MERIT BADGE SERIES

ROBOTICS

STEM-Based

ROBOTICS

"Enhancing our youths' competitive edge through merit badges"

BOY SCOUTS OF AMERICA®

Note to the Counselor

There are many different robot platforms (or frameworks) available to Scouts, and there will be many ways to earn this merit badge. *Any Scout who is interested in earning this merit badge should meet with the merit badge counselor before beginning any work on the robot.* You should both agree on a project to provide assurance that the robot or robotic subsystem the Scout plans to create meets the requirements of this merit badge.

As the merit badge counselor, you will need to determine the following:

1. Based on the difficulty of the project, determine whether the Scout should complete this merit badge as part of a team (if the project is sufficiently complex) or if the Scout's project is simple enough that it should be completed individually.

2. Whether the robot or robot subsystem the Scout plans to build will accomplish a task sufficient for the requirements.

3. Whether the Scout understands all the subsystems of robotics.

An individual robot project will not be as complicated as a team project. A robot that can be assembled by an individual Scout should not qualify for a team of Scouts. If a summer camp or district has programmable kits that Scouts can share and use, each Scout should complete their own task and develop their own robot design that is documented in the Scout's robot engineering notebook.

> Before spending a lot of time and effort on the requirements, the Scout should be sure you will give credit for that work.

SUSTAINABLE FORESTRY INITIATIVE

Certified Chain of Custody
Sustainable Forestry
FPO
SFI-01042

Competitions are a wonderful opportunity for Scouts to work on the Robotics merit badge as part of a team. Scouts in a team competition will likely have learned a great deal about robotics by the end of the season and enough to earn the merit badge. These competitions generally require several weeks of preparation and generally are sufficiently complex to require Scouts to take on specialized roles on the team.

For instance, a Scout may be responsible for an articulated arm that has several motors or actuators. The Scout should be able to explain that part of the robot and the design process of that part of the robot in detail. The Scout may be unable to fully describe the rest of the robot but should be able to describe the robot's other subsystems, including the programming. A Scout must show in a notebook that they designed some part of the robot. Even as part of a team, each Scout should be individually responsible for a design component of the robot. Simply being on a team is insufficient for this merit badge.

Please review the examples in this pamphlet regarding suggested tasks (see "Building a Robot"). A task should demonstrate robot motion, and the use of sensor feedback in the programming at a minimum. Each example requires some form of structured programming, such as a loop, while, or if/then statement. A simple timed task or a task that requires only one program step (e.g., go forward 2 seconds and return, bump against the wall and return) would not fulfill requirement 4.

Many reasonably priced robot kits meet the needs for requirement 4. A remote control car, plane, or boat will not qualify unless extensive modifications are made. Some "robot" kits are no more than puzzles; they create a predetermined robot that completes a predetermined task. There is no room for original design or programming in these robots, and they will *not* meet the Robotics merit badge requirements.

Earning the Robotics merit badge requires a Scout to understand how robots move (actuators), sense the environment (sensors), and understand what to do (programming); the Scout shall demonstrate robot design in building a robot or robotics sub-system. You should help ensure that the Scout has sufficiently explored the field of robotics to understand what it is about, and to discover whether this may be a career field of interest.

Requirement 4b calls for the Scout's robot design to have at least 2 degrees of freedom to ensure the robot has some complexity. For example, a robot that has two motors will have 2 degrees of freedom. A robot that has three output ports is capable of 3 degrees of freedom.
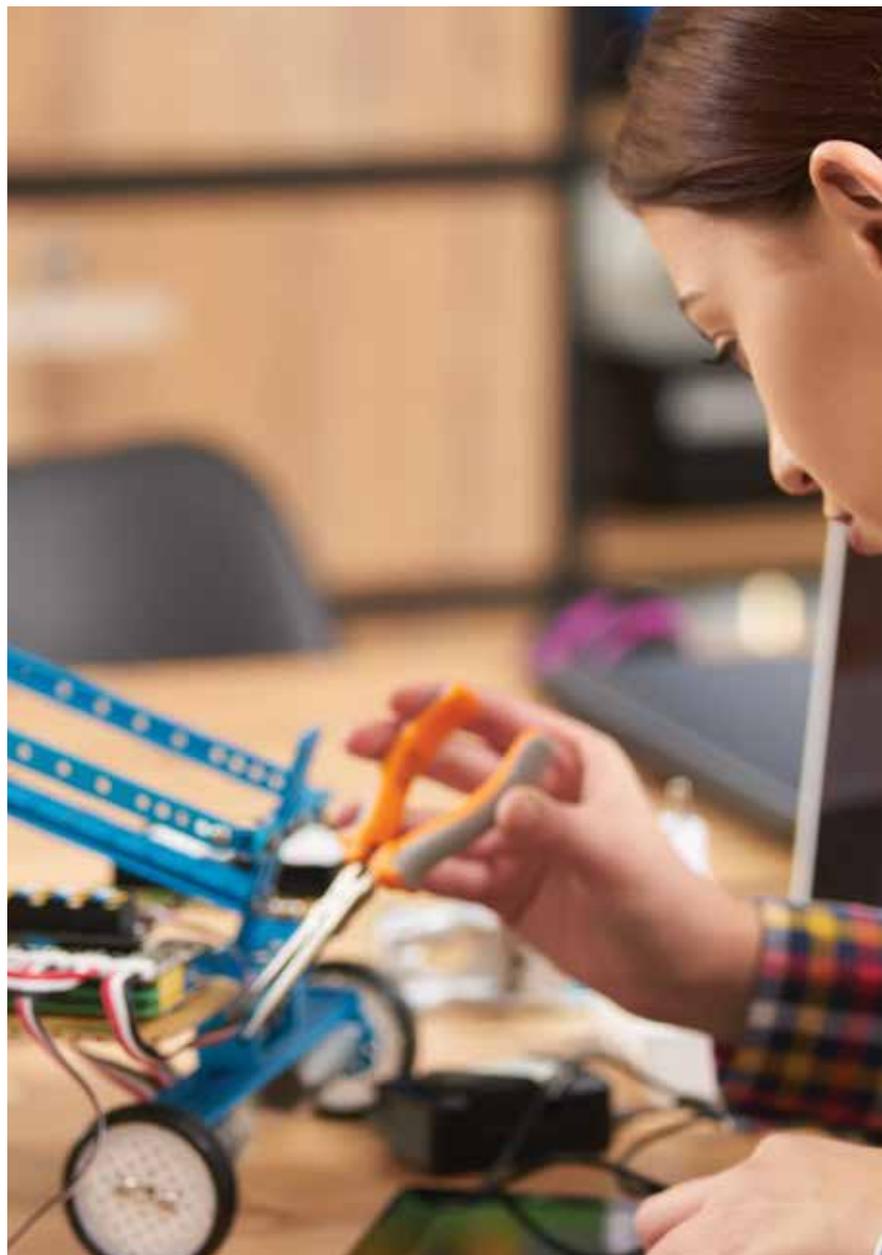
Requirements

Always check www.scouting.org for the latest requirements.

1. Safety. Do each of the following:

   a. Explain to your counselor the most likely hazards you may encounter while working with robots and what you should do to anticipate, mitigate and prevent, and respond to these hazards. Describe the appropriate safety gear and clothing that should be used when working with robotics.

   b. Discuss first aid and prevention for the types of injuries that could occur while participating in robotics activities and competitions, including cuts, eye injuries, and burns (chemical or heat).

2. Robotics industry. Discuss the following with your counselor:

   a. The kinds of things robots can do and how robots are best used today.

   b. The similarities and differences between remote-control vehicles, telerobots, and autonomous robots.

   c. Three different methods robots can use to move themselves other than wheels or tracks. Describe when it would be appropriate to use each method.

3. General knowledge. Discuss with your counselor three of the five major fields of robotics (human-robot interface, mobility, manipulation, programming, sensors) and their importance to robotics development. Discuss either the three fields as they relate to a single robot system OR talk about each field in general. Find pictures or at least one video to aid your discussion.

4. Design, build, program, test. Do each of the following:

   a. With your counselor's approval, choose a task for the robot or robotic subsystem that you plan to build. Include sensor feedback and programming in the task. Document this information in your robot engineering notebook.

   b. Design your robot. The robot design should use sensors and programming and have at least 2 degrees of freedom. Document the design in your robot engineering notebook using drawings and a written description.

   c. Build a robot or robotic subsystem of your original design to accomplish the task you chose for requirement 4a.

   d. Discuss with your counselor the programming options available for your robot. Then do either option 1 OR option 2.

      (1) Option 1. Program your robot to perform the task you chose for your robot in 4a. Include a sample of your program's source code in your robot engineering notebook.

      (2) Option 2. Prepare a flowchart of the desired steps to program your robot for accomplishing the task in 4a. Include procedures that show activities based on sensor inputs. Place this in your robot engineering notebook.

e. Test your robot and record the results in your robot engineering notebook. Include suggestions on how you could improve your robot, as well as pictures or sketches of your finished robot.

5. Demonstrate. Do the following:

a. Demonstrate for your counselor the robot you built in requirement 4.

b. Share your robot engineering notebook with your counselor. Talk about how well your robot accomplished the task, the improvements you would make in your next design, and what you learned about the design process.

6. Competitions. Do ONE of the following.

a. Attend a robotics competition and report to your counselor what you saw and learned about the competition and how teams are organized and managed.

b. Learn about three youth robotics competitions. Tell your counselor about these, including the type of competition, time commitment, age of the participants, and how many teams are involved.

7. Careers. Name three career opportunities in robotics. Pick one and find out the education, training, and experience required for this profession. Discuss this with your counselor, and explain why this profession might interest you.

Contents

What Is Robotics?

From space probes to underwater exploration, from medicine to manufacturing, from law enforcement to search and rescue, robots are highly useful in people's lives today and will be an even bigger part of our lives in the future. Robotics—the technology of designing, building, and operating computer-controlled robots—is a large and growing field.

Robotic surgery requires smaller incisions, causes less discomfort, and lets patients recover faster.

The uses for robots seem almost endless. In agriculture, robots cultivate and harvest fields. In the mining industry, robots do the dirty work of digging and hauling mineral deposits. In microelectronics manufacturing, robots perform precision assembly work where parts must be placed exactly in making components. In medicine, robots perform delicate surgery around nerves, on the eyes, and on other vital organs.

There is no single, agreed-upon definition for robot. Some people believe the only real robot is one that acts on its own or autonomously. Others call virtual computer agents that do things like gather information over the internet "robots." For the purpose of this merit badge, a robot is defined as a machine that performs mechanical tasks using programming, sensors, actuators, mechanical structure, and motion. Remote-control cars, helicopters, boats, etc., or computer simulations are not robots, for the purpose of this merit badge.

In the movies, robots are often depicted as evil androids attempting to replace humans. Modern robots have little in common with these science fiction robots. Robotics designers today focus robot development on tasks that people dislike doing or find difficult.

The Four D's of Robotics

Robotics tasks generally fall into four categories that humans do not want to do or don't do well. These are known as the four D's of robotics: *dangerous, dirty, dull,* and *difficult.*

Dangerous. Robots are used for tasks too dangerous for humans. Bomb squads often use robots to defuse or detonate explosive threats. Robots go places like deep space, miles below the ocean surface, and into collapsed buildings, volcanoes, and combat situations that would be extremely dangerous for people. Soldiers, firefighters, police, and rescue workers use robots to do things that would otherwise put people in danger.

Dirty. Robots are used in dirty places and tasks, such as inspecting sewers, wastewater pipelines, and storage tanks. Robots scrub ships of barnacles. They are also useful in disposing of or containing toxic materials like those at nuclear power plants.

Dull. Robots are good at highly repetitive or dull tasks that humans may find dreary and boring. Robots are used in automotive plants, for example, to make repetitive welds, to paint cars, and to bolt parts together.

Difficult. Robots are used to perform tasks that are difficult for people to do. Some of these tasks are delicate, like surgery; others require heavy lifting and strength beyond what people can do safely, like lifting huge amounts of material in mines.

Five Centuries of Robots

The first robots appeared hundreds of years ago and were not called "robots." They were entertaining mechanical toys that used clockwork or mechanical programming. They could do complex tasks using gears and latches, like a player piano can play music without a musician. The Japanese *karakuri ningyo* toys of the 1700s or Leonardo da Vinci's 15th-century design for a mechanical knight are examples of early robots.

Robot, the Name

The term "robot" was first used in 1921 in the play *R.U.R. (Rossum's Universal Robots)* by Czech playwright Karel Capek. *Robota* is a Czech word that means "forced labor." The story is a familiar one: An inventor builds robots that turn on their human masters. The science fiction writer Isaac Asimov is also credited with inventing the term "robotics" and developing the concept of robots as intelligent machines in his book *I, Robot.*

Leonardo da Vinci is credited with creating the first robot in human form (1495). Here, a mechanical knight is being built using da Vinci's plans.

While truly intelligent machines have yet to appear, robots with humanlike "features"—humanoids—have been created. This humanoid robot, ASIMO (*A*dvanced *S*tep in *I*nnovative *Mo*tion), can walk on two feet.

Artificial Intelligence

In the late 1950s the field of artificial intelligence (AI) was born. An important field for robotics, AI is the study of ways to program a computer or machine to re-create the reasoning, deduction, and problem-solving abilities of humans and animals. So far, this field has had only limited success in matching the intelligence of animals, not the higher-thinking abilities of humans.

The abilities of early mechanical devices only hinted at the eventual potential of robots.

Human and Robot Teams

Some of the most useful robots today complement humans by working side by side with people.

A major focus of AI is understanding the differences between machine computing and human thinking. Machines can recall data and compute math equations far quicker and more accurately than humans. They can also be made more rugged than humans, so they can go places we cannot.

People, however, are creative and can recognize patterns more rapidly than computers or robots can. People generally enjoy being creative and looking for patterns. We do not need or want robots to do that for us.

In addition, humans can adapt to changing situations much better than robots. Most robots cannot even adjust their grip to pick up a hammer or a rake. People learn early how to adapt to changes in our environment, or how to adapt to grip different tools.

This understanding has refocused much recent work in robotics. Instead of attempting to replace human thinking with robot computing, engineers use robot computing to complement human thinking. The robot does what the robot does best, and the human does what the human does best. The combined strengths make for powerful teams.

Industrial Robots

The first industrial robot, Unimate, was installed in a General Motors manufacturing plant in 1961. The Puma series of robots soon followed the Unimate onto the manufacturing floor. These robots kicked off the industrial robotics revolution seen in manufacturing today. They do jobs that are too dangerous or too dull and repetitive for people. They are also faster and more accurate than humans. Because of the gains in speed, accuracy, and efficiency, robots can reduce the overall cost and increase the quality of manufactured products.

Robots as Explorers

Robots have become our eyes and ears in new worlds. Robots can be made more rugged than humans. They also do not mind being sent to distant places, possibly to be destroyed and never to return.

Robots have explored the extreme depths of our oceans and the farthest reaches of our solar system. Jason Jr. is an underwater robot that helped to discover and explore the sunken *Titanic* in 1986. NASA scientists and engineers have used space robots to explore the sun, comets, asteroids, and planets of our solar system. In 2004 the rovers Spirit and Opportunity landed on Mars. While they were designed to last only 90 days in the harsh Martian environment, Spirit lasted seven years and Opportunity lasted 15.

The Robotics merit badge logo was based loosely on NASA's Mars rovers Spirit and Opportunity.

Robotic explorers have increased our knowledge of our little corner of the universe.

Personal Robots

For years people have dreamed of owning a robot to do their dull and dirty jobs. The Roomba®, a robotic vacuum cleaner introduced in 2002, has become the most successful of a group of robots known as personal robots. The Roomba is inexpensive enough that many people can afford one to clean their floors. There are also robot lawn mowers, robot pool cleaners, sentry robots to watch your home when you are away, and gutter-cleaning robots.

The category of personal robots, like this Roomba® 560 vacuum cleaner, is growing slowly but surely.

Major Fields of Robotics

Sophisticated robot design typically requires the talents and expertise of teams of engineers. There are five fields of expertise in robotics.

Operator Interface. A robot must be able to communicate with its human controller. The interface is the mechanism of communication between the person and the robot. For example, a joystick controller for a video game is an interface used to communicate with the game.

Mobility or Locomotion. Locomotion is how the robot gets from place to place—how it moves in its environment. Mobility can be achieved with wheels, legs, fins, propellers, and much more.

Manipulators and Effectors. The parts of the robot that interact with objects might touch things, pick them up, place them in containers, spray them with paint, and more are called manipulators or effectors. Examples include claws, pushers, and mechanical arms and fingers.

The *operator* is the human controlling the robot.
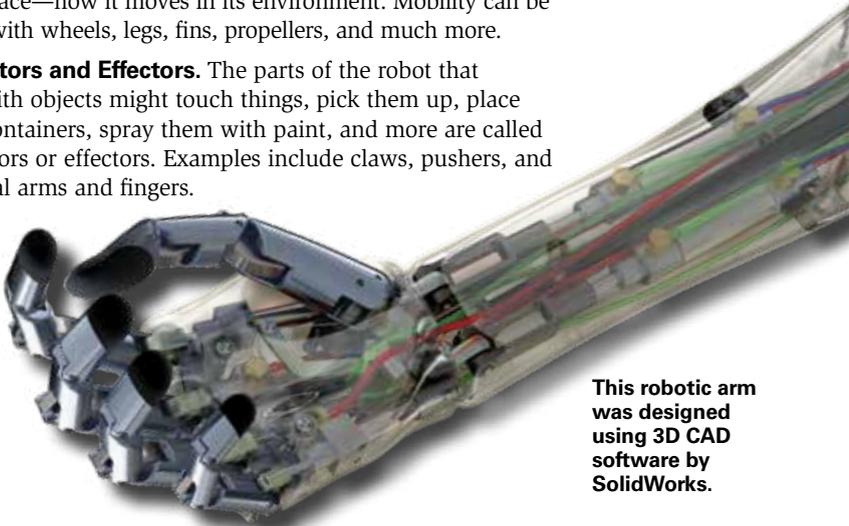
This robotic arm was designed using 3D CAD software by SolidWorks.

Programming. Programming is how you talk to a machine. Some forms of advanced programming allow a robot to learn and adapt to changes in its environment.

Sensing and Perception. A robot needs information from sensors to know where it is, to go where it needs to go, and to avoid obstacles.

## Operator Interfaces

Autonomous Interfaces

An *autonomous* robot is controlled by its internal computer. An autonomous robot system is much like the human autonomic system (a part of the nervous system that controls involuntary actions like heartbeat, balance, and breathing). It operates and regulates itself without our awareness or interference.

A common computer and robot interface is a keyboard. Autonomous robots are generally programmed by the operator using a keyboard. The program directs the robot through the *controller,* a small computer on the robot.

Another common programming interface for autonomous robots is a *teach pendant.* Teach pendants are similar to a handheld video-game controller that has a small video screen. These controllers have many control options but not as many as a keyboard.

With a teach pendant, the programmer programs the robot using pictures rather than words. Programming with pictures is called *graphical* programming. A teach pendant is an example of a *graphical user interface* (GUI).
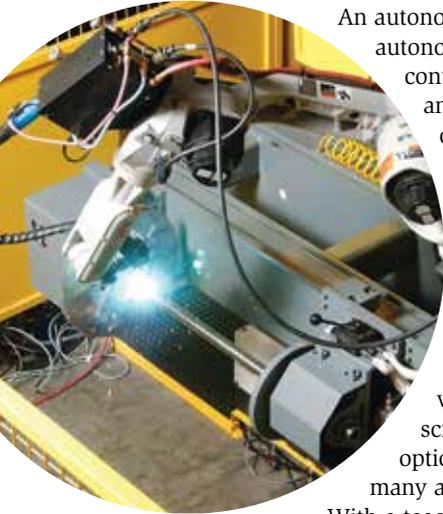
Most robots in manufacturing are autonomous.

Robotic arm teach pendant

A robot may have many arms that can move in many directions. Controlling all the different arms at once can be difficult. To simplify things, a master and slave control system is used. The master is a smaller version of a robot that the operator moves around directly. The positions of the master unit are then sent to the robot, which is the slave. This makes for an intuitive and easily trainable interface for a robot.

Teleoperator Interfaces

A *teleoperated* robot (telerobot) is controlled by a human using a control device at a remote location, also known as remote manipulation. Teleoperation and remote control are similar. Remote-control cars and airplanes are controlled completely in real time by a person. Teleoperated robots are much more complex, however. They receive some commands from the operator, but they also have sensors that sense the environment and provide feedback to the operator. Such robots are programmed to react to the environment if necessary.

A telerobot may perform a task a few yards away and as far as miles away from its human operator.

The interfaces mentioned so far can be used to teleoperate a robot as well as to program a robot to do autonomous tasks. The following are some unique interfaces for teleoperation.

Virtual reality (VR) systems immerse the operator in a real-time control environment. NASA's Robonaut system, for example, can use a teleoperation control system with gloves and a VR helmet to control a pair of dexterous hands and arms. The VR helmet receives video from the cameras placed in the robot's head where the eyes would be. The VR gloves control the movement of the robot's arms and fingers: When the operator moves an arm or finger, the robot moves its arm or finger. These controls allow operators to feel as though they are within the body of the robot and directly controlling it.

NASA's Robonaut made a guest appearance at the 2010 National Scout Jamboree in Virginia.

Biofeedback or *neurofeedback* is another way to teleoperate a robot. This technology uses electroencephalogram (EEG) sensors to directly detect the nerve activity of a human body and interpret it as commands to a robot. The user thinks "arm move," and the robotic arm moves. This method of control is increasingly common for prosthetic (artificial) limbs that are used by people who have lost an arm or leg.

An interesting human-machine interface is the *exoskeleton.* An exoskeleton robot is a wearable robot that gives the wearer increased strength and endurance. You may have seen these featured in such movies as *G.I. Joe, Ironman,* and *Avatar.* Exoskeletons are their own interface. As the human wearing it moves, so does the robot. This interface also provides nonvisual feedback to the operator. For example, an operator wearing a properly designed exoskeleton can lift weights far greater than would be possible for the person to lift without the exoskeleton. The operator will feel a weight in the arm, but does not feel its full weight.

> Prosthetics are becoming more and more robotic—robots controlled by the user's mind and nerves.

Feedback systems that provide force information directly to the operator through sensors are of two different forms: *haptic* and *force reflecting.* Force-reflecting feedback directly transmits the force the robot feels back to the operator in the same form. As the robot arm pushes something, the human controlling it feels the same push, even if the robot is miles away. Haptic feedback changes the information and provides it in a different way. Instead of the human's arm being pushed, the operator may feel a vibration.

Telerobotic interfaces all provide some form of feedback to their operators: force feedback, for example, or visual display information from cameras, or at least text information like speed, temperature, and location. All this feedback comes from sensors that monitor and report the environment to the robot or operator.

Robotic exoskeleton developed by the U.S. Army

Mobility or Locomotion

To do their jobs, robots need to move. They can move in many different ways. Some need to move only mechanical arms or grippers. Others need to be fully mobile, able to travel from place to place. The parts of the robot that create motion are called *actuators.* There are many types of actuators used in robots including motors, servos, and pneumatic cylinders.

Electric Motors. Most robots use electric motors. These popular actuators are typically used to rotate a wheel, gear, or propeller. Direct current (battery-operated) motors are often used in portable or mobile robots; alternating current motors are used in industrial, fixed-position robots.

Servos. Servos are similar to motors in that they have a rotary action and work on electricity. But instead of just rotating continuously, they can also rotate very accurately to a specific position and stop. Servos are used to rotate gears, joints, and fins. They range from small hobby servos for remote-control airplanes to large industrial servos that are used to move objects weighing many tons.

Pneumatic or Hydraulic Actuators

Linear Actuators. Linear actuators are used to push or pull structural elements instead of rotating them. These types of actuators can exert a tremendous amount of force very quickly in one linear (straight) direction. These use pressurized air (pneumatics) or fluid (hydraulics) to cause their motion.

Rotary Actuators. A rotary actuator generates a rotational force that can be used to move robot joints. Rotary actuators work in much the same way as linear actuators, except the push is circular rather than straight.

Energy for robot movement can come from many sources, including electricity, springs, or air or fluid pressure. Energy sources are described in "The Power Subsystem," later in this pamphlet.

Other Actuators. Other types of actuators include:

- Muscle wire—a type of wire that contracts (shortens) when electricity is run through it. Also known as shape memory alloy (SMA), Nitinol, or Flexinol®, muscle wire typically contracts less than 5 percent and is useful in small robot applications.

- Electroactive polymers—a plastic material that can contract up to 400 percent when electricity is applied. These polymers are popular for robotic facial muscles and the arms of humanoid robots.

Locomotion

Robots can move around in many ways using actuators.

Two-Wheeled Balancing. By using sensors and feedback, a robot can keep track of its orientation and maintain its balance on two wheels. An example is the popular Segway®, a self-balancing electric vehicle that can keep its balance even while scooting sideways along a sloped surface.

One-Wheeled Balancing. Some robots ride on a single wheel or ball and still keep their balance.

Six-Wheeled Robots. A Mars rover or other all-terrain robot is an example of a six- or eight-wheeled robot.

A Segway® personal transporter is an example of two-wheeled balancing.

NASA's Mars rover, the wheeled Curiosity, can travel easily along multiple types of terrain.

The BigDog "pack mule," a four-legged robot, is designed to help carry supplies for U.S. military troops.

It's important to design and equip a robot for its intended purpose. For instance, robots that mimic the motion of a snake may be able to enter small crevices of buildings destroyed by earthquakes to look for survivors.

Walking Robots. A robot that can walk on two legs (like the Honda ASIMO) is called a *biped.* Some robots have more than two legs and can always have multiple legs on the ground to maintain their position and not risk losing their balance. The U.S. Army's BigDog robot, a four-legged cargo carrier, can move across rough terrain that would stop a wheeled vehicle.

Flying Robots. Unmanned aerial vehicles (UAVs) are flying robots. In the military, small UAVs fly in front of patrols so that soldiers can spot potential ambushes. UAVs with special sensors can patrol large areas to spot the enemy setting mines or booby traps. Flying robots are also useful for surveillance in remote areas. The U.S. Border Patrol uses unmanned aircraft along America's borders.

Unmanned aerial vehicle

Underwater Robots. Robots are used underwater for exploration, repairs, mapping, recovering items like the flight data recorder of an airplane that has crashed in the ocean, or cleaning up oil spills.

Climbing Robots. Some climbing robots can adjust their center of mass and move each limb in turn to gain leverage, just like humans. Another form of climber is the gecko-style robot that uses a special pad to climb even smooth surfaces.

Manipulators and Effectors

Besides moving from place to place, robots need to manipulate objects—pick up, modify, or otherwise move or have an effect on objects. There are many types of manipulators and effectors.

Manipulators also use actuators to accomplish their tasks.

Some people refer to the hand of a robot as an "effector" or "end-effector," and refer to the arm as the manipulator. Many robots have interchangeable end-effectors so the robot can be adapted to different uses. In a car factory, for example, the same manipulator might be used to weld a metal seam and to paint the car. The only difference is that each job uses a different end-effector.

Mechanical Grippers. The gripper is a common type of effector. It can be as simple as two fingers that open and close like a claw, or as complicated as a mechanical hand.

Vacuum Grippers. Vacuum grippers use suction to manipulate objects. They are used in a wide variety of applications: to place tiny integrated circuits on electronics boards and to place the windshields on automobiles in a car factory, for example. The main requirement for vacuum grippers is that they have a smooth surface to which they can attach themselves.

Magnetic Grippers. Magnetic grippers use electromagnets to hold and manipulate objects. These are versatile and can control heavy loads, but the objects must be made of a kind of material (iron, usually) that a magnet can pick up.

Ingressive Effector. An ingressive effector uses pins, needles, or hackles that penetrate the surface of the object. These are common for handling textiles and glass fibers.

Programming

A robot generally has an onboard computer that controls it. This requires writing a program in the language the robot speaks.

You are probably familiar with many computer programs: word processors, spreadsheets, and presentation software, for example. *Programming* is the process of creating these programs: writing software in the computer's language. For robots, programmers can use various computer languages. An interesting one often used in artificial intelligence applications is LISP; it can learn from its surroundings.

Sensors

A robot must constantly ask, "Where am I? Where do I need to go? What is in my way?" To answer these questions, the robot uses sensors, which do many of the same things that human senses do. Robots use cameras for eyes and microphones for ears. They use bump sensors to touch. They have temperature sensors to "feel" heat or cold. Other special sensors go beyond human senses.

- The Global Positioning System (GPS) can provide a robot or its operator with the robot's coordinates on a map.

- Radar transmitters can detect objects in the robot's way. Radar bounces radio waves off objects to determine their distance and speed from the radar device.

- Light detecting and ranging (LIDAR) sensors can be more precise than radar at identifying objects. LIDAR uses light instead of radio waves. LIDAR can identify several objects and their distances at once.

- Infrared sensors can detect heat and are used to see at night or in the dark.

You are strongly encouraged to try programming. It will open up worlds of possibilities when you can command the robot to do anything you want. For details and examples, see "The Controls Subsystem" section later in this pamphlet.

Sensors and programming work together to tell the robot where to go and what to do: They make up the controls subsystem. Sensors detect the physical world and convert that information into electrical signals. These signals are transmitted to the computer program that controls the robot in the physical world. In autonomous applications, the program determines the next move. In telerobotic applications, the sensor data is sent back as a signal to the human operator. As the robot moves in the physical world, the sensor values change, and the process repeats until the program ends.
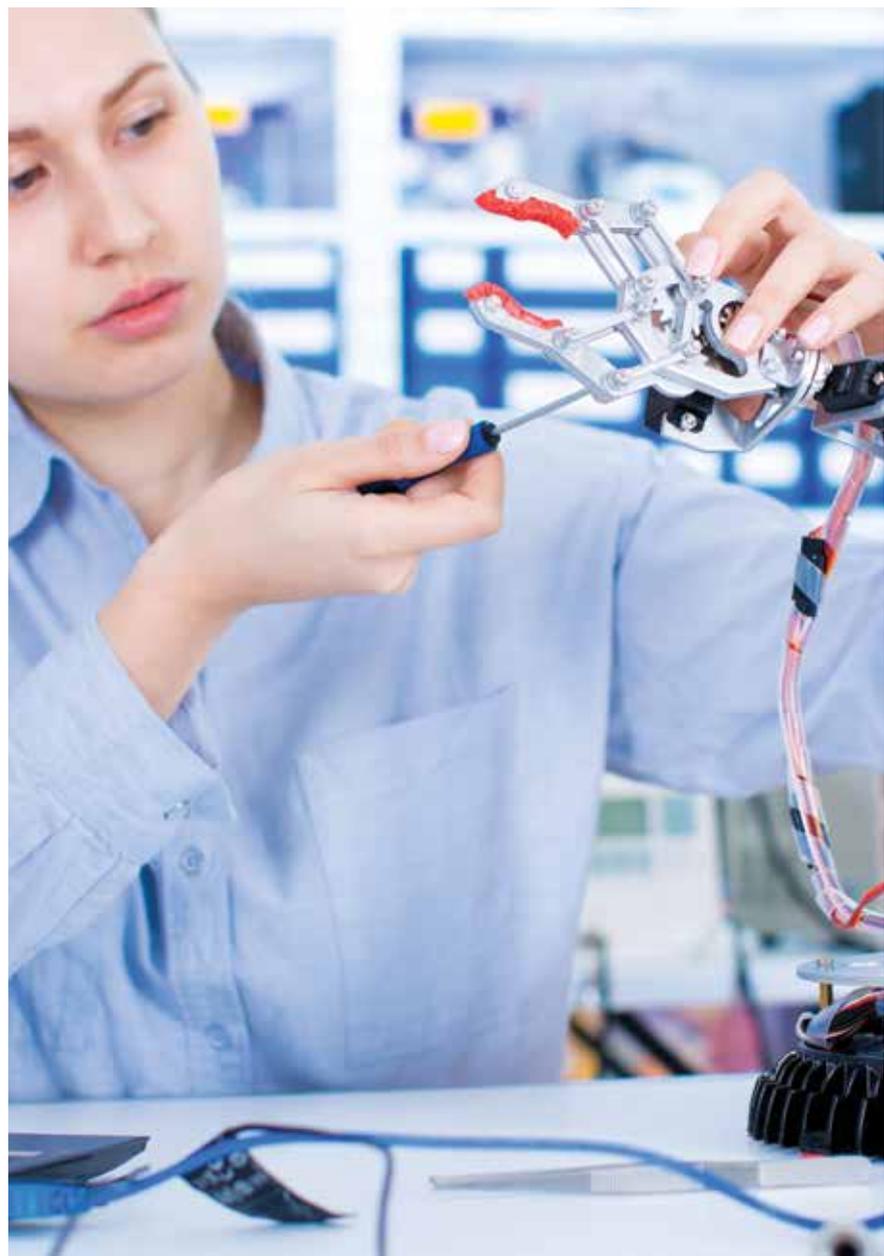
Building a Robot

For this merit badge you will design and build a robot or a functional part of a robot. Building a robot of your own is an exciting project, but getting started can be challenging. Fortunately, you do not have to build your robot from scratch or alone.

> The robot you build for this merit badge may be complicated. If so, you may be able to work on a team to complete the requirements. Discuss this with your counselor before you get started. In any case, you need to be familiar with the mechanisms and structure, the programming, and the sensors of your robot, and you will still be required to demonstrate your individual knowledge.

Before you begin building your robot, be sure to discuss the project with your counselor. Discuss the type of robot you want to build, the task you want your robot to complete, and the safety considerations of building your robot.

Building larger robots requires the use of safety glasses, gloves, or other safety equipment depending on the complexity of your project. In addition, your merit badge counselor may suggest that a parent or adult or another Scout be present during the building of your robot. This is in the Scouting spirit of two-deep leadership and the buddy system—always a good practice.

⚠ Safety Precautions

Robotics combines many different fields of engineering and science like electrical engineering and mechanical engineering. When working with a robot, you must be concerned with electrical safety, materials safety, chemical safety, and power tool safety.

Robotics and First Aid Basics

Discuss with your counselor all the potential hazards stemming from your project. Know the proper first aid for cuts, burns, and eye injuries.

Cuts. To avoid minor cuts and scrapes, be sure you dress appropriately when working on your robot—for instance, jeans, boots, gloves, and so on. Treat minor cuts and scrapes by flushing the area with clean water for at least five minutes, or until all foreign matter appears to be washed away. Apply triple antibiotic ointment (if appropriate and if there are no known allergies) and cover with a dry, sterile bandage.

Burns. Working under close supervision of a responsible adult and being careful when you work with materials that can cause burns will help prevent accidents. For chemical burns, it's important to quickly brush off as much of the chemical as possible using a gloved hand. Flush the area with tap water, taking care not to recontaminate the area. For superficial (first-degree or minor) burns, hold the burn under cold water or apply cool, wet compresses until the pain eases. Cover the area loosely with a sterile gauze pad and bandage. For more serious burns, seek immediate medical attention.

Eye Injuries. Wearing safety glasses or goggles will help protect your eyes from fumes, chemicals, debris, and other foreign matter. If a foreign object gets in the eye, do not rub the eye. Blink the eyes to help tears flush out the object. If that doesn't work, wash your hands with soap and water, then try to flush out the foreign particles with clean running water or clean water poured from a glass or bottle. For more serious injuries, seek immediate medical attention.

Safety First

Before beginning to build, fix, or work on moving parts, be sure the robot's energy sources are disconnected:

- Turn the power switch off.
- Disconnect the batteries.
- Unplug the power cord.

Robot Design Kits

You can complete the requirements for this merit badge using robot design kits, many of which can be put together with snaps or screws. Robot construction kits provide all the parts you need: structural parts (bricks, beams, fasteners), mechanical parts (gears, motors, actuators), controllers (small computers), and sensors. These are design platforms. They allow the builder to create any number of robots that can accomplish various tasks. Examples of these design-and-build kits include the VEX® Robotics Design System, Lynxmotion's Servo Erector Set, LEGO® Mindstorms®, Robotis, Parallax Boe-Bot®, and Fischertechnik.

Parallax Boe-Bot robot

Robot created from VEX Robotics kit

Most of these kits provide parts that are designed to work together with the controller and sensors. Some have all the parts you need to create many different robots. Others include a standard platform that moves and has a controller. To add functionality, you must purchase additional parts separately. Both approaches offer many options for designing your own robot.

The advantage of using a design-and-build kit is that you are working with parts that are made to work together, so you spend your time imagining and building your robot rather than struggling to fit things together. The disadvantage is that these kits can be expensive.
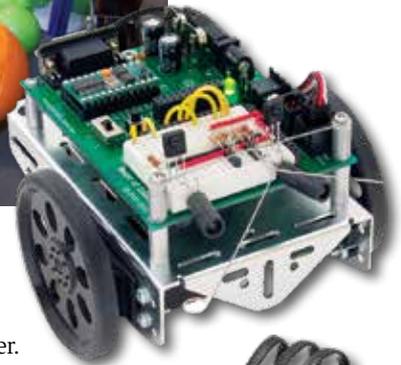
Robot constructed from a LEGO Mindstorms kit

Caution: Some robot kits are like puzzles. They are to be put together in a specific way and can accomplish only predetermined tasks. These robots are typically designed to teach robotics programming or operation, but not mechanical design and construction. Such robots and kits *do not* fulfill the requirements for the Robotics merit badge. To earn the badge, you must design your own robot or part of a robot.

Be sure you discuss your project with your counselor before you get started. That way, you can both determine an appropriate project that will meet the needs of the Robotics merit badge requirements.

Hybrids

If you feel comfortable with electronics, you may try to add a controller to a toy or machine to make it into a robot. Remote-controlled cars, boats, and airplanes can be modified to become robots by adding a controller or central processing unit (CPU). This is not easy. Find someone who has done this before to help you.

In industry, modifying existing machines is common practice. Researchers are automating products to make them robotic, creating smart vacuum cleaners, automated lawn mowers, and robot cars like this one.

Building From Scratch

Building a robot from scratch is not only satisfying, but can also be less expensive. You can make a good robot from a robotics controller, servo motors, and fabricated components. Resources available online and in books can help walk you through the process (see "Robotics Resources" in this pamphlet). If you choose this option, be sure to work with someone who has previously built a robot from scratch.

Many builders start with a robot kit or building kit and add their own parts. PVC pipes are great for underwater robots. Cardboard and discarded polycarbonate make excellent building materials. (Look for polycarbonate plastics rather than acrylic plastics for robot use. Acrylic plastics like Plexiglas might look better than polycarbonate plastics because they are clearer and resist scratches. However, polycarbonate plastics like Lexan are better for robots because they resist breakage and do not shatter when broken.)

When you start from nothing, you begin to realize that many things lying around your home or neighborhood can easily become part of a robot.

Being Part of a Team

In most cases, being on a robotics team will give you opportunities to complete the merit badge requirements. You may learn more on a team than by working alone.

Putting most commercial robots together takes a team of skilled engineers. A Scout, however, does not have to wait to become an engineer to build robots. More than 300,000 students participate each year in robotics competitions around the country. Many other youths build robots as a hobby.

Often, each member of a robotics team is assigned a responsibility. If you have an assigned responsibility, you need to show how you designed your part of the robot. You need to develop an idea and design a solution. You will also need to understand how the robot subsystems work together and be able to describe the chassis, mechanical, power, and controls subsystems. (The subsystems that make up a robot are discussed individually, later in this pamphlet.)

Robot Tasks

Robots can do many tasks. Those that qualify for this merit badge require programming, movement, and sensors to respond to the environment, and use an attachment to interact with the environment in some way. Most robotics competitions present several tasks that satisfy this requirement. (See "Robotics Competitions," later in this pamphlet.) Here are other examples of acceptable tasks.

Sumo Robot. A sumo robot can identify another robot in a circle and push the other robot out of the ring. To accomplish this, it needs to use a sensor to sense the circle and the opposing robot. It must be able to move in the direction of the other robot to push it out of the ring. Finally, the sumo robot must have an attachment to push the other robot out of the ring. You could modify this to a single task: Use the same type of circle from the sumo robot competition. Place 10 cans in the circle and have your robot remove them as quickly as it can.

Rescue Robot. Send a robot to save a "human" toy that has fallen off a table. The robot moves to the edge of the table and stops. It then lowers a string to pick up the human. Finally, the robot rolls the string back up and returns to the base.

Basketball Robot. Program your robot to shoot baskets with a toy basketball. The robot should be able to identify the basket, throw the ball, and then retrieve the ball.

Butler Robot. The robot needs to follow a path (a line on the ground) to find the spot to deliver a beverage. It needs to hold the drink, deliver the drink, and return. You can vary this activity to deliver just about anything—perhaps make a warehouse robot that stores materials in bins.

Energy-Saver Robot. The robot tracks to a light source and turns off the light via a switch.

Greeter Robot. Create a robot to hear the doorbell, move to the door, and extend a hand to welcome guests.

The projects mentioned here are only suggestions. You may want to be more creative with your task—build a robot to cap a simulated oil well deep in the "ocean," for example, or build a robot that "plays" basketball and "shoots" hoops.

The Design Cycle

The design cycle is the process that engineers use to create new things or improve existing things. The design cycle is a creative process, and like robotics itself, its definition is not universally accepted. For the purpose of this merit badge, the design cycle is defined in the following way.

Identify and define the problem. Everyone involved needs to identify the problem and share the same precise definition of the problem that the robot will address.

Understand the problem and pinpoint the constraints. This is a step of further refinement of the problem to determine what is possible at present, the limitations (such as resources), and what will have to wait for another time.

Analyze the problem. Analysis can take many forms. This is where you research your problem for what already exists, what has already been tried, what works, and what does not work from past experience and data.

Evaluate and brainstorm the problem. During this step, you generate ideas and consider a number of possible approaches to the problem. You need to evaluate which one approach your team will pursue.

Synthesize and select the approach. Identify the best approach (what), specify the implementation (how), and build your product.

Communicate the product. You show your product to others to test it.

Refine the product and repeat the design cycle. This is a critical step, and this is why the design cycle is circular. Once a team has created something, the team will begin the design cycle over again. This is why an engineer's job is never finished. It is also why a new version of a television, mobile phone, or car appears every few months.

It is critical for engineers to document their design steps in an engineering notebook, because it is the primary evidence used to grant a patent. The steps the engineers take, and the tests they do, need to be documented clearly and precisely so that others can understand what they did.

An innovative design can provide engineers or their companies with continual income through patents. A *patent* is an idea that belongs exclusively to an individual or a company. If another company wants to use the idea, it must pay the inventor or patent holder for the right to use it. For more information about patents, see the *Inventing* merit badge pamphlet.

Robot Engineering Notebook

You will need to keep a robot engineering notebook, just like a robotics engineer's. You will use the notebook to document the completion of each requirement for the Robotics merit badge. The purpose of the notebook is to document the design process for your robot. The documentation will be helpful as you improve the robot design. It is also a valuable way to communicate to others, like your counselor, what you did.

Many notebook formats are suitable. A composition book makes a good design notebook. The kind with graph paper allows you to draw scale pictures of your robot designs. You should draw several pictures of your designs as you develop them. Use the squares to represent a measurement scale, like an inch or a centimeter. Using the squares will give you more information about the final design size and the placement of parts than would a free drawing without the squares.
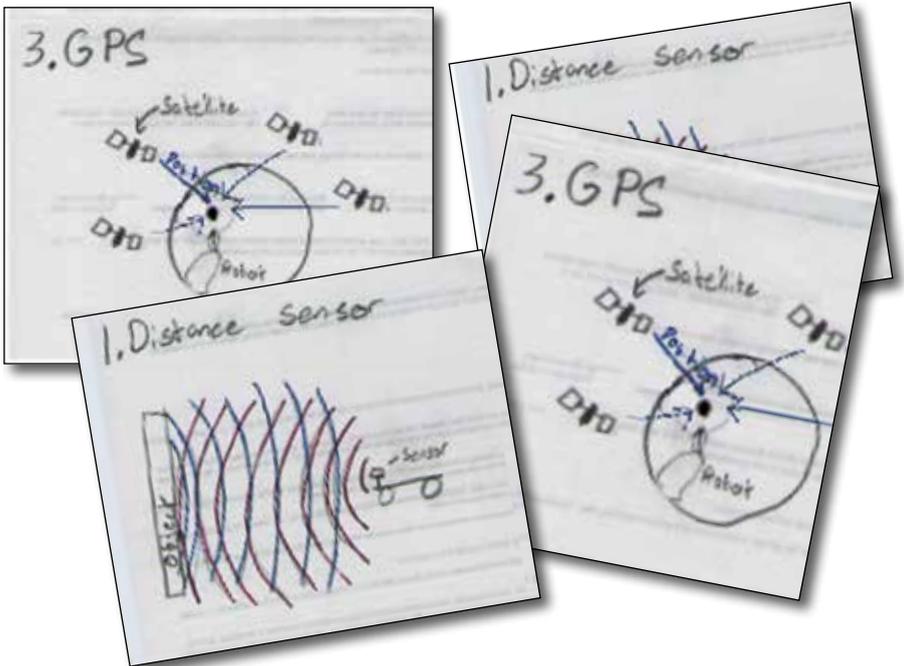
For your robot engineering notebook, collect pictures of robots (from the internet, for example) so that you can discuss them with your counselor.

Robot Engineering Notebook— Required Sections

Put your name in your robot engineering notebook, and include these sections:

- **Robot.** Describe the type of robot you want to build (flying, underwater, boat, land, tank drive, wheeled, etc.).

- **Task.** Describe in detail the task the robot will accomplish.

- **Several Possible Design Ideas.** Discuss with your counselor several possible ideas for your robot design before building it.

- **Tests.** Describe the tests and the designs that did not work that lead to the final robot design.

- **Pseudo Code.** Prepare a detailed flowchart of the step-by-step commands the robot must complete to accomplish the task. (See "The Controls Subsystem" section of this pamphlet.)

- **Software Code.** Include the code or flowchart you used.

- **Code Modifications.** Include any changes you make to the pseudo code or the actual code so that the robot completes the task better.

- **Final Robot Design.** Include a description or picture of what you decided to build.

- **Potential for Improvements.** Describe how you can improve your robot for the next design version.

Feel free to add more to your notebook—notes on other robot designs, cool attachments, other robot languages, or anything else you want.

Show your counselor the notebook at several points as you design your robot. Before you begin building your robot, discuss with your counselor the "Several Possible Design Ideas" section of your notebook. Your counselor can help save you from the time and trouble of working on a faulty design.

The Chassis Subsystem

Just as your body needs a skeleton, your robot needs a well-designed chassis or frame. The chassis gives a robot shape, structure, and strength; protects the robot's computer and other vulnerable components; and provides a place for attaching the robot's other parts.

The chassis can be made from various materials, including wood, metal, plastic, PVC, or composites. The choice of materials depends heavily on the robot's environment (land, water, air, space, etc.). Choose materials that are strong and appropriate for the task at hand.

Remember that the design process is repetitive. Experiment with several different kinds of chassis to see what works best for your robot.

As you build your robot, keep these things in mind: stability, robust design, vulnerability, and interaction with other system components.

Stability

Center of Gravity. You can think of a robot's center of gravity as the average position of all the weight on the robot. Because it is an average of weight and position, heavy objects count more than light ones in determining where the center of gravity is, and pieces that are farther out count more than pieces that are closer to the center of the robot.

CENTER OF GRAVITY

SUPPORT POLYGON

POINT OF CONTACT BETWEEN WHEEL AND GROUND
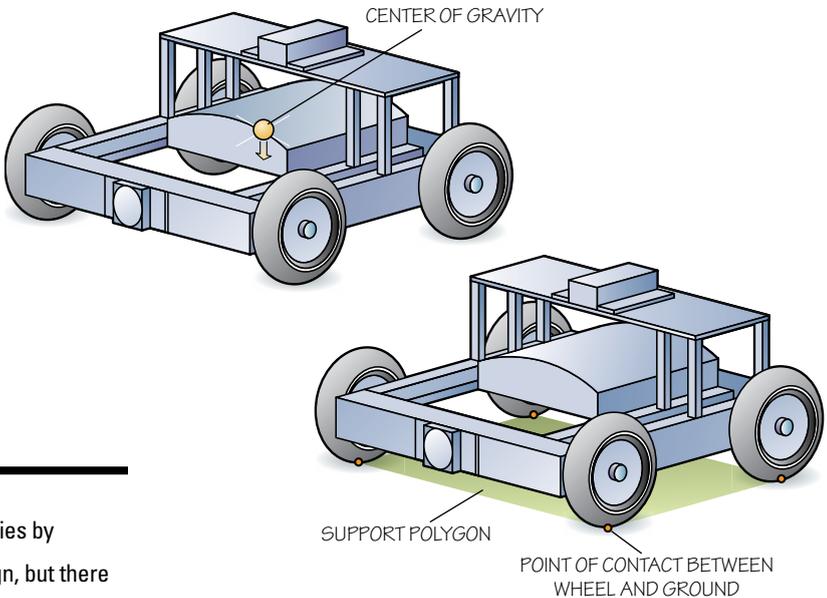
It varies by design, but there is always one support polygon in any stable configuration.

Support Polygon. Squares and rectangles (four-sided figures) and pentagons (five-sided figures) are examples of polygons. The support polygon of a robot is the imaginary polygon formed by connecting the points where the robot touches the supporting surface (the ground, water, air, etc.).

A triangle is the most stable stationary polygon. The most stable sitting or standing object will have three feet like a tripod.

Robots, however, do not sit or stand still; they move. Three points of contact are not stable when a robot is moving. Four points work better. This is why most land robots have four points like wheels touching the ground or surface, not three.

CENTER OF GRAVITY
NOT DIRECTLY OVER
SUPPORT POLYGON
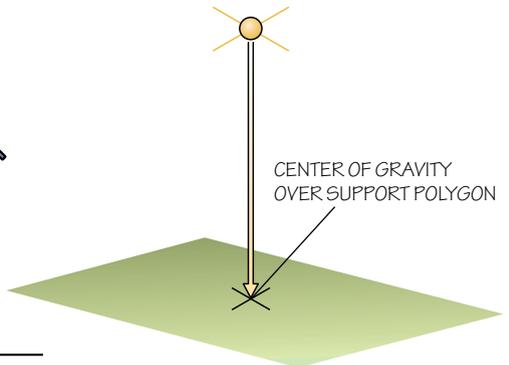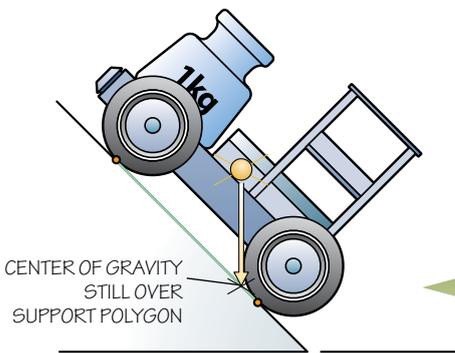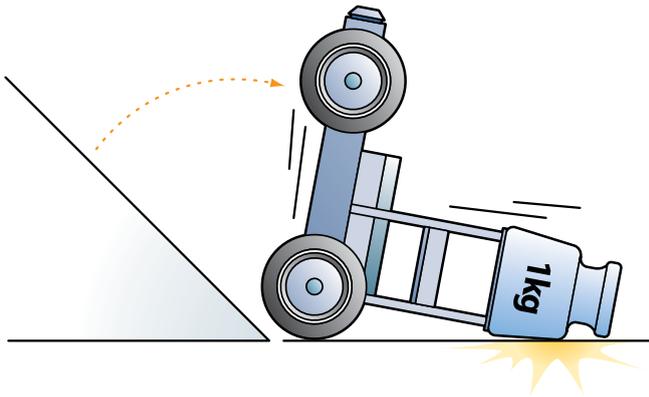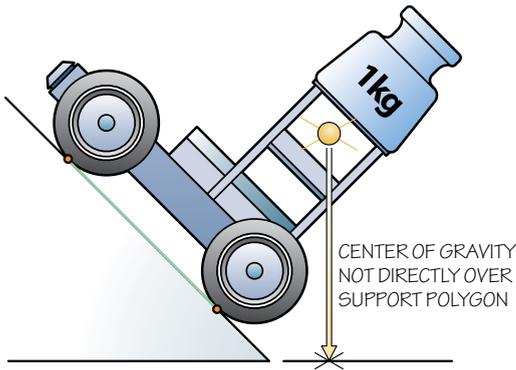
A robot will
be most stable
when its center
of gravity is over
the robot's support
polygon under
all operating
conditions
(diving under-
water, climbing
a ramp on
land, etc.).

CENTER OF GRAVITY
STILL OVER
SUPPORT POLYGON

CENTER OF GRAVITY
OVER SUPPORT POLYGON

Stability Example No. 1

This robot was built with a heavy weight high above the robot's base. While standing still on a flat surface, it is stable because the center of gravity is above the support polygon. But when the robot climbs a ramp, as shown on the previous page, the center of gravity is no longer over the support polygon, and the robot tips.

WEIGHT REPRESENTS
BULK OF GRIPPER OR
SIMILAR APPARATUS

CENTER OF GRAVITY IS
HIGHER BECAUSE MORE
WEIGHT IS MOUNTED HIGH UP

1kg

CENTER OF GRAVITY IS
LOWER BECAUSE
WEIGHT IS LOWER

1kg
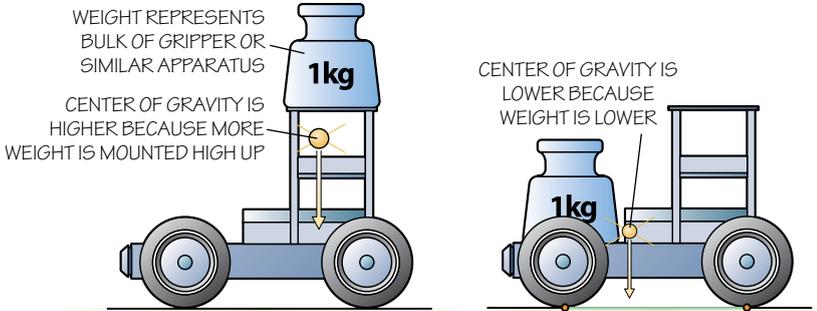
The top-heavy robot can be modified so its weight is lower and forward of its center. Then, climbing a ramp won't be such a problem because the center of gravity is always above the support polygon.

Stability Example No. 2

This robot is designed to pick up a heavy object using a claw. By itself, it is a stable design because the center of gravity is over the support polygon. But when it picks up a heavy object, the center of gravity moves so it is not over the support polygon, and the robot falls over.

CENTER OF GRAVITY IS
DIRECTLY OVER SUPPORT
POLYGON (BUT NOT BY MUCH)

HEAVY WEIGHT NEAR FRONT OF ROBOT MOVES
CENTER OF GRAVITY FAR FORWARD

CENTER OF GRAVITY
NO LONGER OVER
SUPPORT POLYGON
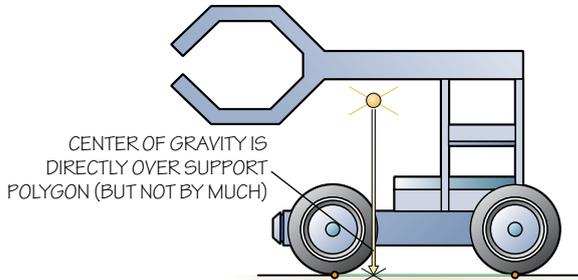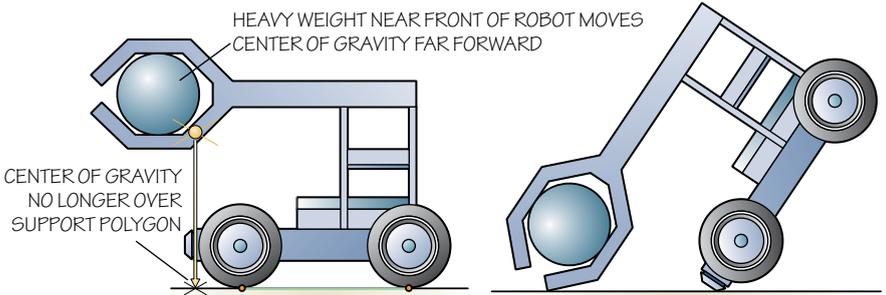
The problem may be solved several ways: (1) move the grabber back so the center of gravity always stays over the support polygon; (2) extend the support polygon by making the chassis longer; or (3) add a counterweight to help keep the center of gravity over the support polygon.

WEIGHT STILL PULLS CENTER
OF GRAVITY FORWARD, BUT
NOT AS FAR AS BEFORE

ADDITIONAL WEIGHT PULLS
CENTER OF GRAVITY BACK
TOWARD CENTER

1kg

CENTER OF GRAVITY REMAINS
OVER SUPPORT POLYGON

CENTER OF GRAVITY REMAINS
OVER SUPPORT POLYGON

CENTER OF GRAVITY OVER NEWLY
EXTENDED SUPPORT POLYGON
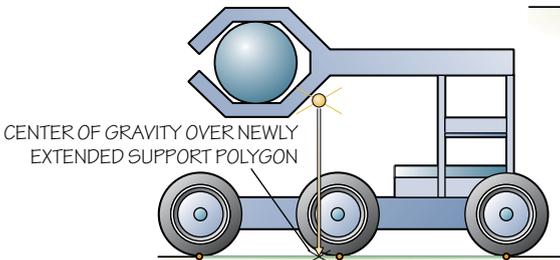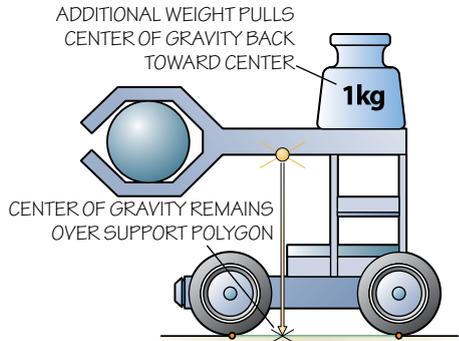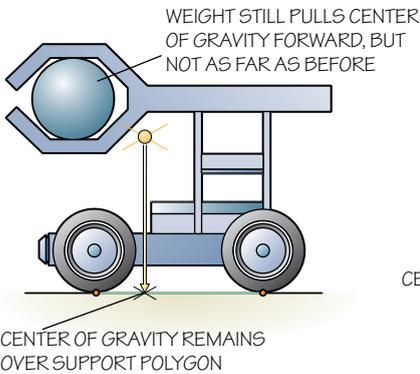
Robust Design

A robust design is one that does not break or fall apart easily. A simple rule to follow to build a robust chassis is the two-point rule: Make sure you have two points of connection on structural pieces.

SIMPLE SCREW

ARM ROTATES TOO EASILY AROUND THIS POINT

TWO SCREWS ELIMINATE ROTATION
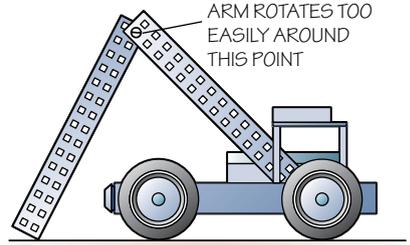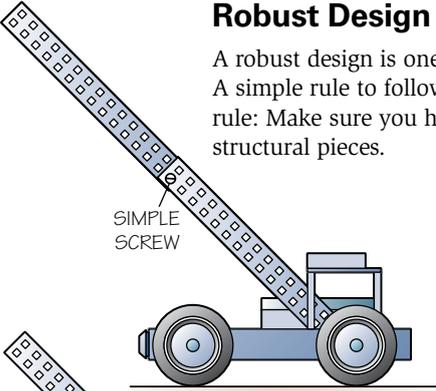
The top piece of the extension is likely to fall because a single screw allows the pieces to pivot.
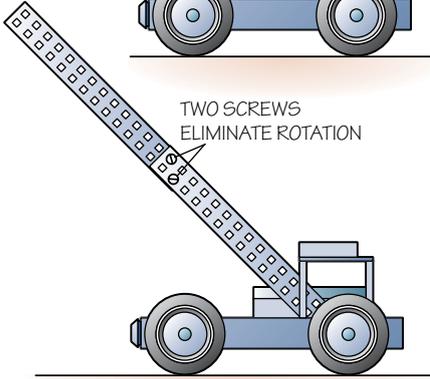
For a more robust design that is less likely to rotate out of position, overlap the metal extension piece with the base piece and use two screws to secure it on both sides.

LONG LEVER ARM CAUSES GREAT STRAIN ON MOUNTING POINT

SINGLE SCREW MOUNTING POINT

SHORTER LEVER ARM CAUSES LESS STRAIN ON MOUNTING SCREW

TWO SUPPORTS

BRACING ARM PROVIDES SUPPORT, SHORTENS LEVER ARM
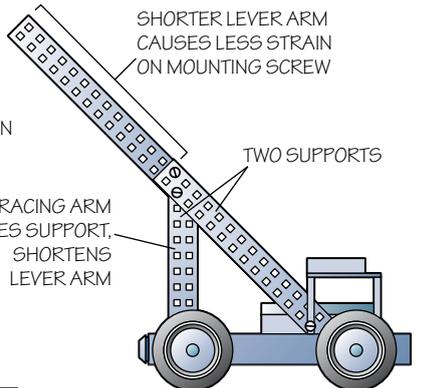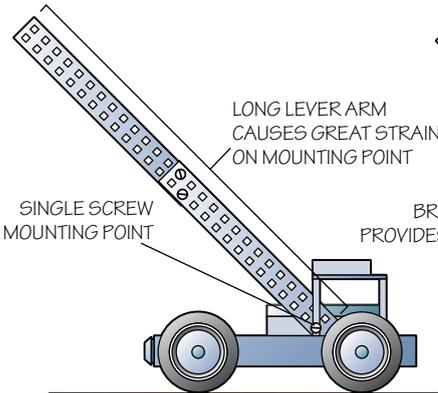
This arm, if not supported, puts stress on the system at the single mounting point. By adding a support, the stress is shared and the arm is more likely to hold.

Vulnerability

Some parts of every robot are more fragile than others. If possible, always plan the chassis design to protect these parts from unwanted physical contact.

MICROCONTROLLER HANGING OUTSIDE ROBOT'S BUMPER IS VULNERABLE TO COLLISIONS

WIRES WILL SNAG ON OBSTACLES OR OTHER ROBOTS

WIRES CAN GET CAUGHT IN ROBOT'S OWN TIRES

MICROCONTROLLER NOW PROTECT FROM COLLISIONS

TIED WIRES ARE LESS PRONE TO SNAGGING.
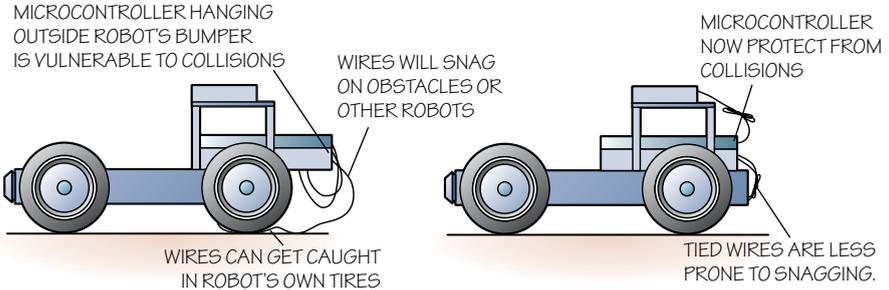
In this example, the onboard computer hangs out beyond the chassis where it can be bumped, and the wiring dangles where it can be snagged and ripped out.

For a more robust design, move the controller inside the chassis and route the wiring so it won't snag.

Safety With Power Tools

Power tools can be dangerous and need to be handled with care. If building the chassis or other parts of your robot requires the use of power tools, talk with your counselor and parent about permission to use the tools and the safety procedures for each.

- Always wear safety glasses when working with power tools, including handheld and table-mounted saws, drill presses, and sanders.

- When using handheld power tools, use a securely mounted vise to hold the material being cut, drilled, or sanded. Use equal caution when working with regular hand tools such as saws and sandpaper.

- Some operations, such as drilling into steel, require oil or a liquid lubricant for a safe and precise cut.

- After a cut is made in a piece of metal or wood, it may be very hot, and the edges may be sharp. Be sure to wear gloves when cutting with power tools.

The *Home Repairs* merit badge pamphlet is a good resource for learning about tools and how to use them safely.

MERIT BADGE SERIES

HOME REPAIRS

BOY SCOUTS OF AMERICA
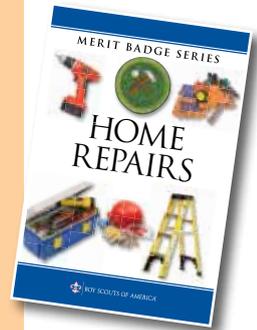
A robot's power system connects to axles, gears, wheels, or other forms of locomotion through the chassis. In many robot designs, the motion and chassis subsystems are tightly integrated.

The Mechanical Subsystem

The mechanical subsystem is the part of the robot that moves the chassis and the manipulators. Like everything else, the mechanical subsystem is attached to the chassis. It's also connected directly to the robot's power subsystem.

The mechanical subsystem manipulates the power system to do many things. This manipulation is called *mechanical advantage*. Forces and energy are at the heart of mechanical advantage.

The many forces on a robot include these:

- *Torque* is the circular force of the motor.
- *Friction* is a force between parts of the robot that may prevent motion.
- *Gravity* is the force that holds the robot on the ground.
- *Lift* is the force that raises a robot off the ground.
- *Buoyancy* is the force pushing things to the surface of water.

Energy is anything that can do work. A simple formula can relate the two concepts: **Force × Distance = Work**

Leverage

Manipulating this formula is what the mechanical subsystem does. Levers are perhaps the simplest parts of a mechanical subsystem. A seesaw is a simple lever. If you have played on a seesaw, you understand that you can lift a much heavier person by moving to the end of the seesaw. You are getting mechanical advantage by putting your weight (the force of gravity on your body) at the end (farthest distance) of the seesaw lever.

CENTER OF MASS

FULCRUM
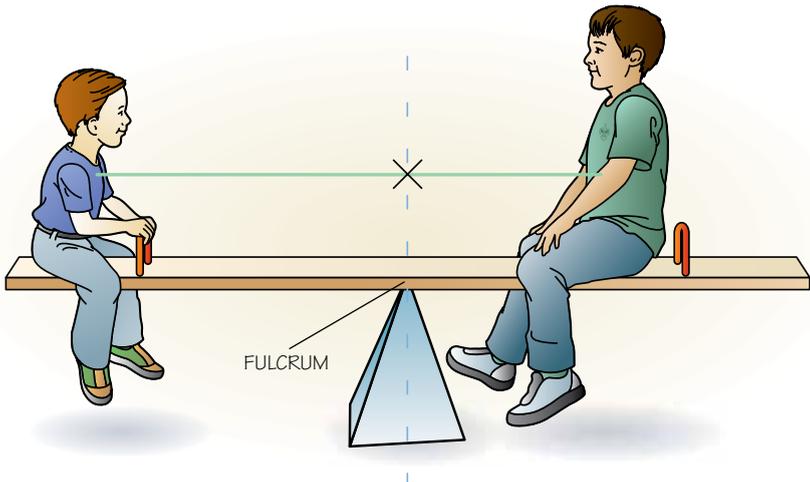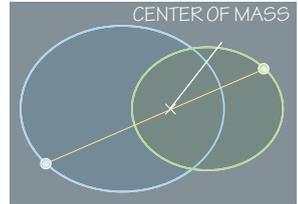
Leverage plays a role in many parts of a robot. A wheel that is a long distance from the chassis (on a long lever), like a hot rod, makes a robot travel in a straight line. This could make it hard to steer. Manipulator arms (like a lever arm) should not be too long or they may break off the chassis.

Torque is force at a distance. You can feel the difference in force at a distance—torque—of an object with your arm. Try this: Hold a weight by your side with your elbow bent. Now stretch out that arm from your body. The weight should seem heavier and harder to hold up because it is further from your body.

Power

A robot's motors and actuators can produce a maximum amount of force and speed for locomotion or manipulation. Force and speed make up the power to the robot. The formula for power is the same as the work formula divided by time:

$$\text{FORCE} \times \frac{\text{DISTANCE}}{\text{TIME}} = \frac{\text{WORK}}{\text{TIME}} = \text{POWER}$$
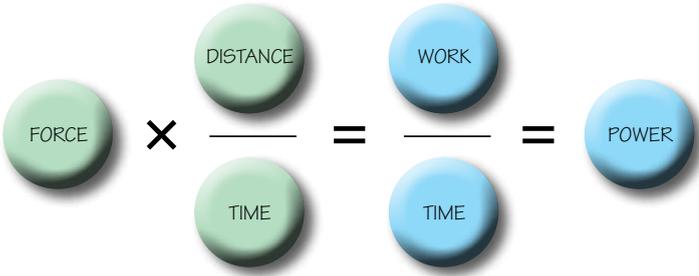
> Notice that "distance over time," or distance divided by time, is *speed.* In phrases such as "miles per hour" or "feet per minute," you can think of the word "per" as meaning "divided by." For example, a distance of 80 miles divided by a time of 2 hours equals a speed of 40 miles per hour.

The *power* of a robot generally does not change. The power delivered by a motor does not change until the energy available from a battery decreases and is discharged. Batteries will weaken as they drain, but that does not make a big difference for the power of the motor. The force and speed *can* be changed, however, and manipulated using levers and gears.

If the power stays the same, force and speed are inversely proportional. *Inversely proportional* means that as force increases, speed decreases: They move in opposite directions. Gears are changing the nature of the energy transmitted and obey conservation of energy principles. You can prove this to yourself by simply plugging in numbers:

> IF FORCE IS 4 AND SPEED IS 3, THEN POWER IS 12.
> FORCE × SPEED = POWER; 4 × 3 = 12.

Now decrease speed to 2. Power stays the same (12) because it is the same motor. The force must go up to 6 to make the equation true again: 6 × 2 = 12. Therefore, as the speed *decreases* from 3 to 2, the force must *increase* from 4 to 6. This holds true for any robot. To increase force, simply reduce the speed with gears.

Gears
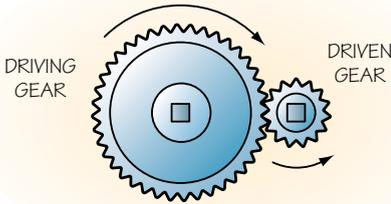
Gears and levers can create more force or speed. The motor on a robot is connected to the gears in the *transmission*. The transmission connects the motors to the wheels through the axles.

To make the robot move faster than the motor, connect a large gear to the motor. Connect this driving gear to a smaller gear—the *driven gear*—that is connected to the robot's wheel with another axle. This will make the wheel travel faster than the motor.

How fast the robot will go is determined by the *gear ratio*. Calculating the gear ratio is simple. First, identify which gear is the driving gear and which gear is the driven gear. Then count the number of teeth on the driving gear, and divide it by the number of teeth on the driven gear.

For example: If the large gear connected to the motor (the driving gear) has 40 teeth and the small gear (the driven gear) that is connected to the wheel has 8 teeth, every time the driving gear (40 teeth) makes one rotation, the driven gear (8 teeth) will make five rotations (40 ÷ 8 = 5). The wheel will travel five times as fast as the motor. This is a 5-to-1 gear ratio.

The cost of this added speed is a loss of force. In this case, the force is the circular force of the motor (torque). The wheel will have one-fifth the torque of the motor.

The opposite is true as well: Using gears, your robot can exert more force or torque than the motor can exert on its own. In the previous example, reverse the driving and driven gears. Connect the small (8-tooth) gear to the motor and the large (40-tooth) gear to the wheel. The small gear is now the driving gear and the large gear is now the driven gear. This gives a 1-to-5 gear ratio: 8 ÷ 40 = 1/5. The wheel will go one-fifth the speed of the motor, but the wheel will have five times as much torque as the motor.

The 24-tooth gear in this string is called an *idler gear*. Idler gears do not improve the mechanical advantage. However, idler gears do serve a purpose in robot designs. You may have noticed that as one gear turns clockwise, the gear it is connected to will turn counterclockwise. An idler gear will make the first and third gear in a string of gears spin in the same direction. Some walker robots use this arrangement of gears.
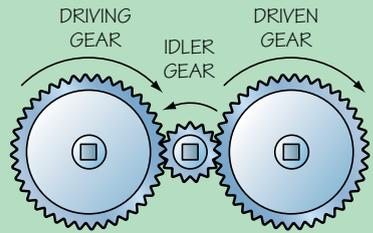
String of Gears

When working with gears, a common mistake is that more gears improve mechanical advantage. Stringing several gears together in a row does *not* increase a robot's mechanical advantage.

This picture, for example, shows three gears. The 40-tooth gear is connected to the 24-tooth gear, which is connected to the 8-tooth gear.
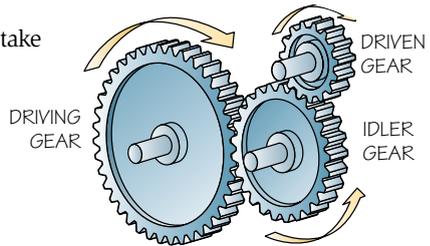
In the first pair, the 40-tooth gear is the driving gear for the 24-tooth driven gear. This gear ratio is multiplied by the gear ratio of the second pair, in which the 24-tooth gear is driving the 8-tooth driven gear. The result is:

$$40 \div 24 \times 24 \div 8$$
$$\text{WHICH EQUALS } 40 \div 8 = 5$$

This is a 5-to-1 gear ratio. (Notice how the 24s cancel each other.) This is no better than the 40-tooth driving gear connected directly to the 8-tooth driven gear.
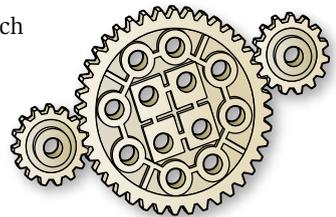
Compound gear

Gear Train

The way to put gears together to multiply mechanical advantage is in a *gear train*. Gear trains are formed when you have more than one gear on an axle.

DRIVEN GEAR

DRIVING GEAR

DRIVEN GEAR
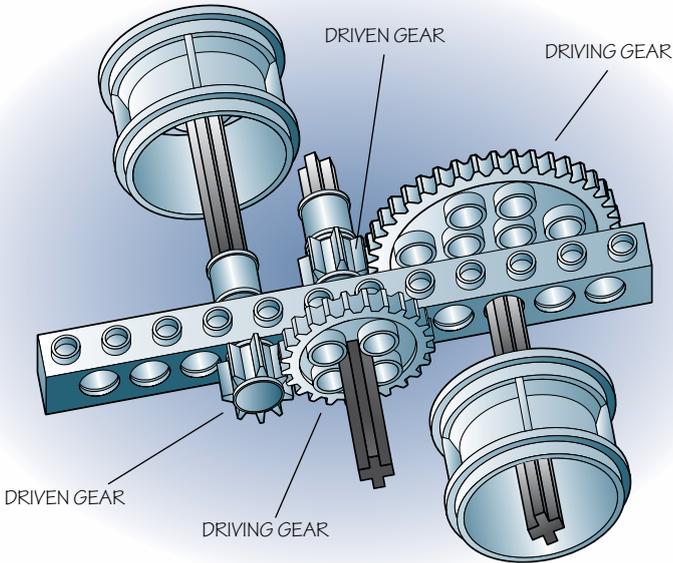
DRIVING GEAR

In this example, a 40-tooth gear is the driving gear. It is connected to an 8-tooth driven gear. On the driven gear's axle is a 24-tooth gear. That 24-tooth gear is a driving gear for another 8-tooth driven gear that is connected to a wheel.

Each mesh of two gears together is a ratio. The first mesh is the 40-tooth driving gear to the 8-tooth driven gear. The second mesh is from the 24-tooth driving gear to its 8-tooth driven gear.

> To get the full gear ratio from the motor to the wheel, multiply the first ratio (40 ÷ 8 = 5) by the second ratio (24 ÷ 8 = 3). The result is 5 × 3 = 15.

This gear train has a 15-to-1 ratio. In other words, the wheel is spinning 15 times faster than the motor.

Chains and Sprockets

Instead of directly meshing with each other, gears can be connected by chains. A gear using a chain connection is called a sprocket. Chains connect the sprockets on bicycles at the pedals and the rear wheel.

For the most part, sprockets and chains work the same way as meshed gears. A 40-tooth sprocket connected by a chain to an 8-tooth sprocket is still a 5-to-1 gear ratio. The main differences are (a) the chain separates the gears by a distance, and (b) the sprockets always rotate in the same direction. Recall that meshed (interlocked) gears turn in opposite directions to each other.

While chains are convenient, they require lubrication and proper tensioning to work correctly. Too much slack in a chain, and it will fall off the sprocket. Too much tension, and the chain will break or burn out the bearings on the sprockets.

Chain drive

DRIVING SPROCKET

CHAIN

DRIVEN SPROCKET
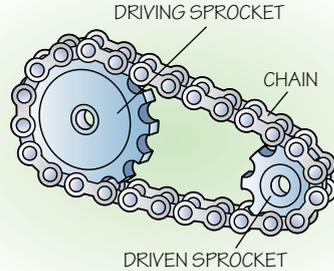
Robot designers find many good applications of gears using chains and sprockets.

Belts and Pulleys

Belts and pulleys are another way to transfer motion over a distance. A *pulley* is a wheel with a smooth rim that has a belt wrapped around it. A pulley has no teeth. So how do you determine the gear ratio?

One way is to turn one pulley one rotation and see how many times the other pulley rotates. If a single rotation of the first pulley makes the other pulley rotate five times, the ratio is 5-to-1. If the rotations do not come out evenly, an easy way to determine the gear ratio is to measure the diameters of both pulleys and take the ratio of the two. The ratio of the diameters will be the same as the ratio of the teeth (if the pulley had teeth).


DRIVING PULLEY

BELT

DRIVEN PULLEY
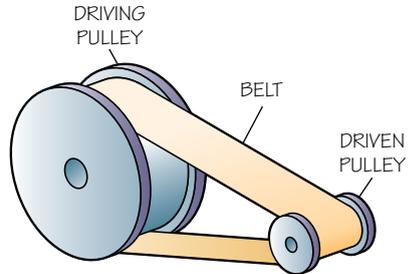
Robot with belts and pulleys
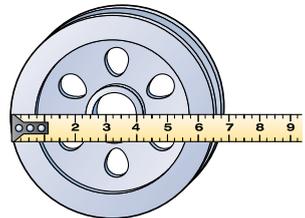
Measuring the diameter of a pulley

Belts have the same issue with tension that chains do—the tension must be just right. Belts and pulleys can also be limiting, because belts usually come in standard lengths that may not be what you want to use. With chains, you can always adjust the length by removing or adding links. Belts tend to be quieter than chains, however. What you decide to use depends on what trade-offs you are willing to make in your design.

Degrees of Freedom

A human arm has seven degrees of freedom: shoulder pitch, shoulder yaw, shoulder roll, elbow pitch, wrist pitch, wrist yaw, and wrist roll. See whether you can identify all the degrees of freedom in your hand. Hint: It's a bunch!

"Degrees of freedom" describes how many directions an actuator or a manipulator can move. For example, a door hinge has one degree of freedom—it can rotate in only one direction.

An airplane has three degrees of freedom: roll, pitch, and yaw. To understand these movements, imagine three lines running through the airplane and intersecting at right angles at the airplane's center of gravity. Rotation around the front-to-back axis is called *roll.* Rotation around the side-to-side axis is called *pitch;* it's an up-and-down movement. Rotation around the vertical axis is called *yaw;* it's a side-to-side movement.

YAW

ROLL

PITCH
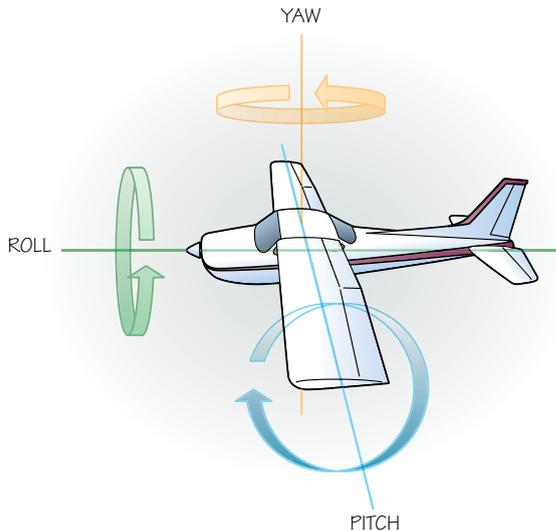
Safety With Mechanical Energy

Be aware that mechanical energy can make robot parts move with great force and speed.

- Springs can provide tension and release a large amount of force quickly. Be sure springs are released before working on mechanisms.

- A large raised arm can fall unexpectedly. Before working on it, be sure it is in a locked position.

- A pneumatic (air pressure) system can release large amounts of energy quickly. Be sure all air lines are purged (emptied) before working on them.

- Loose clothes, long hair, hanging jewelry, neckerchiefs, or probing hands can easily be caught in mechanisms. Wear appropriate clothing and appropriate protective equipment (such as safety goggles) when working on robotic mechanisms.

PINCH POINTS
IDLER GEAR
DRIVING GEAR
DRIVEN GEAR

DRIVING SPROCKET
CHAIN
PINCH POINTS
DRIVEN SPROCKET

PINCH POINTS
SIMPLE SCREW

ARM ROTATES TOO EASILY AROUND THIS PINCH POINT
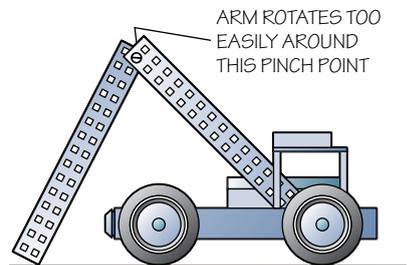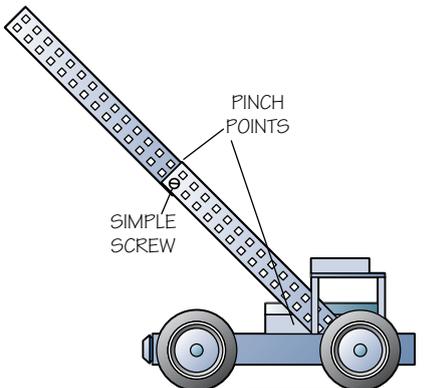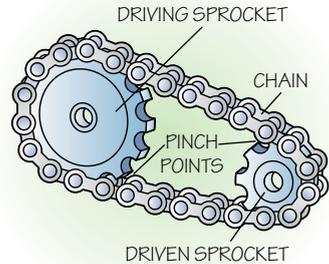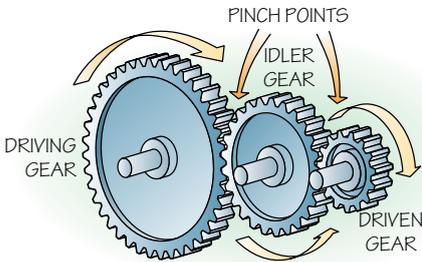
The Power Subsystem

Robots need energy to move, to process information, and to sense the environment. This energy comes from the power subsystem.

Be careful of any energy source. Robots use many types of energy: electricity, air pressure, fluid pressure, springs, rubber bands, heat, and the potential energy of heavy parts hanging in the air. Any of these can hurt you. Always take precautions.
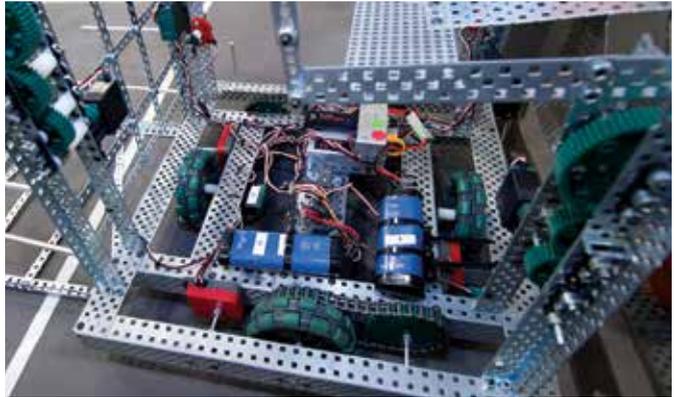
The power subsystem works with the other robot subsystems. It powers the motion subsystem and the controls subsystem.

Electricity

All robots use electrical power, because every robot has a microcontroller (onboard computer) in the controls subsystem. Most robots use electrical energy for other power needs as well, such as moving servos or motors in the motion subsystem.

The power subsystem often adds weight to a robot. The placement of that weight on the chassis affects the robot's stability. And the more weight, the more energy is needed to move the robot. The trade-off between more power versus more weight is a critical design decision.

Electricity can be either alternating current (AC) or direct current (DC). AC electricity in your home and in factories generally comes from wall sockets (and ultimately, from an electric power plant that spins generators to create the electricity). Industrial robots that stay in one place while working on assembly lines use AC current for power. Underwater robots are generally tethered to a ship with a power line—a long power cord that is connected to an AC generator aboard the ship. The power cord adds little weight to the robot and is a good solution to the weight-versus-power problem.

VEHICLE THRUSTERS
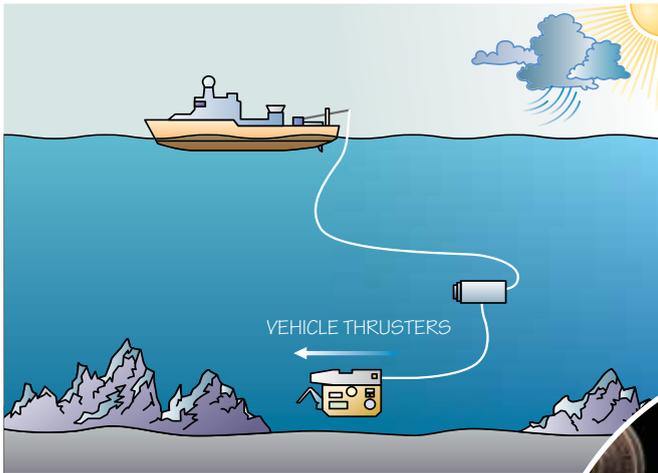
Batteries

Robots that move around, far from a wall socket, usually need to carry their own power in the form of batteries. DC power comes from batteries.

Many different types of batteries are used on robots. Each type has benefits and drawbacks. Flashlight batteries like AAA, AA, C, D-type, or 9-volt dry-cell batteries are inexpensive and easy to find. These batteries are useful for small hobby robots that do not require much power to move. However, if the robot is heavy or gets used a lot, the cost of replacing these nonrechargeable batteries can add up quickly.


Batteries often add significant weight to a robot.

Robots often use rechargeable batteries. These come in all shapes and sizes and are used in many small electronic devices like mobile phones, cameras, camcorders, and portable music players. Types include alkaline, NiCad, lead-acid, and lithium batteries. These batteries tend to be more expensive than nonrechargeable batteries, but they last longer.

Lithium batteries have the most energy and last the longest, but they can get very hot under normal use and can explode if short-circuited. Handle these with care.

A common problem with rechargeable batteries is that the robot owner forgets to recharge them. Some robots have their own recharging system, gathering energy from the environment to renew the battery. A NASA Mars rover, for example, cannot be plugged into a wall socket to recharge. The rover uses solar panels to gather energy from the sun to recharge the batteries that keep it moving and keep its computer working. However, the solar panels add weight to the robot.
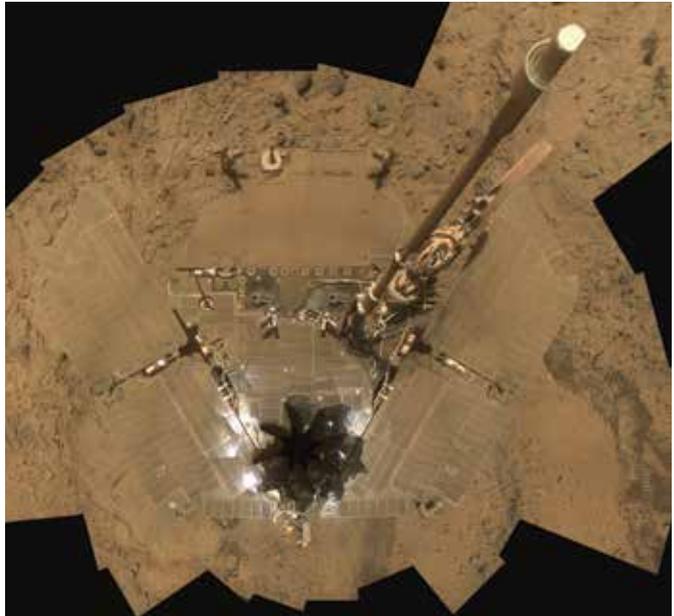
Dust blanketed the solar panels of the Mars rover Spirit so that it blended in with its background. Luckily, the winds on Mars gave the panels an occasional "cleaning," which increased their effectiveness.

Problems With Overheating

Electricity can move motors, but motors are not always the best actuators to move things. When motors are under a heavy load, forced to work hard, they will often get hot. They get hot because the electric energy is not being converted efficiently into motion energy. When electric energy cannot be converted into motion energy, it converts to heat instead.

If a motor is stopped completely, the motor is *stalled*. If a robot's motors are stalled, all the electric energy is converted into heat energy and the motor will heat up fast—a very bad state for a motor. If it heats up too much, the insulation between the wire coils in the motor will melt. A motor may smoke when it is overheated.

By the time you see smoke, the motor is likely to be destroyed beyond repair. When the insulation between the wires melts and smokes, the wire that was protected by the insulation touches the other wires in the motor, and a short occurs. The only way to fix a "smoked" motor is by opening the motor case and rewinding the motor with new insulated wire.

Safety With Electricity

Your robot will probably run on electricity. Take proper precautions:

- Almost everyone is familiar with dry-cell batteries in sizes from AAA to D. These dry cells are *not* designed to be recharged. Dispose of depleted batteries safely at a toxic-waste pickup station.

- Equivalent rechargeable batteries such as nickel-metal hydride (NiMH) will be clearly marked as rechargeable. To recharge these batteries, use only an approved and matched charger.

- Sealed gel batteries provide even more energy to robots. Gel batteries are also designed to be rechargeable. Again, be sure to use a matching charger.

Never short-circuit a battery, and especially not a gel battery. Short-circuiting occurs when the two battery terminals are connected directly to each other with a wire and no resistance between. Batteries contain a large amount of energy, and a short circuit will release all the energy at once. This can cause fires and explosions, and could release toxic chemicals and cause burns.

Soldering

To permanently connect wires, you can solder them together or use wiring terminals. A soldering iron needs to melt solder to join wires. Solder is a mixture of lead and tin, plus a material called flux that helps the solder flow when melted. Burning the flux creates chemical vapors that can be harmful, so be careful to solder only in a well-ventilated area.
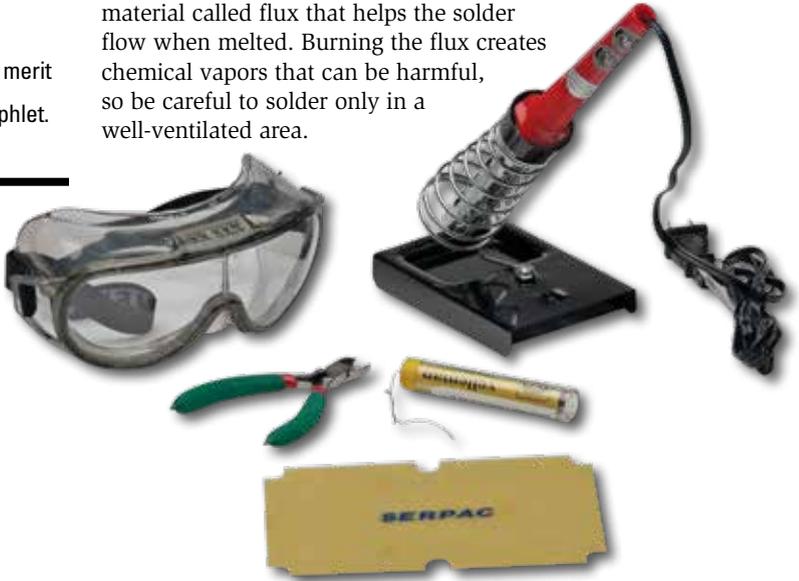
A soldering iron can reach temperatures of 600 degrees F or more. That's much hotter than boiling water (212 degrees F) and can easily burn you.

Soldering Safety Tips

- Solder only under the constant supervision of a responsible and knowledgeable adult.
- Always work in a well-ventilated area.
- Do not allow anyone to drink, eat, or smoke in the soldering area.
- Wear safety goggles.
- Before you start soldering, protect any flammable material with a fireproof shield or wet rags, and have a fire extinguisher nearby.
- Be careful not to let hot solder splash; it will burn you almost instantly.

- If possible, use a soldering-iron stand or clamps when you are soldering, leaving one hand free to hold the solder.

- Do not overheat components, circuit boards, or plastic wires when soldering.

- Oxidation (rust) develops more rapidly when a soldering iron is hot, so try to make sure the iron does not stay hot for long periods.

- Never touch the tip of a hot soldering iron. Keep the iron in a holder when you aren't using it. Don't lay it down on your bench or work area.

- Never leave a soldering iron on unattended.

- During cleanup, have an adult clean oxidation of the soldering iron.

- When you are finished soldering and cleaning up, thoroughly wash your hands.

Pneumatics and Hydraulics

In some robots, pneumatics and hydraulics can be better alternatives than electric motors.

- Pneumatic devices get their energy from compressed air, usually stored in a tank called a reservoir.

- Hydraulic pressure is created with fluids (water, motor oil, brake fluid, transmission fluid, etc.). In a hydraulic system, when pressure is applied at one spot, the pressure is transmitted through the fluid, which causes something else to move. For example, car brakes are hydraulic. When the driver pushes on the brake pedal, the brake fluid is compressed and pushes the brake pads against the wheel discs.

These energy sources are especially good when a powerful, fast force is needed by an actuator. This may be the case if the robot needs to punch through something or shoot something out a tube. Recall that motors lose force as they gain speed.

Hydraulics are great for gripping, holding, and maintaining a position with force.

Also, pneumatics and hydraulics are good for grabbing things and holding them in one place. Hydraulics are used on car brakes for this purpose. They can hold brake pads with lots of force and not be damaged. A motor would quickly burn up if used as a brake because a brake does not move, and a motor that is not moving burns up. Gripping an object may cause a motor to stall and eventually overheat.

The drawback of pneumatics and hydraulics is the added weight. These systems require electronically controlled valves that run on batteries, and a storage tank to hold the compressed air or fluid.

Pneumatic and hydraulic pistons operate similarly. When air or a fluid is forced into one end of the cylinder, it pushes the disk inside the cylinder to the other end, which pushes a rod that is connected to the disk.

Pistons

A pneumatic cylinder has an air valve that allows air to flow into the cylinder and push on a piston, forcing the piston to the other end of the cylinder. Pictured are some typical, small, robotics pneumatic pistons.
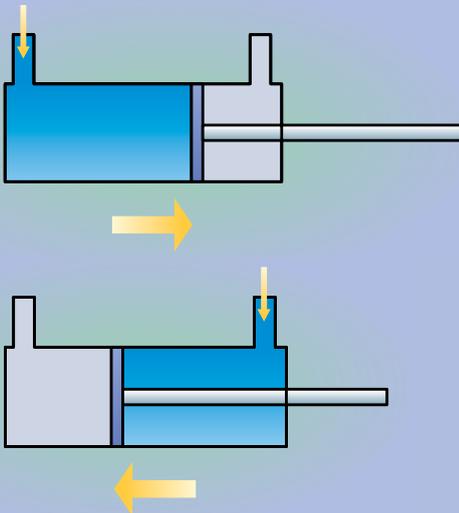

A double-acting pneumatic piston requires more complicated valves and controls than a single-acting type—air pushes from both ends. In the single-acting pneumatic piston, air enters from one end only, and the device uses a spring for return force.

Industrial robots that stay in one place can use pneumatic lines just as they use electric cords. The lines connect to the air tanks and compressors that supply pressurized air. The compressed-air tanks can be in another room; they are not on the robot, and therefore do not add weight.

For mobile robots, however, that must carry all their energy with them, adding pneumatics or hydraulics will add significant weight (which also must be balanced on the robot chassis so it stays stable). When designing your robot, you must decide whether the extra energy from these alternative power systems is worth the added weight.
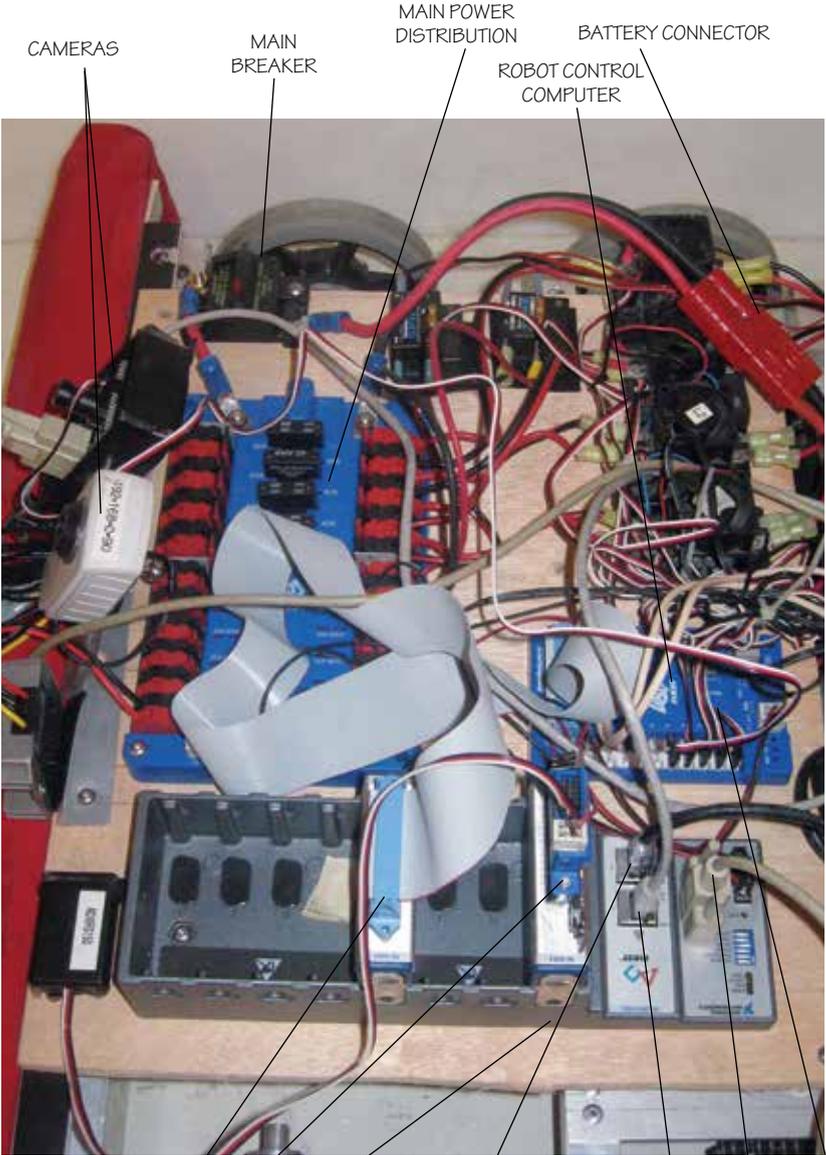
CAMERAS

MAIN BREAKER

MAIN POWER DISTRIBUTION

ROBOT CONTROL COMPUTER

BATTERY CONNECTOR

DIGITAL SENSOR INPUTS

ANALOG SENSOR INPUTS

AIR PRESSURE PUMP

ETHERNET FROM PC CONNECTED TO WIRELESS RECEIVER TO RUN THE ROBOT (BLACK CABLE)

WEBCAM ETHERNET

SECOND CAMERA SERIAL PORT

DIGITAL SIGNAL ROUTER

The Controls Subsystem

The controls subsystem controls what the robot does, similar to the way your brain controls your actions. The subsystem consists of a computer, the microcontroller on the robot, and all of the controller's associated inputs and outputs.
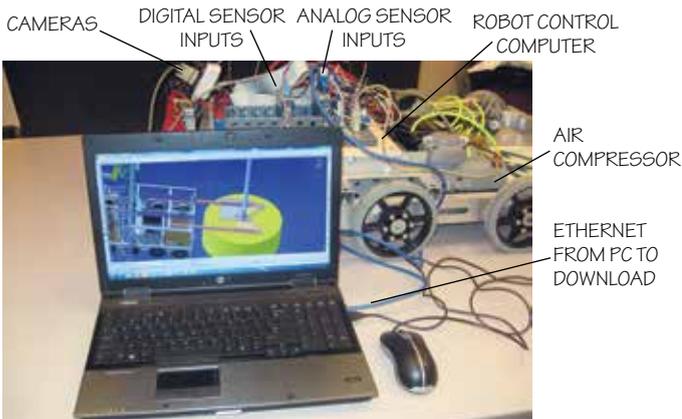
The controller on a robot is like a computer except it does not have a keyboard or mouse.

CAMERAS — DIGITAL SENSOR INPUTS — ANALOG SENSOR INPUTS — ROBOT CONTROL COMPUTER — AIR COMPRESSOR — ETHERNET FROM PC TO DOWNLOAD

The operator uses a computer to write the program that will run the robot. The computer translates the program and sends it to the robot's controller. This screen shows a partial CAD rendering of a robot at a goal.

The robot controller has a large number of input and output ports.

- An *output* is something that gets commands from the controller to do something. Outputs on a robot are the motors, servos, and actuators that control the robot's wheels, flaps, fins, arms, or manipulators.

- The *inputs* are sensors that receive information from the environment and send it into the robot controller. Inputs can also come from an operator—joysticks, lights, etc. These work together to answer the questions the robot must always know: "Where am I? Where am I going? What is in my way?"

Sensors

Sensors interact with the environment in some way that causes them to send a signal to the controller. The controller, using its program software, decides what to do with the signal from the sensors. Sometimes the controller will send an electrical signal to an output, like a motor to move the robot.

Light sensors are also common on robots. A light-sensing robot can be programmed to follow a lit path, hide in the dark, or switch on its headlights when it enters a dim area (down a tunnel or under a bed, for example).

MAIN POWER GROUND

WEBCAM BEING TESTED

MAIN POWER +12V DC

SECOND CAMERA WITH ONBOARD VIDEO PROCESSOR

SERIAL OUTPUT TO ROBOT COMPUTER

ETHERNET OUTPUT TO ROBOT COMPUTER

A bump sensor, for example, signals that the robot has run into something. Adding a bump sensor may be as simple as affixing a small switch to the robot. In a simple bumper sensor, an internal spring holds the switch open until the robot collides with a wall, table leg, or other barrier. Then the switch is pushed in, making an electrical connection and sending an electrical pulse—a signal—to the robot's onboard controller.

How It Works

Here is an example of how the control system works:

The robot controller sends an electrical signal to the outputs to tell the motors to move forward; the motors begin to spin. (Where am I going? Forward.) The robot moves forward until a bump sensor (What is in my way? Something solid.) is pushed by the robot running into a wall. (Where am I? At a wall.)

The sensor sends back to the controller an electric signal that it has been pushed. The controller receives the signal and responds according to the way it was programmed (Where am I going?) by immediately sending another signal to the motors to reverse direction, turn, and move forward again.

The robot repeats this cycle until it reaches its destination or is told to stop by the program.

Bumper switch
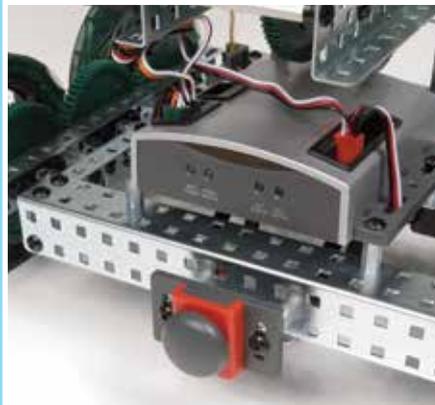
Robot with bumper switch installed

Programming

Microprocessors (computers) are mainly a collection of on or off switches. In machine language, on is represented with a 1, and off is represented by a 0. The only commands a computer can understand are 1's and 0's.

The operator or programmer writes the program in plain language, and it is translated or compiled to the robot in *machine language*. All programming languages must ultimately talk to the computer in machine language, which is the lowest common language for all microprocessors.

Program a Friend

Programs generally follow a specific way of thinking, or logic. Programmers must use the logic to write their programs.

Programming another person is a quick, easy way to learn about programming logic. Try to write a simple program using English words to blindfold a friend, sit at a table, and direct your friend to pick up and stack some blocks or books. Use only simple terms like "move left hand forward," "pick up," "set down," etc. Do *not* say things like "Walk across the room and go out the door."

Doing this activity even briefly will help you understand the first rule of programming.

First general rule of programming: A robot will do exactly what you tell it to do, but not always what you want it to do. (To find out more about programming, earn the Programming merit badge.)

You may ask your friend to turn left, meaning "turn 90 degrees to the left." But the person may turn around in a complete circle to the left. Your friend did as you said, but not what you meant. Precision is important in programming. You will quickly learn that the more precise you are with your instructions, the more likely you will produce the desired result.

Pseudo Code

The thought process of programming a friend is similar to the initial thought processes all programmers go through when they begin to program robots for a task. They break down the larger task into programmable-size chunks. Programmers often create flowcharts of commands that will lead to the final task. Such flowcharts are called pseudo code. You used pseudo code to program your friend.

Pseudo code generally cannot be used to program robots. LEGO® Mindstorms® software and VEX IQ Blocks comes close, however. They are a graphical software package that allows you to combine a series of software icons or pictures to program. The LEGO® Mindstorms® is built on LabVIEW software that is also icon-based. LabVIEW is powerful software that many engineers use for highly complex robots like NASA spacecraft. Mindstorms® is a first step to learning LabVIEW.

High-Level Languages

Programming directly in machine language, the most basic level (just 0s and 1s), can be done, but it is tedious and slow. Most languages that programmers use instead are *high-level languages*.

The two main categories of high-level robot programming are *text-based* programming and *graphical* programming.

- Text-based programs like Java, C, and C++ use English to program the robot's computer. The operator types commands on a keyboard. The English must be translated into machine language using another special piece of software called a compiler. Once the robot receives the commands in machine language, it can then execute (do) them.

- Using a graphical language instead of typing text, the operator drags and drops graphics blocks onto the screen and connects them with lines indicating the flow of the program. These are then compiled and executed.

Text-Based Programming

Text-based programming takes skill to translate pseudo code into programming that a robot can understand. To write programming code in languages such as Java, C, and C + + you need to learn the *syntax* (rules or patterns) of the language. Text-based languages have statements, and statements within statements. The full syntax of these languages is beyond the scope of this pamphlet. You will have to learn some of this syntax, however, to understand your robot program. Review the manufacturer's instructions for the kit you plan to use, and consult with your counselor for guidance.

Sequences of Commands

Programs are either linear or structured. Linear programs go in a sequence from first to last. The first command to the robot is executed first, the last command is executed last, and all the intervening commands are executed in the order they appear.

Structures like loops and "go to," "if/then," and "while" statements cause the commands to move around in the program.

- *Loops* are used to make a segment of code repeat a specific number of times.

- *Go to* statements cause the program to jump to another area in the program.

- *If/then* statements are conditional statements. They usually work with sensors. If the sensor has been activated, then the program will do something. If the sensor has not been activated, then the program does something else.

- *While* statements are also conditional. They will run a part (or subroutine) of the program *while* something is happening. After the "something" ends, this subroutine returns to the rest (or main part) of the program.

"Robot move forward, stop, and return to the starting spot" is an example of linear programming. One command follows the next.

Note that "Return to starting point" is too large a statement. This statement will have to be broken down into smaller chunks that the robot can understand. This statement is not sufficient even for pseudo code.

Warehouse Robot Pseudo Code

Try writing pseudo code for a warehouse robot.

From point A, left wheel forward;

Right wheel forward;

If light sensor sees dark line,
turn right wheel forward,
left wheel stop;

If light sensor sees white, turn left
wheel forward, right wheel stop;

Loop while only one light sensor
is dark or light;

If both light sensors see dark line,
stop wheels;

Lift manipulator;

Push manipulator forward.

Drop part in bin;

Pull manipulator backward.

Lower manipulator;

Return to point A.
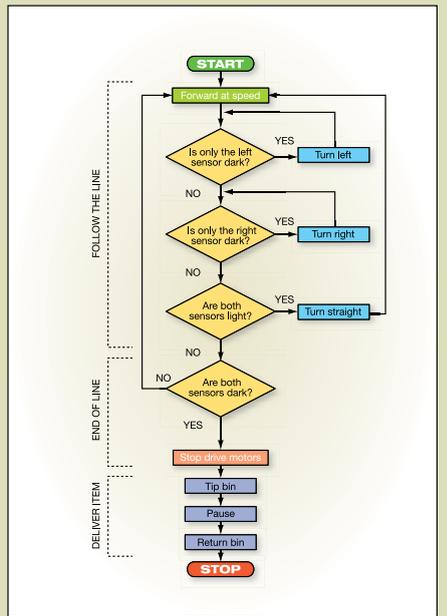
This pseudo code includes *if/then* and *loop* statements. Following the line to the warehouse bin required the structured statements. To follow a line, the robot must find the edge of the dark guiding line with its light sensor. Following the line requires the robot to zigzag along the edge of the line, first moving into the line and then moving away from the line, toward the dark, then toward the light.

The end of the line is marked with a long line across the top of the guide line. This line is wide enough that both of the robot's light sensors will see it at the same time. This is the signal to the robot that it is at its destination. It now jumps out of the loop and on to the rest of the program. The dropping of the part in the bin is linear programming.

The process of writing pseudo code helps the programmer decide on the sequence of events in the program. Once established, programming icons or proper programming syntax can be used to assemble the programming code into the robot program.

Troubleshooting

Writing a good program with no errors the first time is nearly impossible. After writing some code, programmers will usually troubleshoot it for problems and errors. To troubleshoot the code, the programmer compiles the code to the robot. If the compiler has trouble translating the programming code, it will send back an error message to the programmer. At this stage, errors are common.

Programmers should constantly test their code to see that it works error-free. The errors that appear when troubleshooting large amounts of code can be confusing, if not impossible to understand well enough to fix. Take the advice of experienced programmers: If you are going to write programs, get into the habit of checking your code every few lines. That way, when an error message appears while troubleshooting, you will know the problem is only a few lines away.

> Second general rule of programming: Never write a lot of code before troubleshooting it.

Comments in the Code

Talking to a machine is not like talking to a person. The purpose of several lines of code in a program may not be obvious to others who are trying to interpret the code. Every programming language allows the programmer to embed comments into the code. These comments are the programmer's explanation for the code that follows. Large programs may become quite complicated because the code may jump around a great deal as it is processed by the robot controller. The larger the program, the more important it is to embed comments in the code to explain what is happening.

Experienced programmers always embed many comments into their programs so that others who would like to use or modify the code can do so more easily. Comments also allow programmers to attribute an elegant, clean piece of code to its creator. Programmers can either take credit for the code they have written or give the credit to someone else who did a good job. Giving and receiving credit for doing something well is always a good thing.

> Third rule of programming: Comment your code as much as possible.

> If you do not program the robot yourself, you will be expected to thoroughly understand how the program interacts with the sensors and actuators to make the system function. You can create a simple flowchart or pseudo code to demonstrate your knowledge.

```
int motorLeftInput1 = 0 ;    // Motor1 inputs connected t
int motorLeftInput2 = 1;
int motorLeftPwm= 3;         // Motor1 PWM pin proces
int motorLeftEnable= 2;

int motorRightInput1 = 8 ;   // Motor1 inputs connect
int motorRightInput2 = 9;
int motorRightPwm= 11;       // Motor1 PWM pin pr
int motorRighttEnable= 10;

//Ultrsonic sensor declare
#define trigPinL 5 //Ping processor specific assignm
#define echoPinL 4 //Receive processor specific as

#define trigPinR 7 //Ping processor specific assign
#define echoPinR 6 //Receive processor specific a

int distanceL; //Storing current Sonar value for LE
int distanceR; //Storing current Sonar value for R

int distance_thresh = 60; // desired distance to ob

int speedL; //sending moving value to LEFT mo
int speedR; //sending moving value to RIGHT

int echo_return;

void setup()    //associate variables to the ha
{
    pinMode(motorLeftInput1, OUTPUT);
    pinMode(motorLeftInput2, OUTPUT);
    pinMode(motorLeftEnable,OUTPUT);
    digitalWrite(motorLeftEnable, HIGH);
    digitalWrite(motorLeftInput1 , LOW);  //
    digitalWrite(motorLeftInput2,HIGH);

    pinMode(motorRightInput1, OUTPUT);
    pinMode(motorRightInput2, OUTPUT);
    pinMode(motorRightEnable,OUTPUT
    digitalWrite(motorRighttEnable,HIGH
    digitalWrite(motorRightInput2, LOW);
    digitalWrite(motorRightInput1, HIGH
```

Robotics Competitions

Youth robotics competitions range from local competitions for 30 students from a single town to major, organized tournaments featuring thousands of students, corporate sponsors, and teams from all over the world. Only a small sample is described here. The "Resources for Kits and Competitions" section, below, will help you get started finding the right robotics tournament for you or your team.

When using the internet, you should do so only with your parent's permission and knowledge.

Some Popular Competitions for Youth

AUVSI Foundation. The Association for Unmanned Vehicle Systems International Foundation conducts robotic competitions for high school and college students. From aircraft and ground to aquatic vehicles, the AUVSI Foundation competitions focus on designing, building, and deploying a fully autonomous robot and competition with students from around the world.

BEST™. Boosting Engineering Science and Technology (BEST™), founded in 1993, runs a school-based, sports-like robotics competition for middle and high school students. For information, search the internet (with your parent's permission) for "BEST robotics."

Botball®. The Botball Educational Robotics Program is an elementary through high school competition that uses time-trial scoring. Tournaments are held throughout the United States. The robots are autonomous only, and the rules allow many types of robotics controllers and construction. The final rounds of each tournament feature a head-to-head autonomous challenge. Search the internet for "Botball."
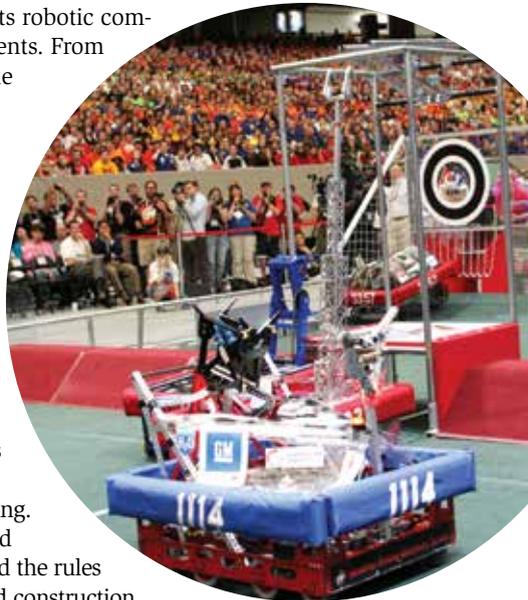
FIRST® competition

FIRST®. For Inspiration and Recognition of Science and Technology (FIRST®) runs both autonomous time-trial and teleoperated sports-like robotics competitions for youth from ages 6 to 18. Younger students use LEGO kits (FIRST® LEGO® League or Jr. FIRST® LEGO® League—FLL®/Jr.FLL™). Older students choose between competitions featuring either a blend of LEGO and metal (FIRST® Tech Challenge—FTC®) or a custom-fabricated 130-pound robot (FIRST® Robotics Competition—FRC®). Thousands of FIRST teams can be found throughout the United States and in many other countries. The competitions welcome school, club, community, and Scout teams. Search the internet for "FIRST robotics."

MATE. The Marine Advanced Technology Education (MATE) Center coordinates an international student ROV (remotely operated vehicle) competition that takes place underwater. Primarily for middle school and high school students, MATE requires sophisticated construction and programming. It is a time-trial competition, with one vehicle trying to complete a series of tasks. Search the internet for "MATE ROV competition."
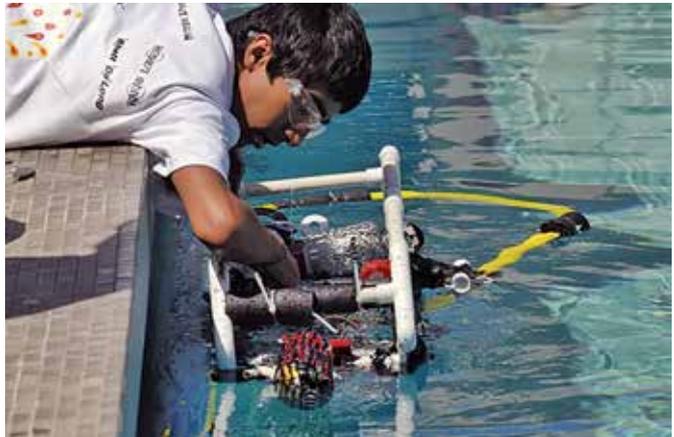
MATE competition

Robofest®. Robofest® is an annual autonomous robotics competition for students in grades five through 12 and college. It challenges teams of students to design, build, and program robots to compete in two age divisions. Robofest robots participate in autonomous challenges using time-trial scoring. School, club, community, and Scout teams are welcome. Search the internet for "Robofest" or "Lawrence Tech robotics."

TSA-VEX® Robotics. The TSA-VEX Robotics Competition is a head-to-head sports-like competition put on by the Technology Student Association. TSA's program uses VEX® robotics parts, and the robots may compete in autonomous, teleoperated, or hybrid challenges. A tournament may be open to TSA school chapters only, or to all teams in an area, so be sure to check with your local TSA chapter. Search the internet for "TSA-VEX robotics."

Trinity College Fire Fighting Home Robot Contest. The Trinity College firefighting contest is among the oldest robotics competitions. It has divisions for younger people, high school students, and adults. Teams build fully autonomous robots that explore a scale-model house and extinguish a fire (a candle). Regional events are held worldwide, and the championship is at Trinity College in Hartford, Connecticut. Search the internet for "Trinity firefighting robot."

VEX® Robotics. The VEX® Robotics Competition (VRC) is an international competition offering separate sports-like divisions for elementary, middle school, high school, and college students. School, club, community, and Scout teams are welcome. Teams participate in hundreds of VRC competitions—autonomous, tele-operated, and hybrid autonomous/teleoperated—in more than 25 countries, followed by a world championship in the United States. Search the internet for "VEX IQ Challenge," "VEX Robotics Competition," or "VEX Robotics College Competition."

Trinity College also sponsors a "RoboWaiter" competition to build a robot that can retrieve a plate of food and transport it reliably to a table.
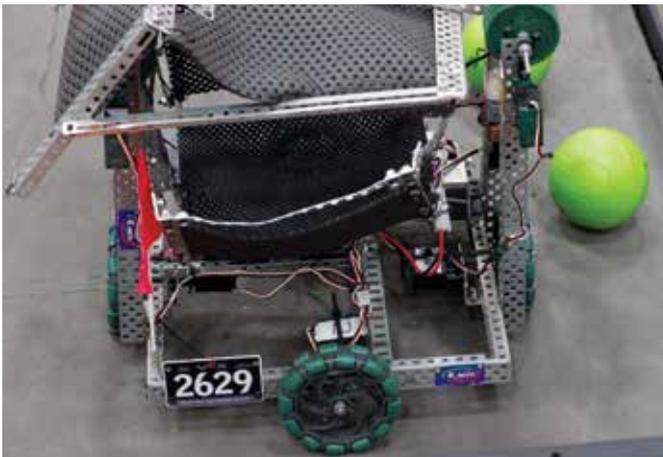
VEX® Robotics competition

Visitors at Roboworld

Resources for Kits and Competitions

Hobbyist and student kits and competitions are a fast-changing part of the world of robots. In place of website and company information, which may soon be obsolete, here are tips for finding information that should provide useful results even as website links and companies change.

Use a Search Engine

Use a search engine (with your parent's permission) to find what you need. When looking for kits or competitions, search "robotics" or the name of a company, competition, or kit. For example, try searching for these terms:

- Robotics kits (returns hundreds of thousands of entries)

- Robotics kits for students

- Robotics kits for kids

- Robotics kits reviews

- LEGO robots

- Parallax robots

- VEX robots

Use an Internet Gateway

Use a portal to find information. You might begin your search at an internet gateway such as these.

Carnegie Mellon Robotics Academy (www.cs2n.org) has information on several popular robotics platforms and youth competitions, and provides many resources.

NASA Robotics Alliance Project (robotics.nasa.gov) provides fun information about robots and is a gateway to several popular youth robotics competitions.

Other useful robotics portals include magazine websites (search for "robot magazine"), robotics clubs (search "robotics club" and "robotics society"), and professional groups such as the Institute of Electrical and Electronics Engineers (IEEE.org).

Robot Events (www.robotevents.com) is maintained by the Robotics Education and Competition Foundation and is a clearinghouse for information on youth robotics competitions.

Robotics Education and Competition Foundation (recf.org) connects students, mentors, and schools with engaging technology-based programs and competitions.

In addition to your counselor, use traditional means. Go to a library or talk to your teachers, Scout leaders, parents, and fellow Scouts. Robots draw widespread interest—you will be surprised by how many people can help you.
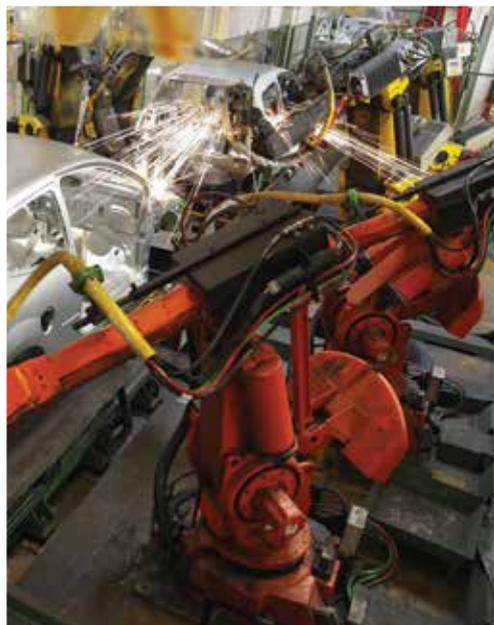
Careers in Robotics

From deep seas to deep space to your closet at home, robots are everywhere. Only a few of the possible career paths are discussed here.

Basic Training

Starting a career in robotics begins right now. Take as many classes as possible in math, physics, computers, and design. Get as much practical, hands-on experience as possible by participating in clubs or organizations pertaining to robotics or engineering.

You will need more than a high school degree to work with robots. If you want to be a robotics technician, many junior or community colleges offer robotics and automation associate degrees. Robotics technicians operate and maintain the robots; if you want to design robots you will need to get an engineering degree.

Most people who seek a career in robotics major in mechanical, electrical, structural, biomedical, or industrial engineering or computer science. These degrees will prepare you to be able to design several components of a robot system. To design a complete robot, you need to know more than one of these fields. Most robotics engineering programs today are graduate-level programs that integrate several engineering disciplines. Nevertheless, there are many opportunities around robots these days.

The skills used for robotics crosses into many industries, such as computer software, automotive, electrical power generation, and many more.

Robotics careers are as varied as the tasks that robots can do. You might choose a specialized discipline, such as programming or mechanisms. What works in one industry may also be applied in another industry, such as bomb-disposal robots in law enforcement and the military, or remotely operated repair robots in a nuclear power facility, space, and deep sea environments.

Exploration

Exploration is a prime field for robotics, to reduce the risk to humans working or living in extreme environments such as the deep ocean or space. Such hazardous environments present ideal conditions to use robots.

Space. NASA needs robotics engineers to design space robots. Space is a hostile environment for humans, but not for robots. In space, one side of a robot, astronaut, or spacecraft may experience temperatures of 200 degrees F on its sunny side and *minus* 200 degrees on its shady side. Space is also a vacuum. If an unprotected human was exposed to the harsh environment of space, it would be fatal. Robots can be designed to handle this harsh environment.

The entertainment industry also offers many possibilities for using robots in movies and amusement parks.

The space shuttle's robotic arm has helped astronauts perform countless tasks safely inside the space shuttle or International Space Station. Rovers such as those sent to Mars have helped people explore the Martian surface without risk to human life. The robot does not mind being stuck in a small capsule for the six-month journey to Mars. The robot does not mind the stifling cold. And, most of all, it does not mind being left on Mars once its mission is complete.
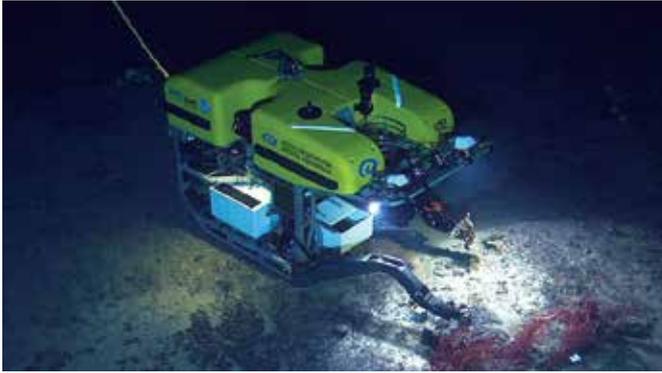
Robots help
us learn about
new environments
before we
send people to
those places.

Deep Sea. The National Oceanographic and Atmospheric Agency uses autonomous underwater vehicles, or AUV robots, to explore the oceans and atmosphere. Earth's ocean floors are largely unexplored by humans because the deep ocean environment is dangerous to humans. Robots are being used to explore the ocean floor so that we can determine what resources are available there and how we can better protect the ocean floor environment. Many private industries—especially petroleum companies—use AUVs to support deep-sea petroleum exploration and drilling. The Deep Water Horizon oil spill was eventually capped by AUVs.

The Military or Law Enforcement

Anywhere safety is a major concern, robots can be found. Teleoperated robots provide air, land, and sea surveillance that helps locate dangers. Bomb squad and fire-control robots allow public safety officers to stand off from dangerous situations like fires, collapsed buildings, or hostage situations. As firefighters and police officers become more familiar with robots, the demand for them will increase.

Robots have revolutionized how soldiers fight wars. There are a growing number of flying, swimming, and rover robots used in the military. The need for robot operators and robot designers in the military is likely to grow quickly in the near future.

Aside from
assisting humans
in discovering the
wreckages such
as the *Titanic* and
salvaging many
other sunken
treasures, AUVs
have also helped
in discovering
new forms of
underwater life.

Medicine

Medical robots are rapidly changing the field of medicine. More and more robots are being designed to support doctors in all aspects of their work. Robots are being used to analyze test data and identify DNA defects much faster than in the past. Lighter materials and more compact mechanisms have greatly improved artificial limbs, giving amputees the opportunity to live a more normal life.

Robots also are being used in the operating rooms. Teleoperated systems give surgeons greater control over their instruments. Surgeons can now do delicate operations on the eye, prostate, and other small organs they only dreamed about operating on a few years ago.

These same robots make it possible for doctors to perform surgeries over long distances. By connecting a surgical robot to the internet, a doctor in Los Angeles can do surgery on a patient in New York. Nanotechnology, with its extremely tiny machines, is transforming medicine by decreasing or eliminating the need to cut into a patient.

"Nanobots" may soon be injected into patients' bloodstreams to administer medicine, repair the body, or fight off infection.
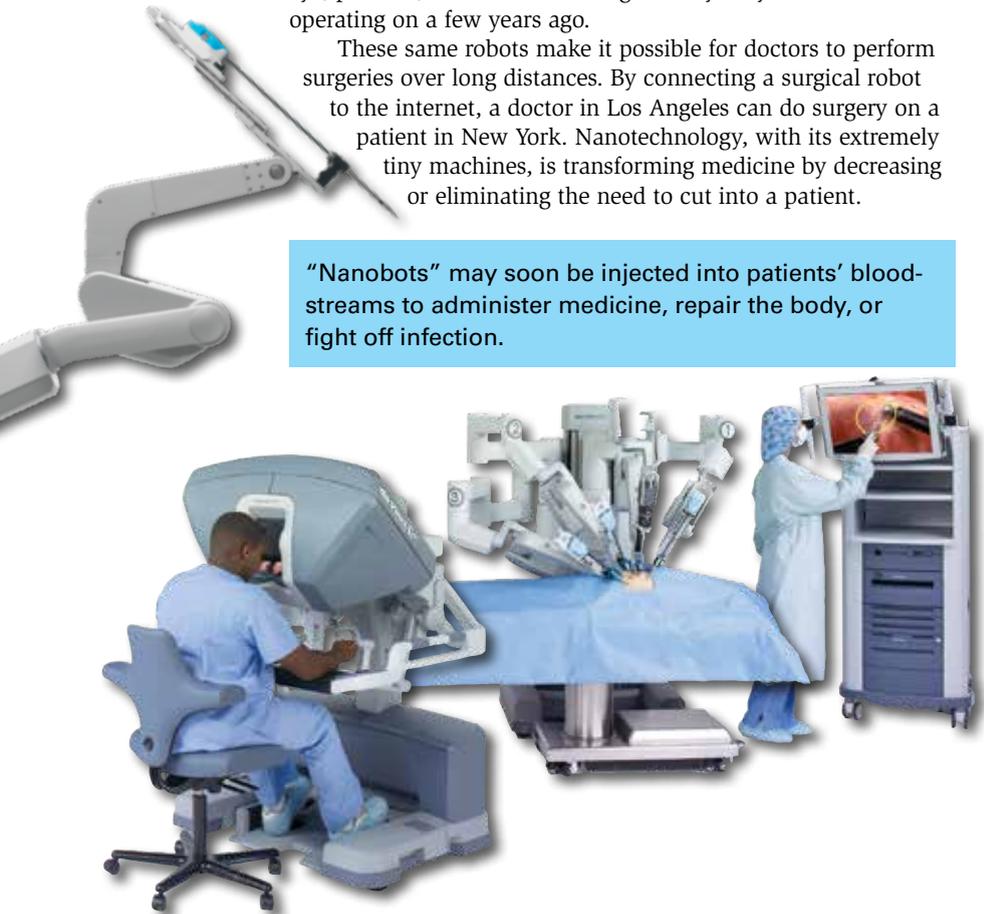
Industry

In industry, robots that work on assembly lines are built and programmed to repeatedly perform specific tasks, resulting in quicker production and more consistent quality. Teleoperated robots are used in construction, mining, and agriculture. Robotic transportation moves hazardous materials around warehouses or manufacturing plants. Obstacle avoidance becomes important in robotic transportation when the risk is high for humans or the job is routine enough to delegate to an autonomous robot, thus leaving the human free to do other things.

Personal Robots

Besides robots for government and business work, there is a growing number of personal robots to do our dirty work like vacuuming the floor or cutting the grass. iRobot and other companies need robotics engineers to develop the next generation of personal robots to make each of our lives easier and more efficient.

All of these robots still require human operators. All of these robots need to be maintained and improved. Many more robots need to be developed to address other needs that we have. There is and will continue to be a great need for robot operators and robot designers for years to come.

Mechatronics systems intelligently combine mechanical and electrical elements with control systems and information technology to perform complex tasks. "Mechatronics" is another way of saying "intelligent mechanical systems." It is an up-and-coming career focus in robotic technology that can be applied in many other fields.

Robotics Resources

Scouting Literature

Composite Materials, Digital Technology, Electricity, Electronics, Engineering, First Aid, Game Design, Home Repairs, Inventing, Metalwork, Model Design and Building, Programming, Safety, Space Exploration, Textile, Welding, and *Woodwork* merit badge pamphlets

With your parent's permission, visit the Boy Scouts of America's official retail website, www.scoutshop.org, for a complete listing of all merit badge pamphlets and other helpful Scouting materials and supplies.

Books

Clark, Dennis, and Michael Owings. *Building Robot Drive Trains*. McGraw-Hill, 2002.

Cook, David. *Robot Building for Beginners*, 2nd ed. Apress, 2010.

Domaine, Helena. *Robotics*. Lerner, 2005.

Gibilisco, Stan. *Concise Encyclopedia of Robotics*. TAB Books, 2002.

Henderson, Harry. *Modern Robotics: Building Versatile Machines*. Chelsea House, 2006.

Mataric, Maja J. *The Robotics Primer*. MIT Press, 2007.

McComb, Gordon. *Constructing Robot Bases*. McGraw-Hill, 2003.

———. *Robot Builder's Bonanza*, 4th ed. McGraw-Hill, 2011.

Payment, Simone. *Robotics Careers: Preparing for the Future*. Rosen, 2011.

Petruzzellis, Thomas. *Electronics Sensors for the Evil Genius*. McGraw-Hill, 2006.

Predko, Myke. *123 Robotics Experiments for the Evil Genius*. McGraw-Hill, 2004.

Randolph, Ryan P. *Robotics*. Rosen, 2009.

Valk, Laurens. *The LEGO® Mindstorms® NXT 2.0 Discovery Book: A Beginner's Guide to Building and Programming Robots*. No Starch Press, 2010.

Williams, Karl. *Build Your Own Humanoid Robots*. McGraw-Hill, 2004.

DVDs

NOVA: The Great Robot Race. PBS, 2006.

Robotics. History Channel, 2001.

Online Resources

"How to Build a Robot." Razor Robotics, www.razorrobotics.com/building-robots

"Robotics Tutorials." ElectronicsTeacher.com, www.electronicsteacher.com/robotics/robotics-tutorial

Also see the websites and sources described under "Robotics Competitions" (See "Resources for Kits and Competitions" on page 76).

The next time you are online or at the bookstore, check out magazines like *Servo, Nuts and Volts,* and *Popular Science.* These publications are great resources for robotics-related information.

Organizations and Websites

**American Society of
   Mechanical Engineers**
3 Park Ave.
New York, NY 10016-5990
Toll-free telephone: 800-843-2763
www.asme.org

**AUVSI Foundation
   Robotic Competitions**
Association for Unmanned Vehicle
   Systems International
www.auvsi.org

Carnegie Mellon Robotics Academy
Computer Science Social Network
www.cmu.edu/roboticsacademy

**Institute of Electrical and
   Electronics Engineers**
2001 L Street, NW, Suite 700
Washington, DC 20036-4910
Phone: 202-785-0017
www.ieee.org

National Robotics Week
www.nationalroboticsweek.org

RoboGames
www.robogames.net

**Robotics Education and
   Competition Foundation (REC)**
recf.org

Technology Student Association
1914 Association Drive
Reston, VA 20191-1540
Toll-free telephone: 888-860-9010
www.tsaweb.org

Get Interactive with *Scout Life* magazine

Unlock the BSA's interactive merit badge resource center! Enhance your merit badge learning experience by tapping into a mind-blowing explosion of videos, games, activities, photos, the latest resources, and much more. Visit **scoutlife.org/merit-badges**.

Acknowledgments

Laszlo Hideg, Ph.D., P.E.—Electrical engineering components, electric motor controls, Chrysler LLC; *Programming* merit badge pamphlet team member; FIRST® Robotics competition judge, volunteer, and mentor

Tarek Shraibati—professor, Department of Manufacturing Systems Engineering and Management, California State University–Northridge

Richard Tyler—manager, Western Region, Robotics Education and Competition Foundation; president, Exothermic Robotics

Special thanks to the Science and Engineering Education Center at the University of Texas at Dallas.(seec.utdallas.edu), a leader in robotics and hands-on education, for providing the services of Dr. Kenneth Berry. Dr. Berry coordinated the development of the requirements and the pamphlet.

We are grateful to Carnegie Mellon Robotics Academy (www.cs2n.org) and Robin Shoop, its director, for their generosity in including the BSA as part of the FIRE initiative (Fostering Innovation through Robotics Exploration). This program helps foster student interest in robotics, computer science, and technology, and helps engage young people in robotics competitions, preparing them "to be future innovators."

Thanks also to the Association for Unmanned Vehicle Systems International (AUVSI), NASA, and the Robotics Education and Competition Foundation (RECF) for their assistance with providing visuals for the pamphlet and their close guidance and support throughout this entire project. The BSA thanks Eagle Scout Christopher Hideg, an electrical engineering student at the University of Detroit–Mercy. He is also a FIRST Robotics mentor and past competitor.

Photo and Illustration Credits

ROBOTICS RESOURCES

NASA/Jet Propulsion Laboratory–Caltech/ Cornell, courtesy—page 56

NASA Johnson Space Center/Joe Bibby, courtesy—page 19 *(Robonaut)*

National Oceanic and Atmospheric Administration, courtesy— page 81 *(AUV)*

©2011 Parallax Inc., courtesy. Use of Boe-Bot and Boe-Bot Robot Kit photos is with permission of Parallax Inc. Boe-Bot is a registered trademark of Parallax Inc. (www.parallax.com/go/boebot)— page 29 *(Boe-Bot robot)*

RoboCoaster Ltd., courtesy—page 21

Segway Inc., courtesy—page 22 *(Segway)*

©2011 Tec Weld Automation Inc., courtesy— page 18 *(manufacturing robot)*

U.S. Army/Daren Reehl, courtesy—page 20

U.S. Marine Corps/Lance Cpl. M.L. Meier, courtesy—page 23 *(BigDog)*

U.S. Navy, courtesy—cover and page 23 *(unmanned aerial vehicle)*

VEX Robotics—page 65 *(both)*

Wikipedia.org, courtesy—page 13 *(ASIMO robot)*

Wikipedia.org/Humanrobo, courtesy—page 77

Wikipedia.org/Francesco Mondada and Michael Bonani, courtesy—page 60

Wikipedia.org/Lance Cpl. Bobby J. Segovia, courtesy—page 19 *(telerobot)*

Wikipedia.org/Red Sunset, courtesy—page 18 *(teach pendant)*

Wikipedia.org/Mario Taddei, courtesy—page 13 *(mechanical knight)*

All other photos and illustrations not mentioned above are the property of or are protected by the Boy Scouts of America.

John McDearmon—pages 46 *(both),* 52, and 55

John McDearmon, Rick Folea, and VEX Robotics—all illustrations on pages 38–43; pages 48–51, 53, and 61

John McDearmon and Laszlo Hideg, Ph.D., P.E.—page 69