, which is fixed to the ground body.

In *Mechanism Design*, the *LCS* is assigned automatically, usually, to the default datum coordinate system of the body (either part or assembly), and the mass properties are calculated using part geometry and material properties referring to the *LCS*. Datum axes and points are essential in creating the motion model since they are employed for defining connections and the location of external load application.

Connections

A connection in *Mechanism Design* can be a joint, cam, or gear that connects two bodies. Typical joints include a pin, slider, bearing, cylinder, etc. The connection will constrain the relative motion between bodies. Each independent movement permitted by a connection is called a degree of freedom (dof). The degrees of freedom that a connection allows can be translation and rotation along three perpendicular axes, as shown in Figure 1-10.
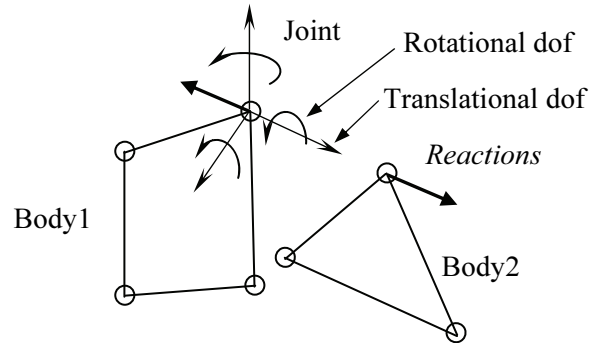
Figure 1-10 A Typical Joint in *Mechanism Design*

Note that joints are created by employing regular assembly placement constraints, such as align, mate, etc. However, instead of completely fixing all the movements, certain dof's (translations and/or rotations) are left to allow designated movement. For example, a ball joint is created simply by aligning two datum points in their respective bodies, allowing all three rotational dof's.

The connections produce equal and opposite reactions (forces and/or torques) on the bodies connected. The symbol of a given joint tells the translational and/or rotational dof that the joint allows in regard to movement. Understanding the basic four symbols shown in Figure 1-11 will enable you to read any existing joints in motion models. More details about joint types available in *Mechanism Design* will be discussed in later lessons. A complete list of joints available in *Mechanism Design* can be found in Appendix A.
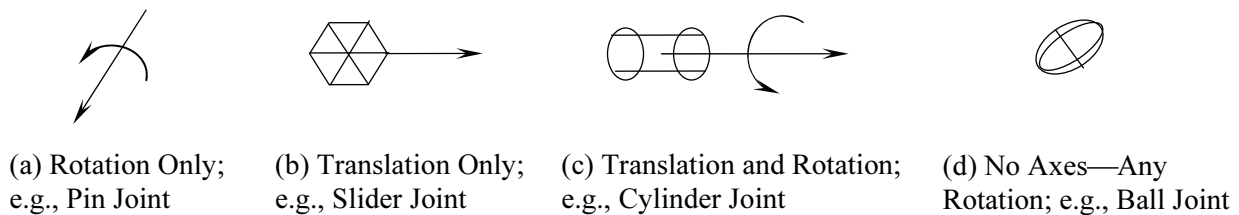
(a) Rotation Only;   (b) Translation Only;   (c) Translation and Rotation;   (d) No Axes—Any
e.g., Pin Joint   e.g., Slider Joint   e.g., Cylinder Joint   Rotation; e.g., Ball Joint

Figure 1-11 Basic Joint Symbols

Degrees of Freedom

As mentioned earlier, an unconstrained body in space has six degrees of freedom, three translational and three rotational. When joints are added to connect bodies, constraints are imposed to restrict the relative motion between them

For example, a pin joint allows one rotational motion between bodies. As defined in the engine example shown in Figure 1-12, joint *Pin1* restricts movement on five dof's so that only one rotational motion is allowed between the propeller assembly and the ground body (*case.asm*). Since the engine case is a ground body, the propeller assembly will rotate along the axis of the pin joint, as illustrated in the symbol shown in Figure 1-12. Therefore, there is only one degree of freedom left for the propeller

assembly. For a given motion model, you can determine its number of degrees of freedom using the Gruebler's count.
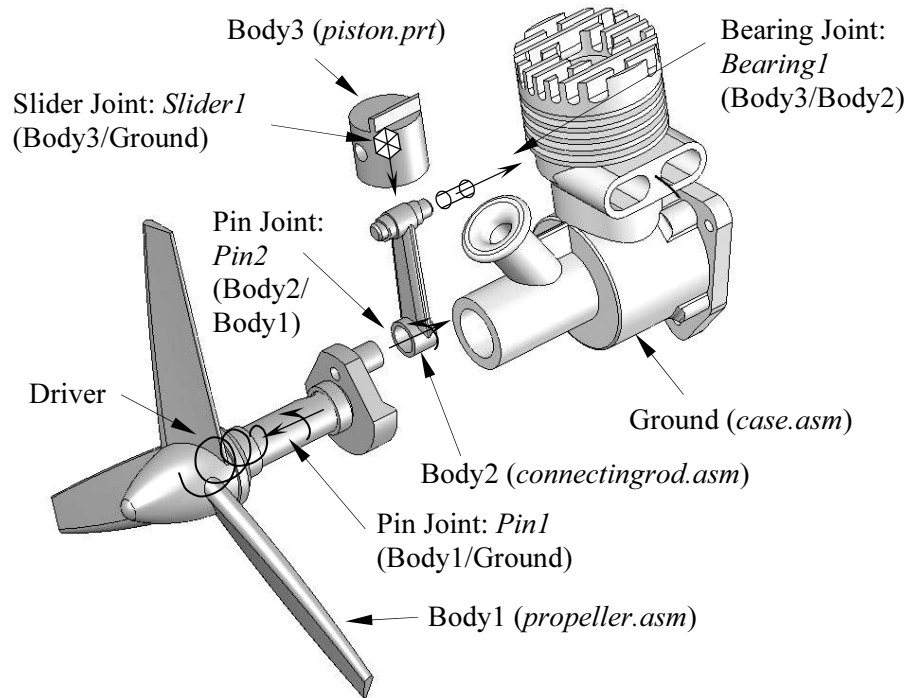
Figure 1-12 A Typical Motion Model in Exploded View

Mechanical Design uses the following equation to calculate the Gruebler's count:

$$D = 6M - N - O \qquad\qquad (1.3)$$

where *D* is the Gruebler's count representing the overall free degrees of freedom of the mechanism, *M* is the number of bodies excluding the ground body, *N* is the number of dof's restricted by all joints, and *O* is the number of motion drivers defined in the system.

In general, a valid motion model should have a Gruebler's count *0*. However, in creating motion models, some joints remove redundant dof's. For example, two hinges, modeled using two pin joints, support a door. The second pin joint adds five redundant dof's. The Gruebler's count becomes:

$$D = 6 \times 1 - 2 \times 5 = -4$$

For kinematic analysis, the Gruebler's count must be equal to or less than *0*. The solver recognizes and deactivates redundant constraints during analysis. For a kinematic analysis, if you create a model and try to animate it with a Gruebler's count greater than *0*, the animation will not run and an error message will appear.

The single-piston engine shown in Figure 1-12 consists of three bodies (excluding the ground body), two pin joints, one slider joint, and one bearing joint. A pin or slider joint removes five degrees of freedom, and a bearing joint removes two dof's. In addition, a motion driver is added to the rotational dof of the joint *Pin1*. Therefore, according to Eq. 1.3, the Gruebler's count for the engine example is

$D = 6 \times 3 - (3 \times 5 + 1 \times 2) - 1 \times 1 = 0$

If the Gruebler's count is less than zero, the solver will automatically remove redundancies. In this engine example, if the bearing joint between the connecting rod and the crank shaft is replaced by a pin joint, the Gruebler's count becomes

$D = 6 \times 3 - 4 \times 5 - 1 \times 1 = -3$

To get the Gruebler's count to zero, it is often possible to replace joints that remove a large number of constraints with joints that remove a smaller number of constraints and still restrict the mechanism motion in the same way. *Mechanism Design* detects the redundancies and ignores redundant dof's in all analyses. In dynamic analysis, the redundancies lead to an outcome with a possibility of incorrect reaction results, yet the motion is correct. For complete and accurate reaction forces, it is critical that you eliminate redundancies from your mechanism. The challenge is to find the joints that will impose non-redundant constraints and still allow for the intended motion. Examples included in this book should give you some ideas in choosing proper joints.

Loads

Loads are used to drive a mechanism. Physically, loads are produced by motors, springs, dampers, gravity, tires, etc. A load entity in *Mechanism Design* can be a force or torque. The force and torque are represented by an arrow and double-arrow symbols, respectively, as shown in Figures 1-13 and 1-14. Note that a load can be applied to a body, a point in a body, or between two points in different bodies.

Figure 1-13 The Force Symbol   Figure 1-14 The Torque Symbol

Drivers or Servo Motors

Drivers or servo motors are used to impose a particular motion on a mechanism. Servo motors cause a specific type of motion to occur between two bodies in a single degree of freedom. Servo motors specify position, velocity, or acceleration as a function of time, and can control either translational or rotational motion. The driver symbol is shown in Figure 1-15.
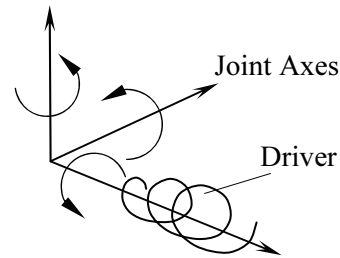
Figure 1-15 The Driver
(Servo motor) Symbol

Note that a driver must be defined along a movable axis of the joint you select. Otherwise, no motion will occur. When properly defined, drivers will account for the remaining dof's of the mechanism calculated using Eq. 1.3.

An example of a typical motion model created using *Mechanism Design* is shown in Figure 1-12. In this engine example, twenty six *Pro/ENGINEER* parts are grouped into four bodies. In addition, four joints plus a driver are defined for a kinematic analysis.

Types of Mechanism Analyses

There are four analysis options supported in *Mechanism Design*: position, static, motion (kinematic, and dynamic), and force balance.

The position (or assembly analysis) that brings the mechanism together, as illustrated in Figure 1-16, is performed before any other type of analysis. The assembly analysis determines an initial configuration of the mechanism based on the body geometry, joints, and initial conditions of bodies. The points, axes, or planes chosen for defining joints will be brought within a small prescribed tolerance.

Static analysis is used to find the rest position (equilibrium condition) of a mechanism, in which none of the bodies are moving. Static analysis is related to mechanical advantage—for example, how much load can be resisted by a driving motor. A simple example of the static analysis is shown in Figure 1-17.
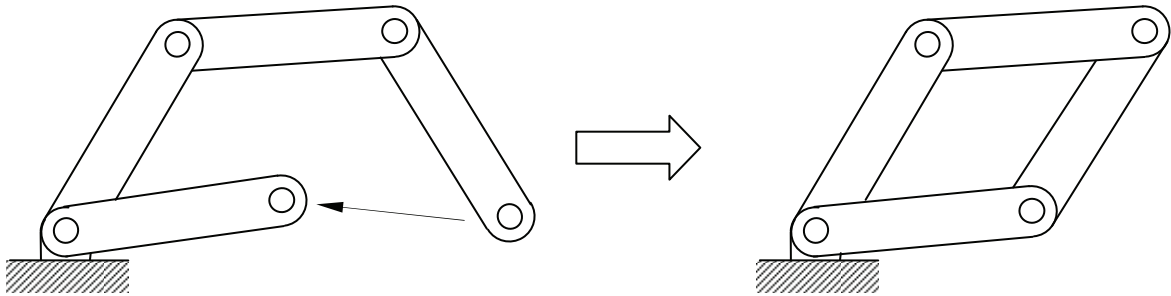
Figure 1-16 Position (Assembly) Analysis
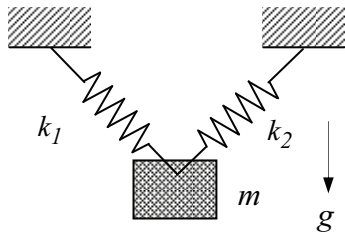
Figure 1-17 Static Analysis
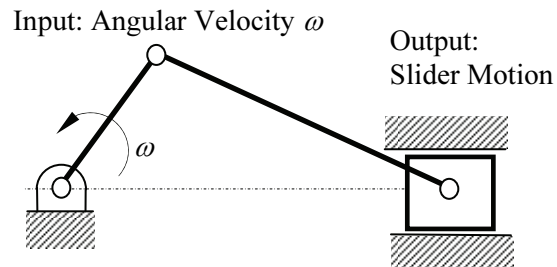
Figure 1-18 Kinematic Analysis

The motion analysis option supports both kinematic and dynamic analyses. As discussed earlier, kinematics is the study of motion without regard for the forces that cause the motion. A mechanism can be driven by a servo motor for a kinematic analysis, where the position, velocity, and acceleration of each link of the mechanism can be analyzed at any given time. For example, a servo motor drives a mechanism at a constant angular velocity in Figure 1-18.
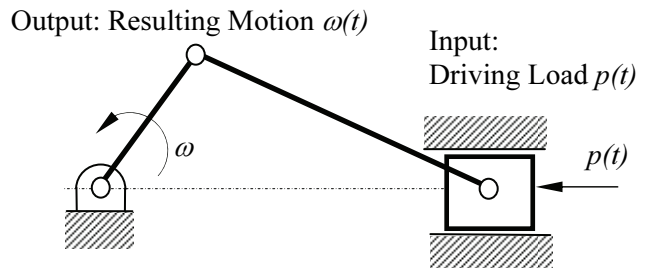
Figure 1-19 Dynamic Analysis

Dynamic analysis is used to study the mechanism motion in response to loads, as illustrated in Figure 1-19. This is the most complicated and common, but usually more time-consuming analysis.

Force balance calculates the required force and torque to retain the system in a certain configuration.

Viewing Results

In *Mechanism Design*, results of the motion analysis can be realized using animations, graphs, reports, and queries. Animations show the configuration of the mechanism in consecutive time frames. Animations will give you a global view on how the mechanism behaves, as shown in Figure 1-20.
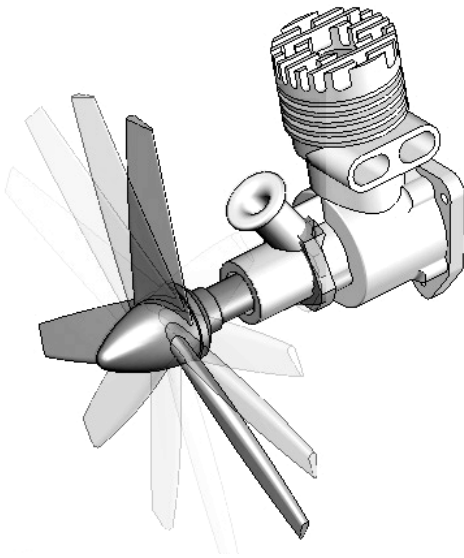
Figure 1-20 Motion Animation
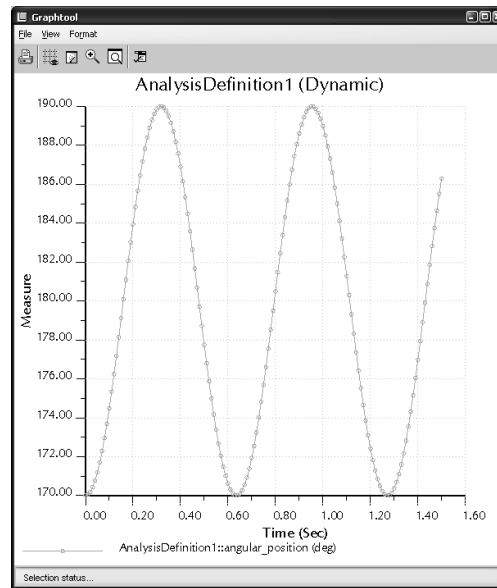
Figure 1-21 Result Graph (Sample)

You may choose a joint or a point to generate result graphs, for example, the graph in Figure 1-21 shows the angular position of a simple pendulum example (please see *Lesson 4* for more details). These graphs give you a quantitative understanding on the behavior of the mechanism. You may also pick a point on the graph to query the results of your interest at a specific time frame. In addition, you may ask *Mechanism Design* for a report that includes a complete set of results output in the form of numerical data.

In addition to the capabilities discussed above, *Mechanism Design* allows you to check interference between bodies during motion (please refer to *Lesson 5*). Furthermore, the reaction forces calculated can be used to support structural analysis using, for example, *Pro/MECHANICA Structure*, a p-version finite element analysis module of *Pro/ENGINEER*.

1.5 Open Lesson 1 Model

A motion model for the single piston engine model shown in Figure 1-1 has been created for you. Download the files from www.schroff1.com, unzip them, and locate the engine model under *Lesson 1*. Copy *Lesson 1* to your hard drive.

Start *Pro/ENGINEER*, set working directory to *Lesson 1*, and open the assembly model: *lesson1.asm*. You should see an assembled engine model similar to that of Figure 1-1.

To enter *Mechanism Design*, simply choose from the pull-down menu

Applications > Mechanism.

You should see *Mechanism Design* window layout similar to that of Figure 1-9. To replay results click the *Replay* short-cut button ◀▶ on the right or choose from the pull-down menu:

Analysis > Playback.

The *Playbacks* dialog box (Figure 1-22) appears. In the *Playbacks* dialog box, click the *Open* button ◻, and select the previously saved playback file *AnalysisDefinition1.pbk* (this file is included in the *Lesson 1* folder). Click the *Play Current Result Set* button ◀▶ at the top left corner. The *Animate* dialog box (Figure 1-23) opens. Click the *Play* button ▶ to play the motion of the engine. You should see the motion animation similar to that of Figure 1-20.
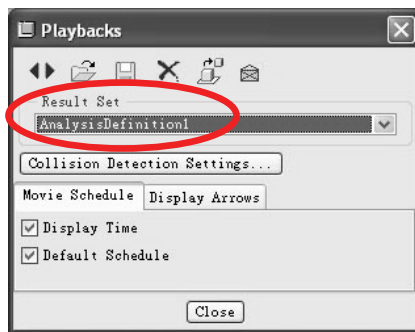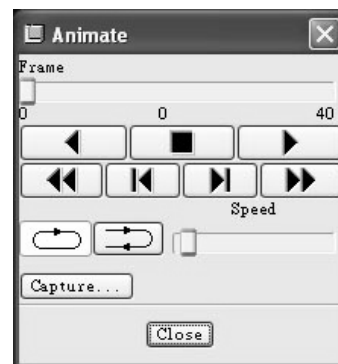
Figure 1-22 The *Playbacks* Dialog Box   Figure 2-23 The *Animate* Dialog Box

1.6 Motion Examples

Numerous motion examples will be introduced in this book to illustrate step-by-step details of modeling, analysis, and result visualization capabilities in *Mechanism Design*. We will start with a simple ball throwing example. This example will give you a quick start and a quick run-through on *Mechanism Design*. *Lessons 3* through *8* focus on modeling and analysis of basic mechanisms. In these lessons, you will learn various joint types, including pin, slider, rigid, etc.; connections, including springs, gears, cam-followers; drivers and forces; various analysis types; and measures and results. *Lessons 9* and *10* are application lessons, in which real-world mechanisms will be introduced to show you how to apply what you learn to more complicated applications. All examples and main topics to be discussed in each lesson are summarized in the Table 1-2.

Note that example files have been prepared for you to go through all the lessons. In addition to *Pro/ENGINEER* parts and assemblies, each lesson folder contains complete motion models as well as simulation result files. You may want to open the motion models and review the simulation results; e.g., play motion animations, to become more familiar with the simulations before going thought the lessons.

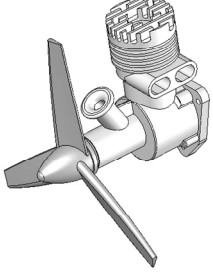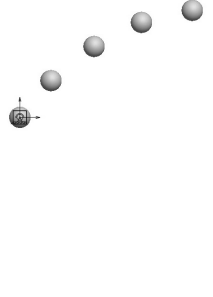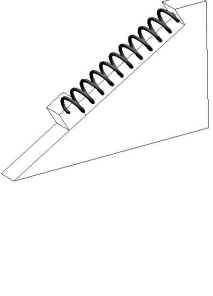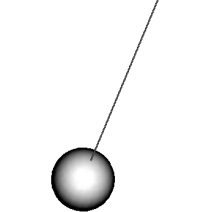Table 1-2 Summary of Lessons and Motion Examples in this Book
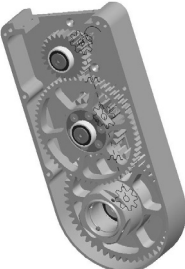
Lesson	Title	Example	Problem Type	Things to Learn
1	Single-Piston Engine		Kinematics	1. General introduction
2	Ball Throwing Example		Particle Dynamics	1. This lesson offers a quick run-through of general modeling and analysis capabilities in *Mechanism Design*. 2. You will learn the general process of using *Mechanism Design* to construct a motion model, run analysis, and visualize the motion analysis results. 3. Simulation results are verified using analytical equations of motion.
3	Spring-Mass System		Particle Dynamics	1. This is a classical spring-mass system example you learned in Sophomore *Dynamics*. 2. You will learn how to create a mechanical spring, align the block with the slope surface, and add an external force to pull the block. 3. Simulation results are verified using analytical equations of motion.
4	A Simple Pendulum		Particle Dynamics	1. This lesson provides more in depth about creating joints in *Mechanism Design*. Pin and rigid joints will be introduced. 2. Simulation results are verified using analytical equations of motion.
5	A Slider Crank Mechanism		Multibody Kinematic and Dynamic Analyses	1. This lesson uses a slider-crank mechanism to discuss more joint types; as well as conduct position (initial assembly), kinematic, and dynamic analyses. 2. In addition to joints, you will learn to create drivers for motion analysis. 3. The interference checking capability will be discussed. 4. Simulation results are verified using analytical equations of motion.

Table 1-2 Summary of Lessons and Examples in this Book (Cont'd)

Lesson	Title	Example	Problem Type	Things to Learn
6	A Compound Spur Gear Train		Gear Train Analysis	1. This lesson focuses on simulating motion of a spur gear train. 2. You will learn how to use *Mechanism Design* to create a gear connection, analyze the gear train, and define measures for gears. 3. Simulation results are verified using analytical equations.
7	Planetary Gear Train Systems		Planetary Gear Train Analysis	1. This lesson is similar to *Lesson 6* but focuses on planetary gear trains. 2. Both single gear and multiple gears systems will be discussed. 3. Some simulation results are found incorrect using analytical equations.
8	Cam and Follower		Multibody Kinematic Analysis	1. This lesson discusses cam and followers. 2. An inlet or outlet valve system of an internal combustion engine will be created and simulated. 3. Position and velocity of the valve will be created to simulate the motion of the system as well as assess the engineering design of the system.
9	Assistive Device for Wheelchair Soccer Game		Multibody Dynamic Analysis	1. This is an application lesson. This lesson shows you how to assemble and simulate motion of an assistive device for playing wheelchair soccer game. 2. Numerous joints, spring, and force will be created for the system. 3. Measures will be defined to assess the design of the system.
10	Kinematic Analysis for Racecar Suspension		Multibody Kinematic Analysis	1. This is the second and the last application lesson of the book. A quarter of a racecar suspension will be employed for kinematic analyses. 2. A road profile will be modeled by using a cam of special profile. The cam will be connected to the tire using a cam-follower connection. 3. Various measures, including the camber angle, will be introduced to assess the design of the suspension system.