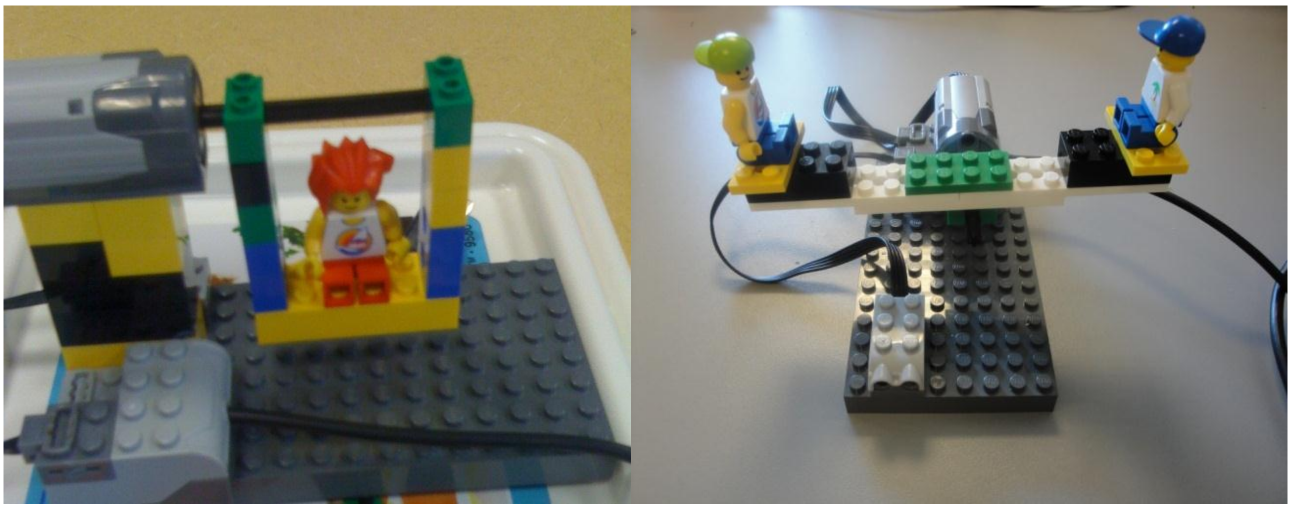
THE PLAYGROUND

First and Second Grade

A Curriculum Unit on Programming and Robotics



DevTech Research Group[1]

Eliot Pearson Department of Child Development

Tufts University

http://ase.tufts.edu/DevTech/tangiblek/

[1] This curriculum was developed by Amanda Sullivan with the help of Louise Flannery, Elizabeth Kazakoff, Adriana Flores, Ethan Peritz, and Amanda Puerto under the direction of Prof. Marina U. Bers.

Table of Contents

Introduction to the Curriculum

This curriculum introduces powerful ideas from computer science, specifically programming in a robotics context, to 1st and 2nd grade children in a structured, developmentally appropriate way. While the curriculum uses the technology of the LEGO® WeDo™ robotic kit, the powerful ideas are applicable to any other robotic construction kit. The term powerful idea refers to a central concept within a domain that is at once personally useful, interconnected with other disciplines, and has roots in intuitive knowledge that a child has internalized over a long period of time. The powerful ideas from computer science addressed in this curriculum include: the engineering design process, robotics, programming, repeat parameters, and sensors. These powerful ideas are explored in the context of a curriculum that draws on the theme of the playground and can be adapted to many other early childhood themes (see Appendix A). Each unit follows the same basic structure: 1) warm up games to playfully introduce or reinforce concepts, 2) introduction of the powerful idea through a challenge, 3) work individually or in pairs, 4) technology circle, 5) free-explorations, and 6) student's assessment. Teachers should adapt the lesson structure and its components to suit their class's needs.

Pacing

The curriculum unit is designed to take place over the course of one intensive week of work (i.e. in a camp setting or during a robotics focused week at school) or over the course of 1-2months with one or two shorter sessions per week. These numbers are certainly not set in stone. Depending on children's developmental levels and prior experience with digital technology, programming, and robotics, students might need more or less time than the guidelines here indicate. One issue for each teacher to resolve is how long to allot for each session, keeping in mind that each lesson can be spread out over several sessions to accommodate the classroom schedule and students' attention spans for this work.

Depending on the students, a class may benefit from between 1 and 2 hours to devote to their robotics and programming activities at a time.

Some classes or students may benefit from further division of the activities into smaller steps or from more time to explore each new concept before moving onto the next, either in the context of free-exploration or with teacher-design challenges. Each of the powerful ideas here can easily be expanded into a unit of study; the activities provide an introduction to each concept. For instance, students could explore a range of different activities and challenges with sensors to learn how they work in more depth.

To supplement the structured challenges, two to three hours of free-exploration are allotted throughout the curriculum. These open-ended sessions are vital for children to fully understand the complex ideas going on with their robotic creations and programs. The free-exploration sessions also serve as a time for teachers to observe students' progress and understandings. These sessions are as important for learning as the lessons themselves! In planning and adjusting the timeframe of this curriculum, free-exploration sessions should not be left by the wayside. Rather, if time is tight, teachers can consider leaving out a particular lesson altogether, giving children enough time to really understand and work with the ideas they are introducing to rather than skimming over all the lessons presented in this curriculum. Free-exploration provides opportunities for playing with materials and ideas. This will help build a solid foundation.

Materials

The robotic construction kit referred to in this curriculum comes is the LEGO ® Company's WeDo™ robotics construction set. This kit, and other LEGO® robotics materials, are available for purchase on the LEGO® Education website at: http://www.LEGOeducation.us/ (See also Resources in Appendix F). In the next few years there may be other robotics construction sets on the market for early childhood education which could be used with this curriculum.

A second, and important, type of material used in the curriculum is inexpensive crafts and recycled materials. The use of crafts and recycled materials, a practice already common in early childhood education, lets children build with a range of elements with which they are already comfortable. It may also bring down the costs associated with acquiring robotic supplies, since a blend of materials may reduce the need for complete, LEGO®-based kits and instead require the purchase of only select robotic parts which give functionality and movement to the creations.

For example, a kindergarten class in Boston created a robotic Freedom Trail, using cardboard boxes to recreate the historical buildings of the city. The children integrated the use of relatively more expensive LEGO® pieces, such as sensors, to bring their constructions to life[2]. This activity, while engaging for the children, also proved very successful for the teacher, who was already familiar with the use of recyclable materials and felt less intimidated by not having to learn about the mechanics of LEGO® bricks.

Pedagogy

The theory of constructionism developed by Seymour Papert (Papert, 1980; 1993) shows that children learn best when they construct digital artifacts and knowledge by playing with and exploring concrete materials. The social context of these explorations is also crucial, and teachers can provide scaffolding by creating a learning environment that supports children's explorations and experimentation. Through questions and observations, the teacher engages students in articulating and extending their own observations, thought processes, and explorations. The teacher may not directly answer students' questions but rather show them how to find it themselves. This kind of exploration fosters an environment in which what we often see as "failure" is actually a natural step of the learning process, a signal to ask questions and explore further. A more detailed account of working with young children and technology, especially robots, can be found in Blocks to Robotics: Learning with

---

[2] Bers, M. (2008). *Blocks to Robots: Learning with Technology in the Early Childhood Classroom.* New York, NY: Teacher's College Press .

Technology in the Early Childhood Classroom (Bers, 2008). See that book or an excerpt from it included in Appendix D for a description of supporting students with planning versus tinkering styles of approaching robotics, programming, designing, and problem-solving.

Theoretical Framework: Positive Technological Development

The theoretical foundation of this robotics curriculum is called Positive Technological Development (PTD) and was developed by Prof. Marina U. Bers from Tufts University (Bers, 2010a; Bers, in press). The PTD framework guides the development, implementation and evaluation of educational programs that use new technologies to promote learning as an aspect of positive youth development. The PTD framework is a natural extension of the computer literacy and the technological fluency movements that have influenced the world of education but adds psychosocial and ethical components to the cognitive ones. From a theoretical perspective, PTD is an interdisciplinary approach that integrates ideas from the fields of computer-mediated communication, computer-supported collaborative learning, and the Constructionist theory of learning developed by Seymour Papert, and views them in light of research in applied development science and positive youth development.

As a theoretical framework, PTD proposes six positive behaviors (six C's) that should be supported by educational programs that use new educational technologies, such as robotics. These are: content creation, creativity, communication, collaboration, community building and choices of conduct.

This curriculum engages young learners in:

1. ***Content creation***, by engaging children in making a robotic artifact and in programming its behaviors. The engineering design process of building and the computational thinking involved in programming foster *competence* in computer literacy and technological fluency. The use of design journals make transparent to the children themselves, as well as teachers and parents, their own thinking, their learning trajectories and the project's evolution over

time. The design process, like the scientific method, gives students a tool for systematically addressing a problem or need. Some children need constraints and top-down planning. Others enjoy working bottom-up and messing around with the materials to come up with ideas. Both learning styles are conducive for building competence in the technological domain and are welcomed and respected in this robotics program by providing different kinds of design journals explicitly designed to encourage differentiated instruction. See Appendix D for an explanation of how design journals should be used and for a sample design journal.

2. ***Creativity****,* by integrating different media such as LEGO® pieces, motors, sensors, recyclable materials, arts and crafts, and graphical elements from the programming language. This curriculum promotes creativity by fostering opportunities for children to develop their own project ideas, as opposed to copy them from a booklet or follow instructions, and by providing different materials for children to work with. As children approach solving technical problems in creative ways, they develop a sense of *confidence* in their learning potential. The PTD approach of working with robotics is based on the promotion of creativity, as opposed to efficiency, in problem solving. The original meaning of the term *engineering*, which derives from the Latin *ingenium,* means "innate quality, mental power, clever invention". However, clever or creative projects are difficult to make and the process can be frustrating. Our approach aims at helping children to learn how to manage frustration. One way for children to manage frustration in difficult lessons is by taking occasional breaks from the task that is frustrating them. For example, if a child is getting increasingly agitated working on a programming task, it is okay for them to take a few minutes to step away and work on the aesthetics of their robot and return to programming when they feel ready. Teachers can also schedule in "breaks" during long building and programming periods if the class is growing frustrated and bring everyone together to

reinforce concepts and troubleshoot problems. The learning environment is set up to create a culture in which it is expected for things not to work and in which succeeding the first time is seen as a rarity and as a sign that the child might not have challenged him or herself. Learning how to manage frustration is also associated with the development of *confidence*. As children go through this program they slowly realize their ability to find solutions, either by trying multiple times, by using different strategies or by asking for help. This is an important aspect of emotional development.

3. ***Collaboration,*** by engaging children in a learning environment that promotes working in teams, sharing resources and *caring* about each other. Most educational robotic programs for older children, such as the National Robotics Challenge and FIRST (**F**or **I**nspiration and **R**ecognition of **S**cience and **T**echnology), are set up as competitions, events where robots have to accomplish a given task – usually out-perform another robot. However, this program, instead of focusing on competition, promotes *caring* and working together. For that purposes it utilizes the collaboration web, a tool used to foster collaboration and support. At the beginning of each day of work, each child receives, along with their design journal, a personalized printout with his or her photograph in the center of the page and the photographs and names of all other children in the class arranged in a circle surrounding that central photo (see Appendix F for an example). Throughout the day, at the teacher's prompting, each child draws a line from his or her own photo to the photos of the children with whom he or she has collaborated. Collaboration is defined here as getting or giving help with a project, programming together, lending or borrowing materials, or working together on a common task. At the end of the week, children write or draw "thank you cards" to the children with whom they have collaborated the most.
4. ***Communication,*** through mechanisms that promote a sense of *connection* between peers or with adults. One of the ways this program engages students in communication is through

technology circles - a time for all, children and adults, to stop their work, put their projects on the table or floor, sit down in a circle together, and share the state of their projects. Technology circles present a good opportunity for problem solving as a community. Technology circles can be called as often as every twenty minutes at the beginning of a project, or only once at the end of a day of work, depending on the needs of the children and the need of the teacher to introduce new concepts. Some teachers have all the children sit together in the rug area for this. There are challenges to be addressed in order for technology circles to successfully serve their purposes. Kids' excitement to use the materials during introductory conversations or their post work tiredness during wrap-up discussions may necessitate special attention to the structure and expectations for group discussions. For instance, these conversations may need to be rather concise, making it a challenge to cover the many complex ideas presented in this curriculum's activities. To cover all the important ideas without losing the children's attention, the discussions might be broken up and held throughout the day rather than all at once. It can also be helpful to make a "Robot Parking Lot" for all the robots to go while they are not being worked on so children have empty hands help them focus at the technology circles. Each classroom will have its own routines and expectations around group discussions and circle times, so teachers are encouraged to adapt what already works in their class for the technology circles in this curriculum.

5. ***Community-building**,* through scaffolded opportunities to form a learning community that promotes *contribution* of ideas. The long term goal of this robotics program is not only to foster computational thinking and technological fluency amongst the participating children and teachers, but also amongst the wider community. In the spirit of the Reggio Emilia approach started by the Municipal Infant-toddler Centers and Preschools of Reggio Emilia in Italy after World War II, projects done by children are shared with the community via an

open house, demo day, or exhibition (Rinaldi, 1998). These open houses provide authentic opportunities for children to share and celebrate the process and tangible products of their learning with others who are also invested in their learning, such as family, friends and community members. Public displays of the learning process serve a dual function: to make learning visible to others and to the children themselves. In this spirit, final projects are shown to parents, friends and community members in the form of open houses. During these open houses, parents, siblings, and other family members visit the class for a demonstration of the children's final projects. Each child is given the opportunity not only to run their robot, but to play the role of teacher as they explain to their family how they built, programmed, and worked through problems.

6. ***Choices of conduct**,* which provide children with the opportunity to experiment with "what if" questions and potential consequences, and to provoke examination of values and exploration of *character traits.* In every classroom, there is always a child who learns quickly about mechanics, and thus has the choice to help her classmates in need or to build a bigger structure. The same is true for those children who become programming experts, those who can problem-solve and those who can mediate conflicts amongst group members. Although differentiation of roles is important for growing a responsible learning community, children are also encouraged to take on new roles and be flexible. Choices of conduct are not only made by children. Teachers make important choices in the way they display and introduce the materials to the children. For example, if the LEGO® building pieces are sorted by types and placed in bins in the center of the room (instead of giving an already sorted robotic kit to each child or group), children learn how to take what they need without depleting the bins of the "most wanted" pieces, such as special sensors or the colorful LEGO® mini-figures. They also learn how to negotiate for what they need. As a program developed following the PTD approach, the focus on learning about robotics is as

important as helping children develop an inner compass to guide their actions in a just and responsible way. One way to encourage positive choices is by using “Expert Badges”. Children who master concepts quickly can earn Expert Badges (a sticker for them to wear). A child wearing an expert badge uses the remainder of the class period helping any students who are having difficulty with the concepts they have mastered. Children wearing Expert Badges and actively helping others will also have an easier time completing their collaboration webs.

Classroom Management

Teaching robotics and programming in an early childhood setting requires careful planning and ongoing adjustments when it comes to classroom management issues. These issues are not new to the early childhood classroom or teacher, but they may play out differently during robotics activities because of the novelty and behavior of the materials themselves. Issues and solutions other than those described here may arise from classroom to classroom; teachers should find what works in their particular circumstances. In general, provide and teach a clear structure and set of expectations for using materials and for the routines of each part of the lessons (technology circles, clean up time, etc). Make sure the students understand the goal(s) of each activity. Posters and visual aids can facilitate children’s attempts to answer their own questions and recall new information.

Group Sizes

The curriculum refers to whole-group versus pair or individual work. In fact, some classrooms may benefit from other groupings. Piloting of this curriculum has shown that kindergarteners are better able to explore the main activities in the lessons when they have their own materials to work with and can go to other students for help, rather than collaborating with the same materials. Whether individual work is feasible depends on the availability of supplies, which may be limited for a number of reasons. However, an effort should be made to allow students to work in as small groups as possible, preferably individually, while working on the challenges. On the other hand, the curriculum includes numerous conversations which are enriched by multiple voices, viewpoints, and experiences. Some classes may be able to have these discussions as a whole group. Other classes may want to break up into smaller groups to allow more children the opportunity to speak and to maintain focus. Some classes structure robotics time to fit into a "center time" in the schedule, in which students rotate through small stations around the room with different activities at each location. This format gives students more access to teachers when they have questions and lets teachers tailor instruction and feedback as well as assess each students' progress more easily than during whole-group work. It is important to find a structure and group size for each of the different activities (instruction, discussions, work on the challenges, and the final project) that meet the needs of the students and teachers in the class.

Managing Materials

Classroom-scale robotics projects require a lot of parts and materials, and the question of how to manage them brings up several key issues that can support or hinder the success of the unit. The first issue is accessibility of materials. Some teachers may choose to give a complete WeDo™ kit of materials to each child, pair, or table of several children. Children may label the kit with their name(s) and use the same kit for the duration of the curriculum. Other teachers may choose to take apart the kits and have materials sorted by type and place all the materials in a central location. Since different projects require different robotic and programming elements, this set-up may allow children to take only what they need and leave other parts for children who need them. A word of caution, however: If materials are set-up centrally, they must be readily visible and accessible so children don't forget what is available to them or find it too much of a hassle to get what they need. Regardless, it is important to find a clearly visible place to set up materials for demonstrations, posters or visual aids to display for reference, and for robotics and programming materials for each lesson.

The second question is of usability. In some cases, children's desks or tables do not provide enough space to build a robot and program it on the computer. Care must be taken to ensure that children have enough space to use the materials available to them. If this is not the case they may tend towards choosing materials that fit the space but not their robotics or programming goal or their interface preference.

Teachers should carefully consider how to address these issues surrounding materials in a way that makes sense for their class's space, routines, and culture. Then, it is crucial to make expectations for how to use and treat materials explicit. These issues are important not only in making the curriculum logistically easier to implement, but also because, as described in the Reggio Emilia tradition, the environment can act as the "third teacher" (Darragh, 2006).

Student's Assessments

Children employ many different concepts and skills to create and program their own robots. The assessments for each lesson distill those ideas down to the 2-3 core powerful ideas of each activity. They are scored on a scale indicating how successful children were at achieving that concept or skill, from 0 (cannot achieve) to 5 (completely achieves). The final projects employ most or all the concepts covered in the lessons, depending on students' individual projects. See Appendix G for copies of assessments.

To keep assessment manageable in a busy classroom and also give children a tool to self-regulate their exploration process and self-assess, the assessment criteria given with each lesson can constitute a sequence of concrete achievements leading up to an "Engineer's License." Each lesson is associated with a different level, e.g. "Sturdy Builder" or "Programmer I," that incrementally completes the license, at which point the child is ready to start a final project. During the course of each lesson, children will explore and learn at different rates. When they have completed an assigned task, children should stop working on their project and signal a teacher that they are ready to be assessed (i.e. by putting a sign on their desk or a raising their hand). While waiting for a teacher to come around, children can work on accompanying literacy and math worksheets found in their Engineering Design Journals or having free-play exploration time. When a teacher comes around, children demonstrate their project for the teacher, who marks that licensure level on their certificate or helps them identify missing components. Children re-attempt any level until they have mastered it. This format allows for individual differences, helps teachers manage the amount of time assessment takes, and provides a fluid way for teachers to assess both individual progress and that of the whole class. Teachers should feel free to come up with their own analogy for the incremental assessment described here as "licenses." One teacher likened it to levels of achievement in video games that you must complete before moving on to harder challenges. See Appendix F for a sample Engineer's License.

The design journal for planning the final project can also provide a means of documentation. Appendix D shows a sample design journal. The writing can be done by children or by teachers taking dictation as appropriate. The components of the journal should be tailored for the nature of the final project. This format can also be adapted simply to document the final version of the project.

Academic Frameworks Addressed:
1st and 2nd Grade

This curriculum is designed for 1st and 2nd grade students and covers many foundational computer science and engineering skills that are not often taught in early childhood. These academic frameworks are taught through a series of powerful ideas: the Engineering Design Process, Robotics, Programming, and Sensors. Each powerful idea has activities and materials (in this case, the activities are tailored to fit the playground theme) that encourage mastery of the powerful idea and the foundational academic subjects that support it. In addition, the curriculum addresses foundational math, literacy, science, and art skills. Within each lesson in this curriculum, there are descriptions of at least one math and one language arts activity that fit in with the powerful idea being taught. Each lesson also features a list of relevant vocabulary words for children to master. Below is a brief summary of how the academic frameworks fit into each of these powerful ideas. To see specific math and language arts worksheets and activities, please see the Engineering Design Journals in Appendix E.

Powerful Idea 1: The Engineering Design Process

The Engineering Design Process refers to a cyclical process engineers use to design an artifact to meet a need. Its steps include: identifying a problem, looking for ideas, developing a prototype, testing, improving, and sharing solutions with others. The Engineering Design Process is introduced in Lesson 1 and is used as a problem solving technique throughout the remaining lessons. The Engineering Design Process is a tool students can continually apply to all subjects, experiments, and projects in the future.

In Lesson 1 (*What is the Engineering Design Process?)* children visit a real playground and later use LEGO® and art materials to build their favorite playground structures. They are given various constraints in the design of their structures and must use the steps of the Engineering Design Process to plan, create, test, and share their structures until they have a prototype that meets all the constraints.

Math

Using LEGO® to build their playground structures, children must reason with shapes and their attributes (Common Core Standards, Grade 1). This lesson will also reinforce skills mastered in previous years regarding labeling and understanding shapes, as children draw out the shapes they use in their Engineering Design Journals (booklets filled with worksheets and blank space where children plan, answer questions, draw, etc.) and talk about why they chose those shapes in their structures. Beginning in the first lesson, children will be given "inventory" activities that will continue throughout the curriculum. Children will be asked to count different types of pieces used, record numbers in a chart, and determine what was used the most and the least. These ongoing inventory activities will serve to meet required data and number skills (Common Core, Grade 1 and 2).

Language Arts

In their Engineering Design Journals, children will use a combination of drawing, dictating, and writing to explain what they have designed and why. Children will use language for self-expression and practicing vocabulary when writing a postcard home to their families about what they saw on their field trip to a playground (NY State Language Arts Frameworks and Standards, Grades 1-2). Children will recall information from the activity to answer questions in their journals (Common Core, Grades K-2). When problem solving and coming up with the designs for their structures, children will participate in collaborate conversations with a partner or group (Common Core, Grades K-2). Finally, children share their work in the Technology Circle for the first time. Technology Circles are a daily activity when the class comes together to share their work, discuss what they have done that day, troubleshoot un-resolved questions as a group, and answer any questions. With prompting and support, children recount their experiences with appropriate facts and details in an articulate manner (Common Core, Grades K-2).

Art

In Lesson 1, children are asked to think creatively and artistically. They will draw out their playground structures during the "planning" stage in their Engineering Design Journals. After their LEGO® structures are built, they will use a variety of arts and crafts materials to decorate and personalize their structures. This exploration of art materials and exercise in imagination fits in with the Benchmark 2nd grade Art Making Standards.

Powerful Idea 2: Robotics

Robotics is an engineering field focused on the creation of robots, machines which can automatically follow instructions to do tasks. Robotics is first introduced in Lesson 2 (*What is Robotics?*). In Lesson 2, children share and learn ideas about what robots are. They are introduced to basic WeDo™ robotics concepts. Using different robotics, LEGO® , and art materials children build functioning robotic swings for their playground.

Math

Children continue their work in describing shapes, and reasoning with shapes in the design and building of their robotic swings (Common Core Standards, Grade 1). Children will complete an inventory activity that involves counting and recording data in a chart (Common Core Standards, Grade 1 and 2).

Language Arts

In Lesson 2, children are introduced to a variety of new words and concepts when shown robotic materials for the first time. Through a variety of group and individual activities, children work on vocabulary as they receive prompting and support for asking and answering questions about unknown words (NY State Language Arts Frameworks and Standards, Grades 1-2). Children continue to use their Engineering Design Journals for writing and drawing explanations for their work and answering questions (Common Core, Grades K-2). Children practice speaking skills in the Technology Circle (Common Core, Grades K-2).

Science

In this lesson, children use the Engineering Design Process to make predictions, make decisions, observe, and manipulate materials (Inquiry Skills, based on NYC MST Standards, Grades K-8).

Powerful Ideas 3 and 4: Programming and Programming with Repeats

A program is a sequence of instructions that the robot acts out in order. Each instruction as a specific meaning, and the order affects the robot's overall actions. Instructions can be modified with a special instruction to repeat. Parameters, extra pieces of information, can make loops repeat forever or for a specific number of times.

Programming is addressed primarily in Lesson 3 through the activities *What is Programming?* and in Lesson 4 through the activity *What are Repeats?.* In Lesson 3, children first build and program a robotic seesaw. When this is mastered, children use repeats correctly to make their seesaws tilt from left to right a particular number of times.

Math

When learning to use the repeat parameters in their programs, children are asked to draw upon previous math knowledge of number names and the count sequence. Children who master this easily can receive more challenging activities using the repeats with their robotic seesaws requiring the use of problem solving through addition and subtraction (Common Core, Grades 1-2). To understand repeats, children must also master to concept of patterns (TERC, Grades 1-2)

Language Arts

Children continue to use their Engineering Design Journals to answer questions and write/dictate/draw about their work (Common Core, Grades K-2). Children participate in structured collaborative conversations with a partner or group in order to troubleshoot problems (Common Core, Grades K-2). Children practice speaking skills in the Technology Circle (Common Core, Grades K-2).

Science

Children must observe, analyze, and report observations (Process Skills, based on the NYS MST Standards). Based on these observations, they make and record their predictions for how they will reach their goal (Process Skills, based on the NYS MST Standards).

Powerful Idea 5: Sensors

A robot can use sensors, akin to human sense organs, to gather information from its environment. Sensor data can become parameters for controlling flow instructions. WeDo™ tilt sensors and motion sensors are introduced in Lesson 4 (*What are Sensors?*). Children first correctly attach and program the tilt sensor to their robotic seesaws (made in Lesson 3). If this is mastered, children attach and program a motion sensor.

Math

In their Engineering Design Journals, children will record data regarding the number of times their seesaw tilts back and forth using the tilt sensor and the motions sensor. If children master this, they may represent their data by creating a chart or graph (Common Core, Grade 1-2).

Language Arts

Children must recall information from their experiences using the different sensors in order to answer questions in their Engineering Design Journals (Common Core, Grade 1). They continue to practice speaking skills in the Technology Circle (Common Core, Grades K-2).

Science

Children practice observing and inferring (Inquiry Skills, based on NYS MST Standards, Grades K-8). They compare and contrast the tilt and motion sensor (Inquiry Skills, based on the NYS MST Standards). They observe cause and effect relationships (Process Skills, based on the NYS MST Standards).

Advanced Lesson: Gears

Gears are mechanical parts with cut teeth of such form, size, and spacing that they mesh with teeth in another part to transmit or receive force and motion. A mastery of gears is necessary for children to create increasingly complex robotics creations. An introduction to gears is listed another optional lesson that teachers can choose to complete or omit.

Math

In Engineering Design Journals, children complete a worksheet on the math behind gears. This includes a discussion of direction, size, and speed (Common Core, Grades K-2).

Language Arts

Children practice speaking and communication skills in a group activity called Human Gears as well as in the technology circle(Common Core, Grades K-2).

The Final Project

The powerful ideas introduced in lessons 1-7 culminate in a final project that draws upon all of the building, programming, math, science, literacy, and art skills children have learned.

Table 1: *Powerful Ideas within the Activities*

Powerful Idea	**Definition**	**Activity**
Engineering design process	A cyclical process engineers use to design an artifact to meet a need. Its steps include: identifying a problem, looking for ideas, developing a prototype, testing, improving, and sharing solutions with others.	***What is the Engineering Design Process?*** Children will use LEGO® and art materials to build their favorite playground structure. ***Goals:*** - To gain experience using the Engineering Design Process in order to solve problems and complete a task -To gain expertise in sturdy building
Robotics	An engineering field focused on the creation of robots, machines which can automatically follow instructions to do tasks.	***What Is a Robot?*** Children share and learn ideas about what robots are. They are introduced to WeDo™ robotics concepts. ***Goals:*** -To gain expertise in the different WeDo™ robotics parts and use this expertise to build their own robotics creations. -To be exposed to a brief introduction to programming instructions.
Programming: Control Flow by Sequencing and Instructions	A program is a sequence of instructions that the robot acts out in order. Each instruction has a specific meaning, and the order of the instructions affects the	***What is Programming?*** Children build and program a robotic seesaw.

	robot's overall actions. Instructions can be modified with a special instruction to repeat. Parameters, extra pieces of information, can make loops repeat forever or a specific number of times.	***Goals:*** - To gain expertise in what a program is. -To figure out and program the proper sequence of instructions in order to make their seesaw go back and forth ***What are Repeats?*** Children are introduced to the concept of repeats. ***Goals:*** -To use repeats correctly to make their seesaws tilt from left to right a particular number of times.
Sensors	A robot can use sensors, akin to human sense organs, to gather information from its environment. Sensor data can become parameters for controlling flow instructions.	***What are Sensors?*** Children are introduced to the tilt sensor and the motion sensor and add them to their seesaws. ***Goals:*** -To gain an understanding of sensors and how they work. -To use a motion sensor and tilt sensor appropriately with their robots.

Table 2: *ITEEA Standards and MA Frameworks Addressed*

Powerful Idea	**Definition**	**Activity**
Engineering design process	A cyclical process engineers use to design an artifact to meet a need. Its steps include: identifying a problem, looking for ideas, developing a prototype, testing, improving, and sharing solutions with others.	***What is the Engineering Design Process?*** Children will use LEGO® and art materials to build their favorite playground structure. ***Goals:*** - To gain experience using the Engineering Design Process in order to solve problems and complete a task -To gain expertise in sturdy building
Robotics	An engineering field focused on the creation of robots, machines which can automatically follow instructions to do tasks.	***What Is a Robot?*** Children share and learn ideas about what robots are. They are introduced to WeDo™ robotics concepts. ***Goals:*** -To gain expertise in the different WeDo™ robotics parts and use this expertise to build their own robotics creations. -To be exposed to a brief introduction to programming instructions.
Programming: Control Flow by Sequencing and Instructions	A program is a sequence of instructions that the robot acts out in order. Each instruction has a specific meaning, and the order	***What is Programming?*** Children build and program a robotic seesaw.

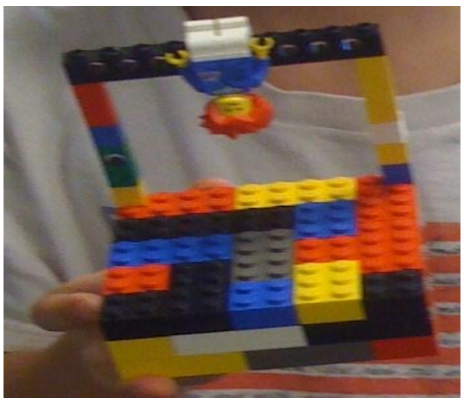
	of the instructions affects the robot's overall actions. Instructions can be modified with a special instruction to repeat. Parameters, extra pieces of information, can make loops repeat forever or a specific number of times.	***Goals:*** - To gain expertise in what a program is. -To figure out and program the proper sequence of instructions in order to make their seesaw go back and forth ***What are Repeats?*** Children are introduced to the concept of repeats. ***Goals:*** -To use repeats correctly to make their seesaws tilt from left to right a particular number of times.
Sensors	A robot can use sensors, akin to human sense organs, to gather information from its environment. Sensor data can become parameters for controlling flow instructions.	***What are Sensors?*** Children are introduced to the tilt sensor and the motion sensor and add them to their seesaws. ***Goals:*** -To gain an understanding of sensors and how they work. -To use a motion sensor and tilt sensor appropriately with their robots.

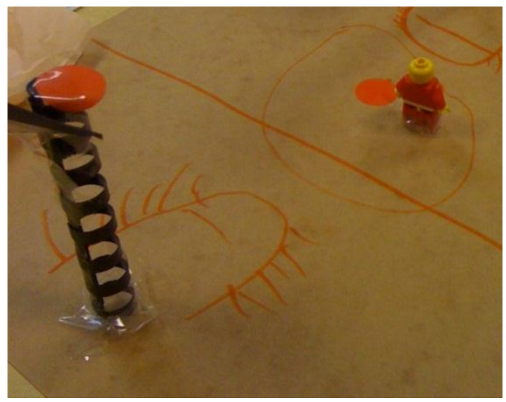
The Curriculum

Overview:

Children use LEGO® and art materials (all non robotic materials) to build their favorite playground structure. The powerful idea in Lesson 1 (building sturdily through use of the engineering design process) will prove important to the success of the children's robots in subsequent lessons and should be rearticulated and discussed during each activity.

Examples of child projects: Monkey bars made of LEGO® and a basketball court made of art materials.





Prior Knowledge	***Objectives***	
	Students will understand that…	*Students will be able to…*
• None, but prior experience building with LEGO® and crafts or recycled materials is helpful.	• LEGO® bricks and other materials can fit together to form ***sturdy structures***. • The ***engineering design process*** is useful for planning and guiding the creation of artifacts.	• Build sturdy, non-robotic playground structures • Use the engineering design process to facilitate the creation of their structure.

Materials / resources:

- LEGO® bricks and a variety of crafts and recycled materials for building and decorating
- Poster showing the steps of the engineering design process (see Appendix C)
- Engineering Design Journals for planning (see Appendix D)

Activity description

Warm- up : Playground Visit

Children will take their Engineering Design Journals on a field trip to a playground (you can show pictures/videos of a playground if this is not possible) in order to connect real life playground structures to the structures students will build using LEGO® and art materials. Children will choose two structures to describe in their journals.

Math Connection: *Shapes on the Playground*

Ask children to identify different shapes they see making up the two structures they have chosen. Back in the classroom, look at LEGO® and art materials and ask children to identify the shapes they see and make another list of their answers. Compare the two lists. What shapes do they have in common? How can we use materials in the classroom to build structures that look like the real ones? In this activity, children will work to identify and describe 2d shapes as well as 3d shapes (see vocabulary section).

Introduce the concepts and the task: "Today we will be building our favorite playground equipment for toy people to use, and we're going to use a tool to help us make sure our structures are sturdy and work the way they are supposed to." Discuss what an engineer is and introduce the steps of the engineering design process (see Appendix C for a poster).

What is an engineer?

An engineer is anyone who invents or improves things (for instance, just about any object you see around you) or processes (such as baking methods) to solve problems or meet needs. Any man-made object you encounter in your daily life was influenced by engineers.

Think Like an Engineer

Everyone in the class is going to start thinking like an engineer! That means looking at the purpose of objects and how they function. What are the different parts that make up the whole? What do they do? Why are they important? Let's look at pictures of some playground equipment (See Appendix F) and ask these engineer's questions.

Ex 1: Slide- What are the different parts of the slide? What function does each part have? Why is each part important?

Ex.2: Swings- How do these swings work? What function does each part have? What would happen if it had different parts?

Jump For Engineers

Look at a series of pictures of naturally occurring and manmade objects (See Appendix F). Jump if you think an engineer built it, stay seated if you don't think so. Why or why not? Discuss.

Lesson 1 Vocabulary
Students should become familiar with the following words:

Review the names of 2D shapes (i.e. circle, rectangle, square, triangle, etc.) if needed
Artifact – something important made by people
Base – the bottom face of a three-dimensional shape
Cube – a three-dimensional shape with six equal faces
Cycle – something that moves in a circle (i.e. the seasons, the Engineering Design Process)
Cylinder – a three-dimensional shape with two circle bases
Design – a plan for a building or invention
Edge – the border of a two-dimensional shape or face
Engineer – someone who invents or improves things
Face – the flat part of a three dimensional shape
Material – something used to build or construct
Prism – a three-dimensional shape with at least three faces
Sphere – a round, three-dimensional shape with no faces or edges
Structure – a building or object made with different parts

Individual / pair work : Students follow the steps of the engineering design process and use LEGO ® and crafts or recycled materials to create their favorite playground or sporting equipment (e.g. slide, basketball court, monkey bars). They may use both structural and aesthetic materials. Students should demonstrate to a teacher that their structures meet the following criteria as they are ready.

The criteria for a successful structure are that:

- At least one toy person can be attached and detached to "use" the structure
- It remains intact when picked up, moved around, and handled

Language Arts Connection: *Postcard Home*

Children will recall their trip to the playground . They will try to remember the structures they saw, the shapes they noticed, and what they liked the most. Children will fill out a blank postcard (in Engineering Design Journals) where they will draw pictures and write a message to their family describing their trip. They will try to use as many vocabulary words as possible. When postcards are complete, cut them out so that children can mail them or take them home!

Note: Working Individually vs. Working in Pairs
Whether students work in pairs versus individually is left up to the teachers' discretion based on several factors. Materials may be limited, making pair work necessary. Teachers may also have goals for children's social development that an explicit focus on sharing and teamwork throughout this curriculum can

support. On the other hand, teamwork can be challenging at this age, so students may benefit from having their own materials and the option rather than the requirement to collaborate with others when it makes sense.

Engineering Experts: Children who finish building their playground structures and master all concepts quickly get to wear a badge that says "Engineering Expert". Engineering Experts walk around and offer help to any classmates experiencing difficulties.

Collaboration Web: As children progress through the lesson, they will complete their collaboration webs. They will draw lines from their picture to the pictures of any classmates who give them help. If children say they didn't receive any help, remind them to think of their partners, class Experts, or if they got any ideas by looking at another classmate's project.

Technology Circle: After finishing, students share their creations. They may:

a. explain the features of their creation
b. show how their creation moves or how the toy person could use it
c. describe the features of their final design that make it sturdy
d. talk about what they found easy and difficult, and
e. share anything they changed from their original plan.
f. share postcards
g. share collaboration webs

Free-play:

Provide opportunities for children to build freely with LEGO® and other arts and crafts materials.

Overview:
Children share and learn ideas about what robots are. They are introduced to LEGO® WeDo™ robotics concepts. Children will build and test their own robotic swings.

Example of a child built robotic swing:

Prior Knowledge	***Objectives***	
	Students will understand that…	*Students will be able to…*
• LEGO® bricks and other materials can fit together to form sturdy structures. • The engineering design process is useful for planning and guiding the creation of artifacts. • Symbols (pictures, icons, words, etc) can represent ideas or things. • Some ability to recognize letters or to read is helpful, but not required.	• Robots need moving parts, such as motors, to be able to perform behaviors specified by a program. • The robotic 'brain' (in this case, the computer) has the programmed instructions that make the robot perform its behaviors. • The computer must communicate with the motors for the motors to function.	• Describe the components of a robot, including the 'brain' (computer), motors, and wires. • Upload a program to a robot • Build sturdy, robotic swings that move.

Materials / resources:

- Pictures of different robots and non-robots
- Large icons for games and reference displays
- Computers with WeDo™ software
- One set of robotic parts for each student/pair
- LEGO® bricks and a variety of crafts and recycled materials for building and decorating

- Some partially built structures (or pictures of them) to show possible attachments

> ***Note:***
> *It is important to establish rules or expectations for how students should treat eachothers' materials, programs, and robots. Find a time for students generate these group expectations. Students may be better able to imagine reasonable expectations after using the robots or programming interface once.*

<u>Activity description</u>

<u>Warm up activities</u>:

1) *Jump for the robots!* Children will be shown about 10 different images of robots and non-robots. They jump up and down if they think the picture shown is of a robot. Later, make an "Is It a Robot?" chart putting these images in one of three categories: Robots, Maybe or Sort of Robots, and Not Robots.

   Yes or No? Students jump up (or make another movement) for statements they think are true and sit

> ***Math Connection:*** *Graphing & Subtraction*
>
> *Incorporate graphing into this exercise by making a chart with True and False for each question along the horizontal axis and number of students along the vertical axis. Have students place a marker (sticker, symbol, etc) with their initials in the "True" column or the "False" column. As a class, children will be able to interpret the graph in order to see whether there were more "True" or "False" responses for each question. Have children figure out the difference between the number "True" and "False" responses for each question for practice with subtraction.*
>
> 1. *Robots are machines (YES).* _____
> 2. *All robots are made of the same materials (NO).* _____
> 3. *Robots must have moving parts (YES).* _____
> 4. *Robots can think by themselves (NO).* _____
> 5. *All robots look like alike (NO).* _____
> 6. *Robots must be able to move around the room (NO).* _____
> 7. *Robots are operated using remote controls (NO).* _____
> 8. *People tell robots how to behave with a list of instructions called a program (YES).* _____
> 9. *Some robots can tell what is going on around them (YES).* _____
>   *(Examples: sensing light, temperature, sound, or a touch.)*
> 10. *Robots are alive (NO).* _____

2) <u>Discussion: What is a robot</u>? As a class, children discuss what they think a robot is and examples of robots they know of. Children and teachers can bring in pictures of these objects later and put them on the "Is It a Robot?" chart. The teacher shows a pre-built WeDo™ playground structure and a non-robotic playground structure. The class identifies that you have to push the non-robot to make it move. You can also push the robot, but (as the teacher shows) you can give it instructions and push a button to make it follow them. Why can the robot do this? It has special parts, which the teacher overviews now. Students will learn about them more later.

<u>Building and Programming a WeDo™ Robot</u>

<u>Introducing the concepts and task</u>: Build robotic swings that are programmed to move.

1. Review the robot's key parts and their functions.
2. *Individual/pair work:* Students build their own robotic swings. Allow the students to build how they see fit, but remind them that a working robot must be connected to a computer 'brain', motors, properly connected wires, and properly connected USB hub. When they think they have a working robot, they attempt to send a test program to their robot (see part 2). This test is to ensure that their robot follows the instruction properly and that it is sturdy. Teachers can help make sure the robots' wires are properly oriented.
3. *Communication with a robot:* Explain that we can tell a robot what to do, as long as we use a language it understands. Encourage the students to offer examples of how people communicate (speaking, writing, drawing, facial expressions, etc) and other languages they (or people they know) can speak. Discuss the idea of translating between languages, and the need to translate what we want a robot to do into the robot's language. A *program* is another word for instructions we give the robot.
4. Show how to use the programming interface on the computer. Briefly describe the icons (children will learn more about programming in the next lesson). In this lesson, children will solely concentrate on programming their motors to move in order to test their robotic creations.

Programming Motors With WeDo™ (From the WeDo™ Resource Guide) :

1) Attach axle to the motor.

2) Attach the motor wire to the LEGO® Hub. It works on either port. Connect hub to computer with the hub's USB cable.

3) Drag and drop the Blocks from the Palette to the Canvas to build the following program:
Start, Motor This Way.

4) Click the Start Block.

The motor moves. The axle turns.

5) To stop the program and turn off the motor, click the Stop button.

Discussion:

What does the motor do?

Turns on and makes the axle move.

What does the Start Block do?

The Start Block is the beginning of the program. After you click the Start Block, the program starts running. In this example, the Motor This Way Block runs.

What does the Motor This Way Block do?

The Motor This Way Block turns on the motor in the clockwise direction.

Lesson 2 Vocabulary

Students should become familiar with the following words:

Automatic – by itself, without help from a person
Axle – a pin, pole, or bar on or with which a wheel revolves
Computer – a machine that gives a robot its program or instructions
Function – the reason a machine or robot was built
Hub – the part of a robot that connects it to the computer or brain
Joint – a part of a robot that can turn
Motor – the part of a robot that makes it move
Robot – a machine that can be programmed to do different things
Wires – the long, skinny tubes that connect all the robot's parts

Language Arts Connection: *How-To Guide*

You need to explain to someone else how to build and program swings the way you did. In their Engineering Design Journals, provide written instructions explaining the steps you went through using as much new vocabulary as possible. Then, draw a picture of your swings labeling all robotic and non-robotic parts. When you are all done, switch how-to guides with a partner and try to follow their instructions! Could you follow theirs? Could they follow yours? If not, what was missing?

Robot Experts: Children who finish building their robots and master all concepts quickly get to wear a badge that says "Robot Expert". Robot Experts walk around and offer help troubleshooting to any classmates experiencing difficulties building a functional robot. Encourage the experts to offer help by going back to the appropriate step of the engineering design process with the person they are helping (i.e. returning to planning or testing).

Collaboration Web: As children progress through the lesson, they will complete their collaboration webs. They will draw lines from their picture to the pictures of any classmates who give them help. If children say they didn't receive any help, remind them to think of their partners, class Experts, or if they got any ideas by looking at another classmate's project.

Technology Circle: After about 45 minutes of building, have the students share their creations with the rest of the class (or a small group). During this time, students can share the parts and features of their robot, share what they found easy or difficult, or share what makes their robot sturdy. What do you think will happen if you make a robot that is missing one of its pieces? Try it out!

Concluding activity: See Appendix B for examples.

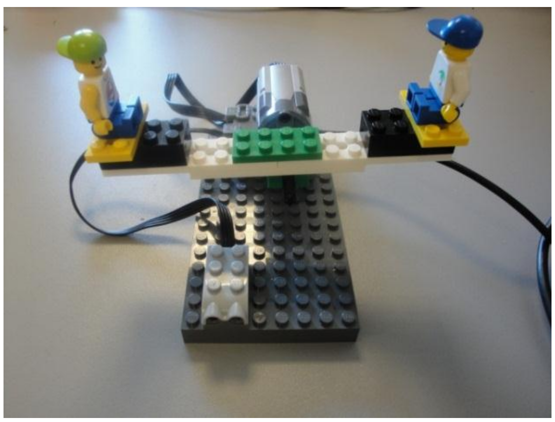
Free-play: Free exploration of building and programming with robotic materials.

Overview: Children build and program a robotic seesaw in order to gain expertise in what a program is.

What Is a Program?

A program is a sequence of instructions that the robot acts out in order. Each instruction has a specific meaning, and the order of the instructions affects the robot's overall actions.

Example of a robotic seesaw (See Appendix F for detailed building instructions):



Prior Knowledge	***Objectives***	
	Students will understand that…	*Students will be able to…*
• A robot is a machine that can act on its own once it receives proper instructions.	• Each icon or "block" corresponds to a specific instruction • A program is a sequence of instructions that is followed by a robot • The order of the instructions dictates the order in which the robot executes the instructions	• Point out or select the appropriate block corresponding to a planned robot action • Connect a series of blocks on the computer • Transmit a program to a robot

Materials / resources:

- Large icons for games and reference displays
- Robotic and non-robotic building materials
- Computers with Wedo software

<u>Activity description</u>

<u>Warm-Up</u>: Play *Simon Says* or another game from [Appendix B](#) to learn/ review each of the WeDo™ programming icons and what each icon represents.

<u>Introduce the concepts and task</u>. Show an example robotic seesaw programmed to tilt back and forth with toy people riding on it. "Today we will build and program robotic seesaws for toy people to play on!"

<u>Activity:</u> Individually or in pairs, children will build their own robotic seesaws. Once built, children will experiment with programming their seesaws to tilt back and forth. Children will use cut-outs or stickers of the programming icons to put together programs in their Engineering Design Journals and work through the accompanying planning and reflecting questions.

Math Connection: *Reporting on Data*

On a blank piece of paper , students write or draw what robotic, LEGO® , and other pieces they used and:

1) *Label the pieces by shape*
2) *Count how many of each shape they used*
3) *As a class, make a chart to see which shapes/pieces were used most and least by students. (Or, have each student make a chart individually)*
4) *Report on the data! (i.e. "today our class used a grand total of __ motors, __ axles, and ___ hubs, etc.)*

Lesson 3 Vocabulary:

Instruction – a direction that a robot will listen to
Program – a set of instructions for a robot
Sequence – the order of instructions that a robot will follow exactly
Variable – something in a program that can change

Language Arts Connection: *Programming Charades*

Children will pair up. One child will make up a program using the WeDo™ instructions and act it out while the other partner guesses what the programming instructions are. Switch roles. Then, come up with a program together that you will write down and act out for the class.

<u>Programming Experts:</u> Children who finish programming their robots and master all concepts quickly get to wear a badge that says "Programming Expert". Programming Experts walk around and offer help troubleshooting to any classmates experiencing difficulties programming.

<u>Collaboration Web:</u> As children progress through the lesson, they will complete their collaboration webs. They will draw lines from their picture to the pictures of any classmates who give them help. If children say they didn't receive any help, remind them to think of their partners, class Experts, or if they got any ideas by looking at another classmate's project.
<u>Technology circle</u>: Children will share their robotic seesaws with the group and talk about their process making and programming them. As a class, they will help troubleshoot problems if any children could not get their seesaws to function properly.

<u>Concluding activity</u>: Simon Says, or another game. See Appendix B for other suggestions.

<u>Free-Play</u>: Students continue to create and upload programs to a robot. As students are ready, prompt them to plan ahead about what they want the robot to do.

Lesson 4
What Are Repeats?

Powerful Ideas:
*Repeats**:** Loops and Number Parameters*

Overview

Students will learn about a new instruction that makes the robot repeat other instructions infinitely or a given number of times. They use these new instructions to program robotic seesaws to move from left to right a particular number of times.

Prior Knowledge	***Objectives***	
	Students will understand that…	*Students will be able to…*
• Arranging instructions in a different order will result in a different program.	• An instruction or sequence of instructions may be modified to repeat. • Some programming instructions, like 'Repeat,' can be qualified with additional information.	• Recognize a situation that requires a looped program. • Make a program that loops. • Use number parameters to modify the number of times a loop runs.

Materials / resources:

- Large icons for games and reference displays
- Robotic and non robotic building materials
- Computers with WeDo™ software

Activity description: Individually or with partners, children use their already built seesaws and modify their programs to utilize Repeats.

Warm-Up: Game or song that uses repetition. See Appendix B for examples.

Introduce the concepts: Repeats

1. Discuss what it means for something to repeat. How does this relate to similar concepts like patterns?
2. Introduce the "Repeat" programming icon. Show the different ways you can program a Repeat.
3. Using a sample robot and program, demonstrate a robot acting out a pattern by repeating certain actions multiple times.

Lesson 4 Vocabulary:

Loop – something that repeats over and over again
Parameter – a limit that a robot will follow
Pattern – a design or sequence that repeats
Repeat – to do something more than once

Math Connection: *Patterns & Counting*

After showing a sample program that is a pattern, children will identify the repeating unit, how many times it repeats, and (as a class) change the program so that it uses a repeat and accomplishes the same outcome.

The task: Children will use repeats correctly to make their seesaws tilt from left to right 5 times (one full cycle tilting from left to right is considered "1" time). Once this is mastered, use repeats to create a more complex repeating pattern (i.e. add a repeating sound or add another repeating pattern that happens after the seesaw moves back and forth 5 times).

Language Arts Connection: *Toothbrush Exercise*

Think about the way your brush your teeth- this is a task that requires some repeating motions (like moving your toothbrush from left to right) and other motions that only happen once (like squeezing out toothpaste). Pretend YOU are a robot needs a program to brush your teeth. Using WeDo™ programming instructions (and made up instructions like "spit" and "rinse"), write out a tooth-brushing program that uses repeats. Switch programs with a partner and act out their instructions. Were they the same as yours? Does anything need to be changed?

Repeats Experts: Children who finish programming their robots and master all concepts quickly get to wear a badge that says "Repeats Expert". Repeats Experts walk around and offer help troubleshooting to any classmates experiencing difficulties programming with repeats.

Collaboration Web: As children progress through the lesson, they will complete their collaboration webs. They will draw lines from their picture to the pictures of any classmates who give them help. If children say they didn't receive any help, remind them to think of their partners, class Experts, or if they got any ideas by looking at another classmate's project.

Technology Circle:

1. Students share their programs and discuss how Repeats work, especially how order is important.

See Appendix B for more examples of possible concluding games.

Free-play:

Students need to explore the new instructions. They should build programs that use (or don't use) them. In doing so, they will gain comfort with sequencing the blocks correctly, how the robot follows instructions depending on where the repeat is placed.

Overview

Children are introduced to the tilt sensor and the motion sensor and add them to their seesaws.

Prior Knowledge	***Objectives***	
	Students will understand that…	*Students will be able to…*
• Examples of human or animal sense organs and that people and animals use information provided by their senses to help make decisions.	• A robot can feel and see its surroundings with a sensor. • A robot can react to collected data by changing its behavior. • Certain instructions (like "Repeat") can be modified with sensor data.	• To use a motion sensor and tilt sensor appropriately with their robots. • Compare and contrast human sense and robot sensors

Materials / resources:

- Large icons for games and reference displays
- Robotic and non robotic building materials
- Computers with WeDo™ software
- WeDo™ tilt and motion sensors
- Projection screen or television to show video clips of different kinds of sensors

Activity description

Warm-Up: Game or song that uses the 5 human senses. For example, see:

Barney Five Senses song at: http://www.youtube.com/watch?v=sipHgMvOc6Y or,

When You Use Your Five Senses at:
http://www.youtube.com/watch?v=Ipb0sEwpWjo&feature=bf_next&list=WL96B2F66D3C3EFB12&index=2

Introduce the concepts: Sensors and Sensor Parameters:

1. Discuss examples of human / animal senses and how these senses let us gather information about what's going on around us, so that we can make decisions based on this information.
2. Show the motion and tilt sensors and explain how they work.
3. We need programming instructions to tell the robot what to do with the information from its sensors. Show the Repeat icons, which are now familiar, and the icons for the Tilt sensor and Motion sensor.
4. Run a sample program, and have students discuss what the robot is doing.

Examples of Robots With Sensors:

Snuffles the Elephant: http://www.youtube.com/watch?v=IdqYgE8Mcqo

WeDo™ Alligator: http://www.youtube.com/watch?v=vbzcMDYyoRk

Robots Using Sensors to Locate a Ball: http://www.youtube.com/watch?v=sXKtV-_gsMo

Robot Goalie: http://www.youtube.com/watch?v=VTlW_d1XRE4

Pressure Sensor: http://www.youtube.com/watch?v=pXMt0fKjJkM

Using WeDo™ Tilt Sensors with a Sample Program (From the WeDo™ Resource Guide):

1) Attach the tilt sensor wire to the LEGO® Hub. It works on either port.
2) Click the Arrow button on the Palette to see all of the Blocks.
3) Drag and drop the Blocks from the Palette to the Canvas to build the following program: Start, Display Background, Wait For, Display Background.
4) On the Wait For Block, drag and drop a Tilt Sensor Input on top of the Number Input.
The Tilt Sensor Input replaces the Number Input.
5) On the second Display Background Block, move the mouse pointer over the Number Input and type 2.
The Input changes to the number 2.
6) Click the Start Block.
The program opens the Display Tab and shows the first background. Then the program waits ***until you tilt the sensor upward*** *and the Display Tab shows the second background.*

Discussion
1) What does the tilt sensor do?
The tilt sensor tells the computer when it is pointed up, down or in other directions.
2) Which Blocks do you use to program the tilt sensor?
Wait For with a Tilt Sensor Input.
3) How does this program work?
The program shows a background in the Display Tab and then waits for someone to tilt the sensor upward. When the tilt sensor tilts upward, the program shows another background.
4) The tilt sensor can also be pointed in other directions. Click on the Tilt Sensor Input in your program to find out how many ways it can be tilted.
Six ways: Up, Down, This Way, That Way, No Tilt, Any Tilt.
5) Change your program to use a different Tilt Sensor Input.
Change the Tilt Sensor Input to any of these other options. Then when the program runs again, it will wait for the new tilt input direction before changing to another background.

Using WeDo™ Motion Sensors with a Sample Program (From the WeDo™ Resource Guide):

1) Attach the motion sensor wire to the LEGO® Hub. It works on either port.
3) Drag and drop the Blocks from the Palette to the Canvas to build the program shown:
Start, Motor On For, Display.
4) Drag and drop a Motion Sensor Input on top of the Number Input that was automatically attached to the Wait For Block. The Motion Sensor Input replaces the Number Input.
5) Click the Start Block. Then move your hand in front of the motion sensor.
*The program waits to see your hand then displays **abc.***

Discussion
What does a motion sensor do?
It sees objects or movement and reports to the computer.
What is the Display Block programmed to show?
The Display Block in this program shows the letters abc. It can also be programmed to show other words or numbers. See the Programming Tip.

The task: The students add a motion and then tilt sensors to their robot (one at a time) and program their seesaws to tilt back and forth until otherwise instructed by the sensors.

Math Connection: *Estimation and Counting*

After students add motion sensors to their robots, have them get together with a partner. The teacher will control a stopwatch. Students will estimate how many times the seesaw will go back and forth in 15 seconds. One partner will record and count the number of tilts while the other starts the program and stops the program (by triggering the motion sensor) at the appropriate time. How did the actual number compare to the estimation?

Lesson 5 Vocabulary:
Direction – the way something is pointing
Motion – the state when something is moving
Power – the speed at which a motor moves
Sensor – a machine that can tell something that is happening around it
Tilt – how off-center something is
Vision – the sense used by the eyes

Language Arts Connections: *Sensor Walk*

Divide the class into two groups: Humans and Robots. Take the class for a walk around the school or neighborhood. Have students keep a list of all the different things they can sense and what part they used to sense it. For example, children in the human group could sense the sunlight with their eyes while students in the robot group would sense this with their light sensor. Children in the robot group do not *need to be limited WeDo™ sensors, but can think creatively about all kinds of sensors a robot might have. Upon returning to the classroom, compare and contrast the Human and Robot lists. Are there some things humans can sense but robots cannot? What about vice versa?*

Sensors Experts: Children who finish programming their robots and master all concepts quickly get to wear a badge that says "Sensors Expert". Sensors Experts walk around and offer help troubleshooting to any classmates experiencing difficulties attaching their sensors or programming with sensors.

Collaboration Web: As children progress through the lesson, they will complete their collaboration webs. They will draw lines from their picture to the pictures of any classmates who give them help. If children say they didn't receive any help, remind them to think of their partners, class Experts, or if they got any ideas by looking at another classmate's project.

Technology Circle: Understanding how the sensors work and how programming them can be challenging. Have students discuss their understandings of the sensors and why different programs do or do not accomplish the goal.

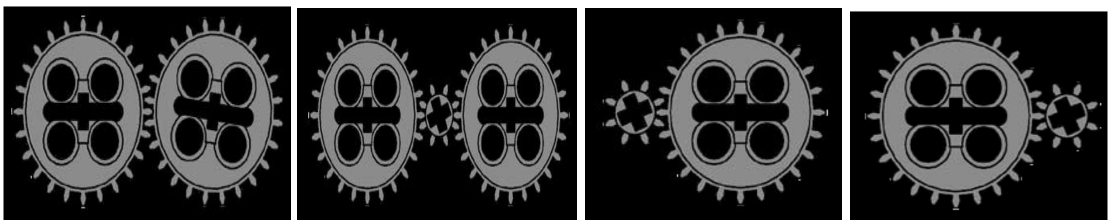
Concluding Activity: Song or game. See Appendix B for examples.

Free-play Have students explore adding sensors to robots and making programs with Repeats and Sensor Parameter instructions. Sensors use complex concepts and often work in unexpected ways. Offer support in observing the robot's behavior so students may fully understand these concepts.

Lesson 6
What Are Gears?

<u>Overview</u>*: In this lesson, children will be introduced to the concept of gears. Children will build models demonstrating gears, gearing up, gearing down, and an idler gear. This lesson can be used to directly lead into an advanced final project that utilizes gears (like the WeDo™ Car, see Appendix H) or it can be used simply as an introduction to the complicated concept of gears.*

Examples of gears, idler gear, gearing down, and gearing up (respectively):



Prior Knowledge	***Objectives***	
	Students will understand that…	*Students will be able to…*
• *Real world items, like bicycles, function by utilizing gears* • *Gears can be used to make something move.*	• *Gears are mechanical parts with cut teeth of such form, size, and spacing that they mesh with teeth in another part to transmit or receive force and motion.*	• *Build and explain models using gears, gearing up, gearing down, and an idler gear*

Basic Properties of Gears

1) When 2 gears mesh, the driver makes follower turn in opposite direction
2) You need odd number of idler gears to make driver and follower turn in same direction.
3) When large driver turns small follower, it's called ***gearing up*** and speeds up gear train
4) When small driver turns large follower, it's called ***gearing down*** and increases torque (turning force)

Materials / resources:

- LEGO® WeDo™ gears found in the WeDo™ construction sets
- Pictures of real world gears kids are familiar (e.g. gears on a bicycle)
- Pictures of idler gears, gearing down, and gearing up
- Engineering Design Journals for planning and reflecting

Activity description

Warm- up : Real World Gears

Show videos and/or tangible examples of gears and how they work.

Videos on Gears:

Small engine animation: http://www.youtube.com/watch?v=_GVq9O9mFCs&feature=related

Watch movement animation: http://www.youtube.com/watch?v=qux3fLCf2FU&feature=related

Inside the watch and movement: http://www.youtube.com/watch?v=cn6lHIhm5Xs

Steam engine train: http://www.youtube.com/watch?v=fhvmC0Qpkk0

Introduce the concepts and the task (10 minutes): "Today we will be building models of different types of gears. We might choose to use some of the models we make today in our final projects." Show examples of the different types of WeDo™ gears and the ways you can program with them. Introduce the basic properties of gears.

What Are Gears?
Gears are a mechanical part with cut teeth of such form, size, and spacing that they mesh with teeth in another part to transmit or receive force and motion.

Math Connection: *Speed and Direction*

The Math Behind Gears

Complete a worksheet on gears (see Engineering Design Journal in Appendix D). What direction are the gears shown turning? Which gears are going faster? Which are going slower?

Lesson 6 Vocabulary:
Crown – a gear shaped like a crown that meshes at an angle, not in a straight line like most gears
Gear – something used to turn many parts using only one motor
Gear Down - using differently sized gears to decrease the speed of a machine
Gear Up – using differently sized gears to increase the speed of a machine
Idler – a gear that goes in between two other gears that isn't connected to the motor or a wheel
Mesh – two gears that connect
Tooth – the part of a gear that turns other gears

Language Arts Connection: *Human Gears*

Get into small groups of around 5 children. You will be creating a system of gears. Children should stand shoulder to shoulder. The first child will turn in one direction gently pushing the person next to him in the opposite direction and so on down the line. Using communication skills and new vocabulary describe what you are doing as you move Once you have a whole line of gears moving, try passing an object (like a ball) down the line of "gears" (children). How did it go? Share with the class.

Individual / pair work : Individually or with groups, build and program models of gears, idler gears, gearing down, and gearing up.

Note: Teachers may wish to use or consult the Getting Started activities on gears, idler gears, gearing up and gearing down available with WeDo™ . To find these activities, open WeDo™ , click on "Getting Started", and then click on "Gears".

Gears Experts: Children who finish building and programming all of the gear constructions assigned and master all concepts quickly get to wear a badge that says "Gears Expert". Gears Experts walk around and offer help troubleshooting to any classmates experiencing difficulties working with the different kinds of gears. Encourage experts to use the proper gear vocabulary when helping their classmates.

Collaboration Web: As children progress through the lesson, they will complete their collaboration webs. They will draw lines from their picture to the pictures of any classmates who give them help. If children say they didn't receive any help, remind them to think of their partners, class Experts, or if they got any ideas by looking at another classmate's project.

Technology Circle: After building, students share their creations and discuss the challenges they encountered programming with gears.

Free-play: Provide opportunities for children to build freely using gears or to begin work on a final project that utilizes gears.

Lesson 7
Final Projects

<u>*Overview*</u>

This project should be tailored to fit with a curriculum unit, project, or event happening in the classroom so that it meets the goals of the teachers and the interests of the students and teachers. Students work together to build and program a robot to demonstrate their understandings and ideas related to the robotics and programming curriculum as well as the content of the project theme or topic. During the course of the final project, students put to use all the concepts learned during the previous lessons but transfer them to a new context. When possible, teachers should encourage the use of crafts and recycled materials.

Language Arts Connection:

-***Invitations***: *Write out and mail invitations to your family inviting them to come to your final project showcase. Add illustrations and information describing your project.*

-***How-To Book:*** *Create a comprehensive How-To Book describing how to build and program the robot you made*

Math Connection:

-***How Many?:*** *As a class, keep a chart that graphs how many of all the different types of robotic and non-robotic parts you used. Make a report to display and share on the presentation day.*

Playground Final Project Ideas:

-Merry-go-round

-Ferris Wheel or other ride (See Appendix F for building instructions)

-School bus or ice cream truck (See Appendix F for car building instructions)

-Soccer/basketball/baseball player

- A playground game like tag or hopscotch

Individual/pair work: Children will work individually or with a partner to plan, design, build, and program a final project from scratch. Children will be encouraged to use advanced topics such as sensors and repeats when programming their robots.

Presentations: Students share:

a. the robot they made,
b. why they chose the features they did for their robot,
c. the goal of their program and why they wanted it to do that / what it represents,
d. the final program they built, and
e. anything that was hard, easy, surprising, interesting, etc about the process.

Materials / resources:

- Large icons for games and reference displays
- Robotic parts for each child or pair to make a robot, plus extras
- Crafts and recycles materials for robots and for building an environment for them to run in
- Computers with WeDo™ software,
- Design journals, small icons for cutting and taping/gluing in the design journals

Appendix A

Robotics across Themes

Robotics across Themes

This robotics and programming curriculum can be used within the context of study on a wide variety of topics. The challenges presented in this curriculum relate to the theme of the playground, but the basic ideas can be reconfigured to make sense with many other topics commonly studied in early childhood settings. Below are some suggestions for such contexts. These ideas may also help children focus in choosing what kind of robot to make if the curriculum uses an open-ended theme such as "communities."

- Person who lives in the community
- Person who works in the community
- Responsive or interactive building or structure
- Vehicles and road-related structures (traffic lights, drawbridges, etc)
- Nature (plants, animals, landscape, weather)
- Safety (alarms, crossing signals)
- Places and structures for entertainment, fun, or commerce
- Basic needs (food, housing)

The activities in this curriculum can easily be adapted to fit other themes. Use your imagination to find a story context for each powerful idea. For instance, if this curriculum were to accompany a unit on the study of animals, the activities might look like these:

Lesson 1: (What is the Engineering Design Process?) Build a sturdy animal that can move like its real-life counterpart.

Lesson2: (What Is a Robot?) Build a robotic animal. (It can have wheels like a vehicle and use crafts or recycled materials to give it the appearance of the chosen animal.)

Lesson 3: (What is Programming?) This song (or another of your choosing) is a fun and concrete way to start the unit regardless of the theme.

Lesson 4: (What are Repeats?) Program animals (maybe at a zoo) to visit each other along different paths.

Lesson 5: (What are Sensors?) Animals have many senses just like people do, so this activity can be adapted in all sorts of ways. As examples, animals might do an action until (or when) someone pets them (touch sensor) or when someone gives them food (the food might cover a light sensor to make it dark).

Appendix B

Songs and Games

Songs and Games

Many common songs and games can be used to support children's understandings of robots and programming concepts. For instance, Simon Says is a way to internalize instructions and understand them in a more complete way, both kinesthetically and verbally. Here are some other suggestions for songs and games to reinforce various concepts from the curriculum. Teachers may think of many more!

- Simon Says, traditional style: emphasizes ways our own bodies move, but without having kids sit out for mistakes,
- Simon Says w/ icons cards: helps students learn new programming icons' symbols, spoken name, and kinesthetic action. Variation: Kids pick icons/strings for peers to act out,Head, Shoulders, Knees, and Toes: emphasizes peoples' body parts vs. robot parts,
- Act out blocks and programs or 'program' a friend to move along a line on the floor,
- 'Memory' card game or other matching game with icons: spurs use of instruction icons' names. When a child finds a match, they name and act out the icon.
- 'The Wheels on the bus.' Variation: Sing with programming instructions,
- Programming Charades: Mentor shows a child a program made from block icons. The child acts out the program. The other children identify what icons made up the program.
- Walk-Through Progams. Make large programming icons that can be placed on the floor. Children literally walk through the program step by step and carry out the actions to internalize how the robot processes its programs. This can be especially helpful when working with "Repeats" and "Ifs."

Appendix C

The Engineering Design Process

The Engineering Design Process

When working with young children and robotics, there are some interesting challenges around helping children structure their problem-solving processes. Marina Bers, in her Blocks to Robots' book (2008), talks about the role of the engineering design process.

> "On the one hand, we want to help them [the children] follow their ideas, but we do not want them to become frustrated to the point they quit the work. On the other hand, we do not want their success to be scripted, too easy, or without failure. One of the approaches for how to handle this is by helping them understand and follow the design process. This is similar to what engineers or software developers do in their own work. They identify a problem. They do research to understand better the problem and to address it. They brainstorm different potential solutions and evaluate the pros and cons. They choose the best possible solution and plan in advance how to implement it. They create a prototype and they implement it. They test it and redesign it based on feedback. This happens many, many times. And finally, they share their solutions with others. This cycle is repeated multiple times."

The following diagram, Figure 1, shows one of many possible simplified versions of the engineering design process. One suggestion for using this graphic is to give each child or pair a small copy of it along with a token, similar to a playing piece in a board game. Children can move their token around the diagram to reinforce the steps of the process. Figures 2-5 show individual steps of the engineering design process to facilitate making a large poster of the whole process for the class to see from anywhere around the room.

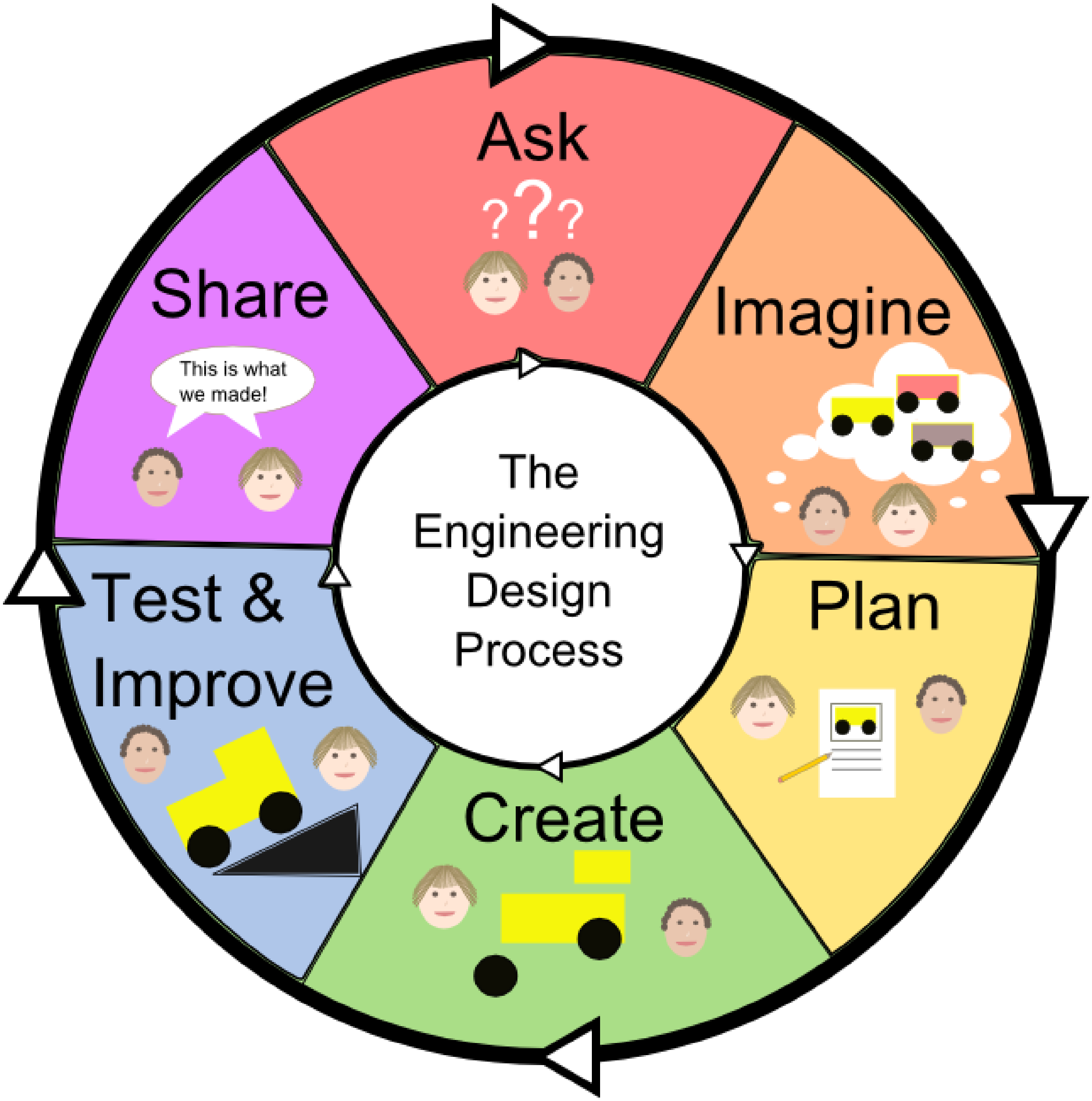


Figure 1: Simplified steps involved in the engineering design process.

Figure 2: Engineering Design Process step 1: Ask a question about a problem you want to solve or a goal you want to accomplish.

Figure 3: Engineering Design Process step 2: Imagine as many different ways to accomplish your goal or answer your question as you can.

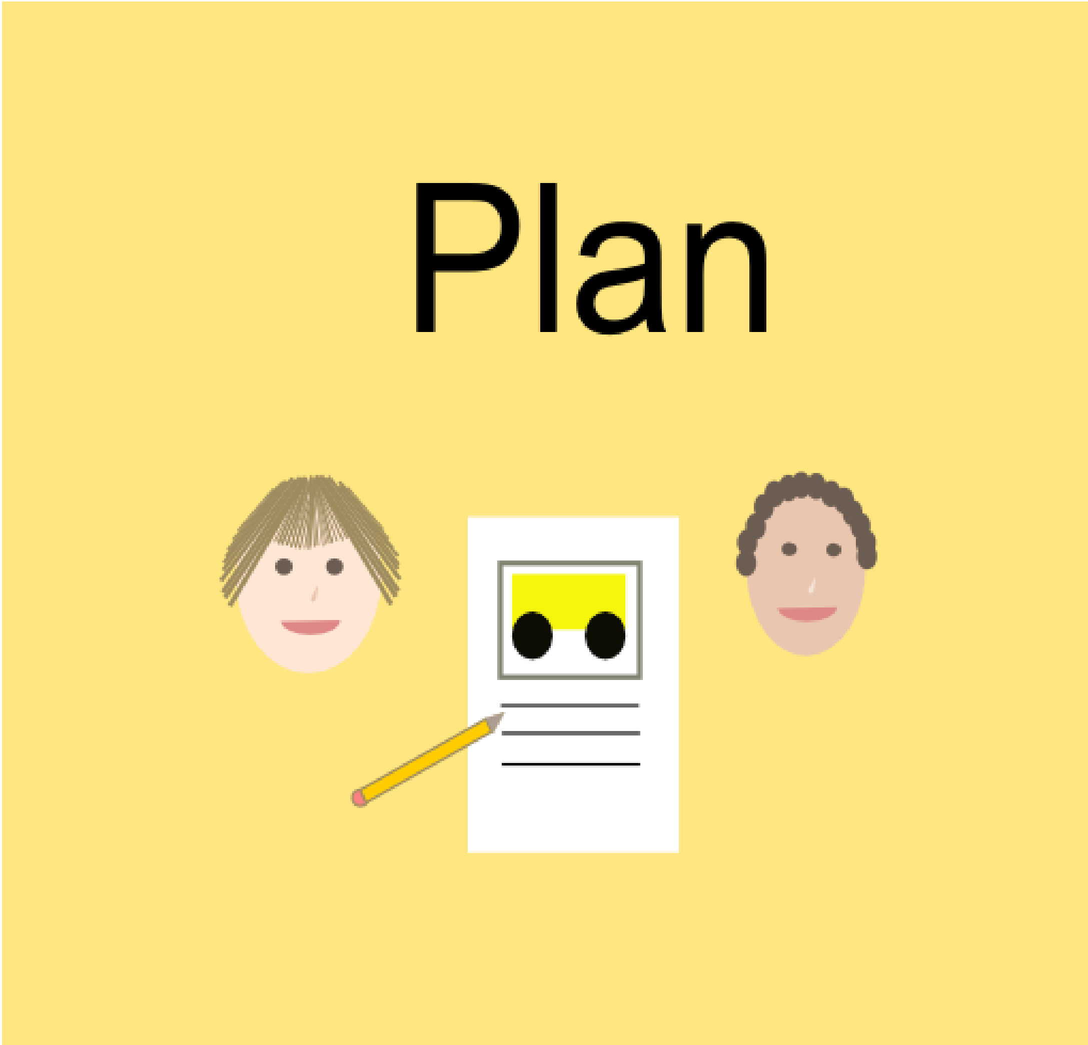


Figure 4: Engineering Design Process step 3: Choose one solution and plan out how to do it in detail.

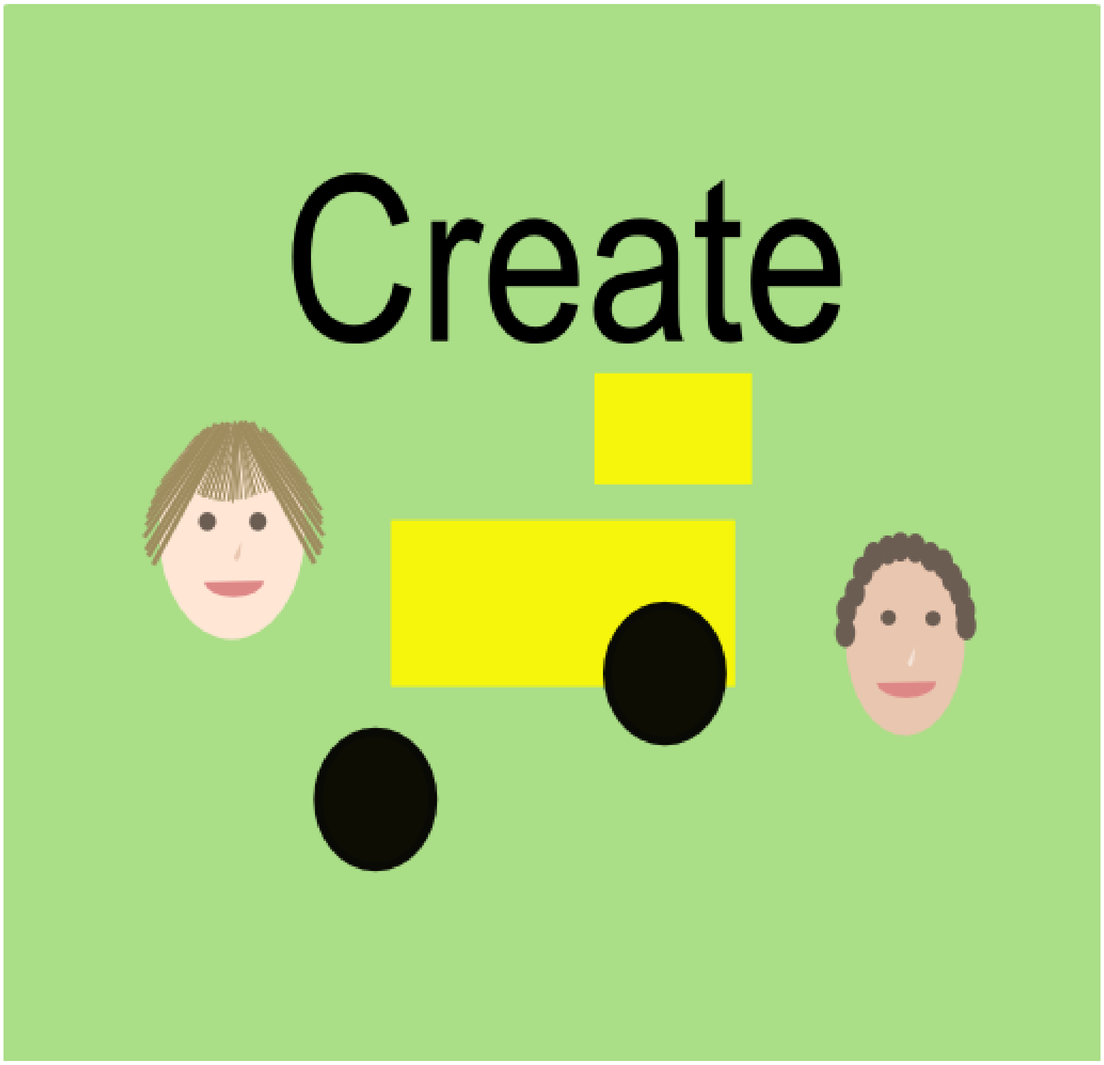


Figure 5: Engineering Design Process step 4: Create a prototype or working version of your plan.

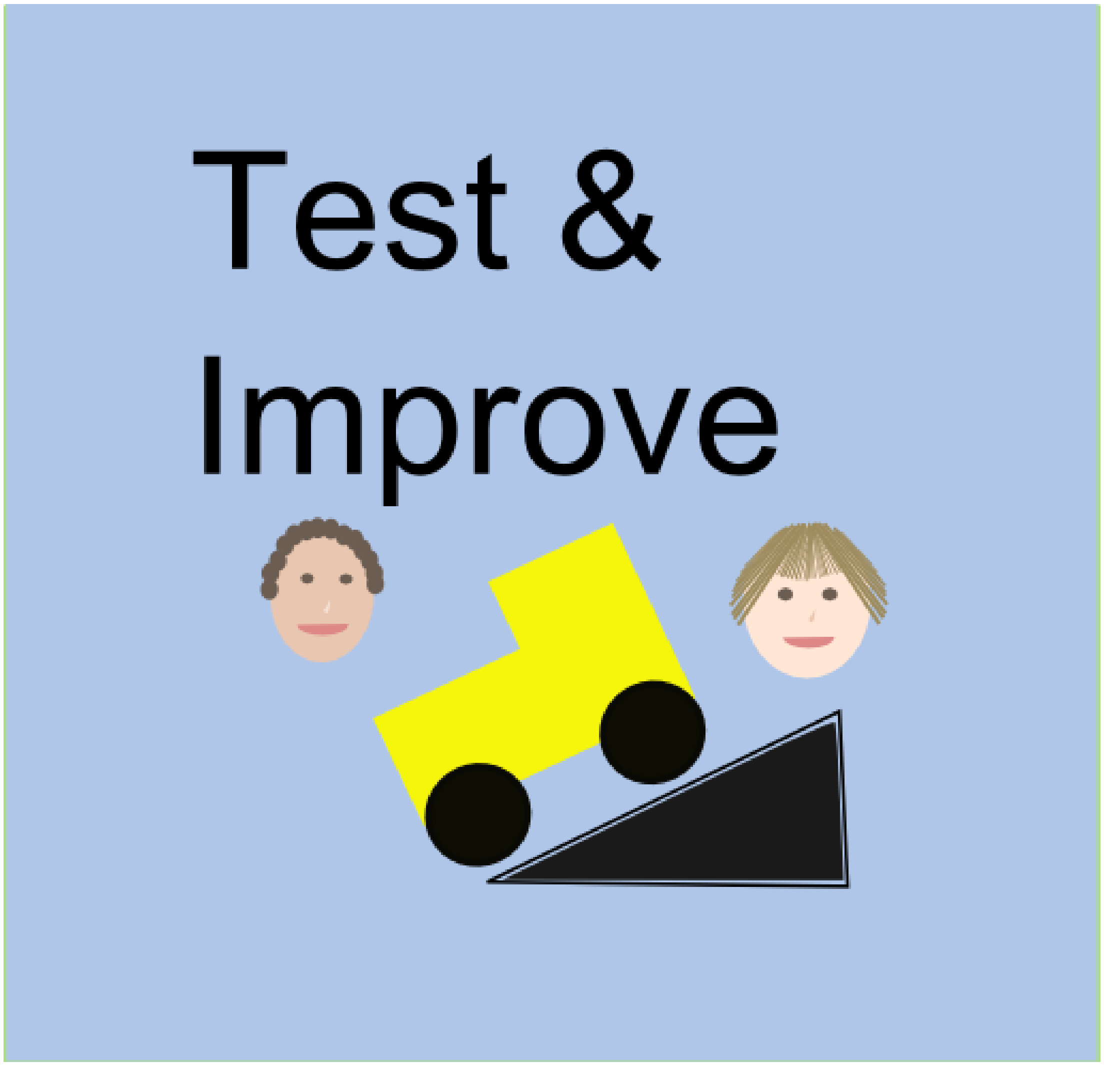


Figure 6: Engineering Design Process step 5: Test your creation to see how well it accomplishes the goals you have for it. Try different ways to improve it and test whether the improvements work better.

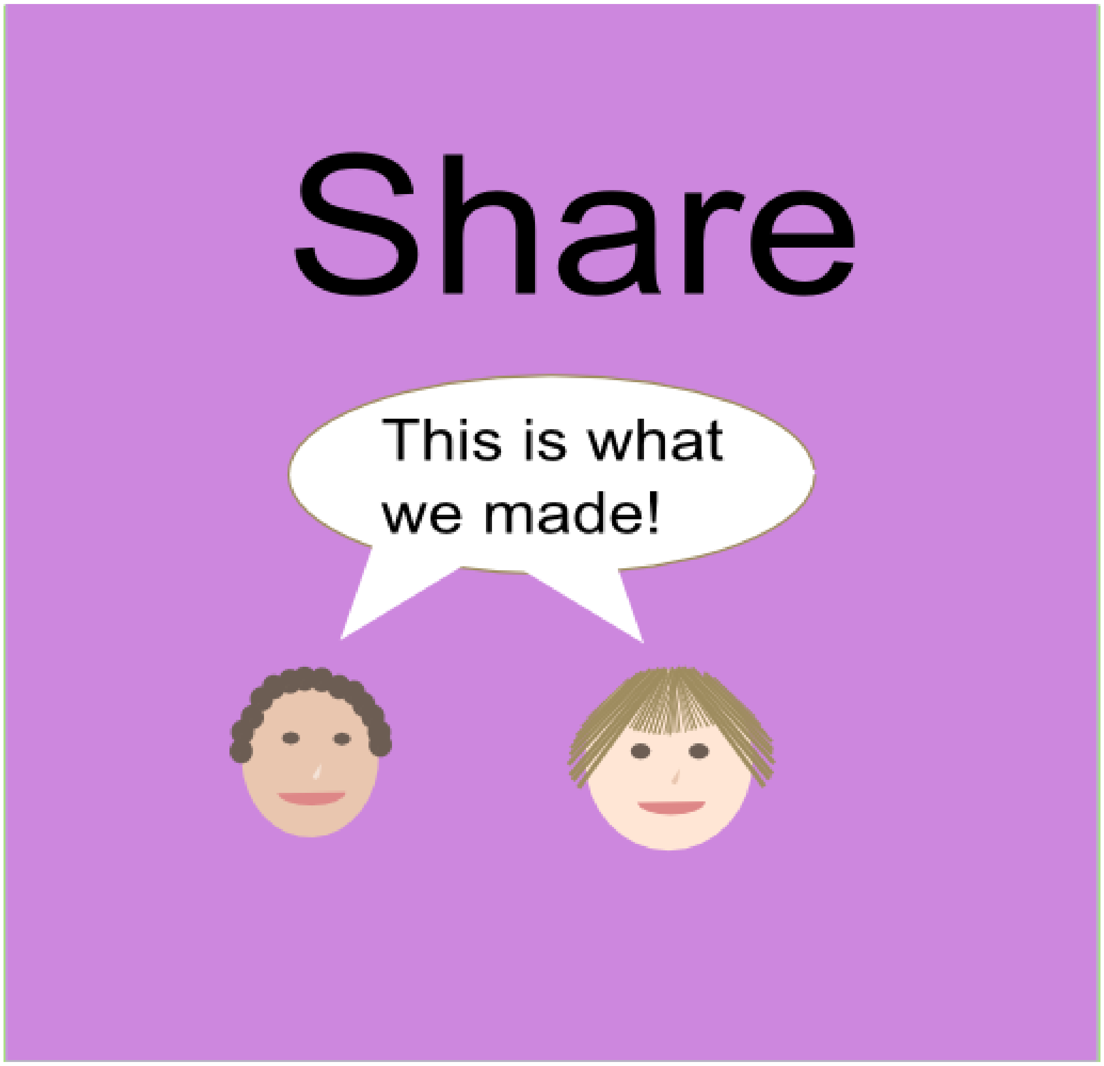


Figure 7: Engineering Design Process step 6: Share what you have done and get feedback.

Appendix D

Design Journals

Design Journals

Providing children with a design journal and with many opportunities to talk about their ideas throughout the process can be helpful. However, before working with design journals it is useful to be aware of different approaches to the design and problem-solving processes. Following is an excerpt from Marina Bers' book Blocks to Robots.

> "As children work on their projects, many iterations and revisions will be done. Design journals make transparent to the children themselves, as well as teachers and parents, their own thinking and the project evolution. […] Some children might choose to avoid using design journals or follow a systematic design process. They do not like to plan in advance. They might belong to a group of learners that Papert and Turkle have characterized as tinkerers and bricoleurs (Turkle & Papert, 1992). They engage in dialogues and negotiations with the technology, their ideas happen as they design, build and program. As Papert and Turkle write, *"The bricoleur resembles the painter who stands back between brushstrokes, looks at the canvas, and only after this contemplation, decides what to do next"* (Turkle & Papert, 1992).

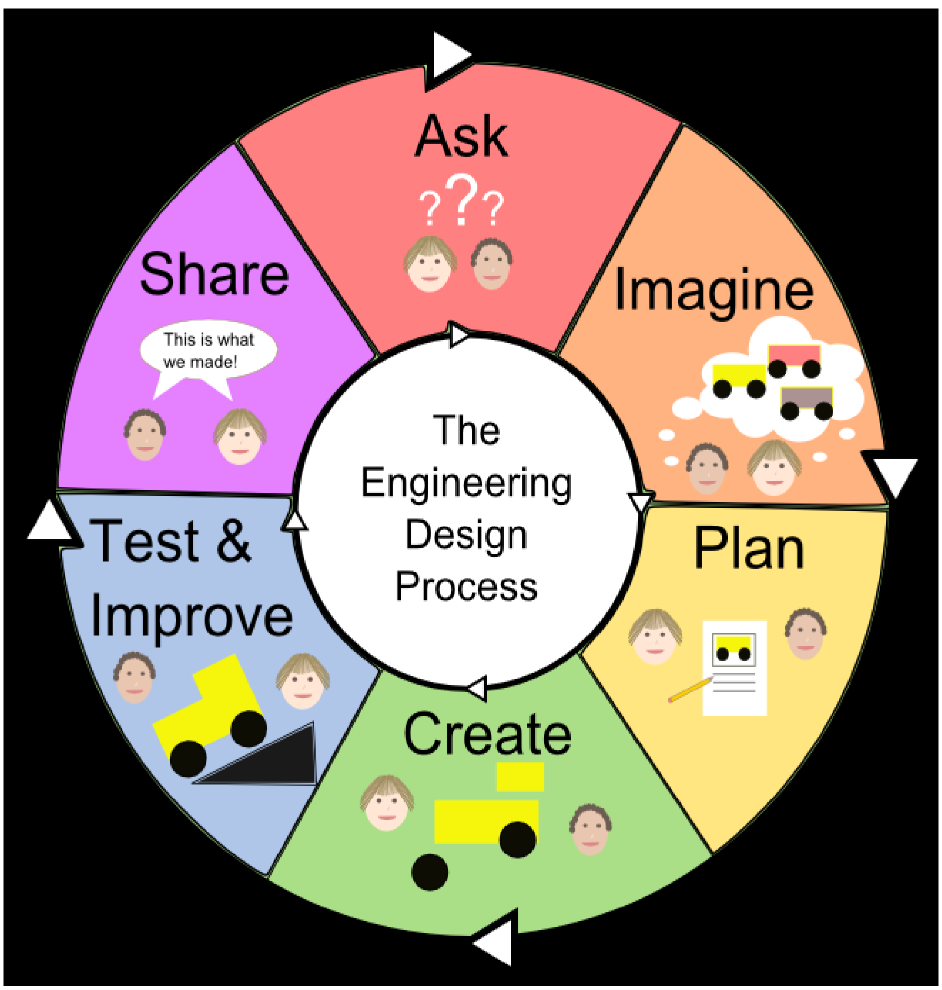
> "Constructionist learning environments allow for different epistemological styles, or ways of knowing, to flourish. Some children want and need constraints and top-down planning because they know what they want to make. Others enjoy working bottom-up and messing around with the materials to come up with ideas. Some methods of teaching robotics and programming, directly derived from engineering and computer sciences, provide structured paths for children to navigate the process from idea to product. For example, the formal steps of the engineering design process presented earlier are laid out in a design journal consisting of teacher made worksheets. This approach might or might not work, depending on the child, the way the learning environment is set up and the educational goals. In this book I advocate both pathways: design journals with a directive focus, in the forms of questions and design journals with lots of white pages, for those children that might want to invent their own strategies. Tinkerers and planners complement each other and can also learn from each other. Constructionist environments should be inviting and supportive to little engineers who thrive working with constraints and making advanced plans, and little tinkerers who create in dialogue with the materials."

Taking these ideas into consideration, a sample design journal (with math and language arts activities tailored to PreK-Kindergarten children) begins on the next page. This journal is a template that can be printed and used directly by teachers or modified to fit their specific goals. Additionally, activities are provided for the advanced lessons (sensors and gears) and should be taken out of design journals for children not pursuing these lessons. Please see the sample Engineering Design Journal on the following page.

Sample Engineering Design Journal

NAME:____________________

MY ENGINEERING DESIGN JOURNAL



AT THE PLAYGROUND

One structure I picked was the ______________________.

It looks like this:

It has these shapes:

The other structure I picked was the ____________________.

It looks like this:

It has these shapes:

BUILDING PLAYGROUND EQUIPMENT

I'M BUILDING A:____

HERE IS A PICTURE OF WHAT IT WILL LOOK LIKE:

TO BUILD IT, I WILL USE THESE SHAPES:

MY VISIT TO THE PLAYGROUND

Choose a friend or family and write them a postcard describing what you did today at the playground. Cut out this postcard and draw a picture of your playground visit on the back. Then write the name and address of the person you want to send it to, put on a stamp, and send it!

Try to use some of these words:

Artifact	Design	Material	Square
Circle	Edge	Rectangle	Structure
Cycle	Engineer		

Dear ____________________,

PLACE
STAMP
HERE

BUILDING SWINGS

HERE IS A PICTURE OF MY SWINGS:

HERE ARE THE SHAPES I WILL USE TO BUILD IT:

I USED THESE PIECES TO BUILD MY SWING:

PIECE	HOW MANY:

TOTAL NUMBER OF PIECES:____________________

I USED THESE COLORS TO BUILD MY SWING:

COLOR	HOW MANY

I USED THESE SHAPES TO BUILD MY SWING:

SHAPE	HOW MANY

HOW TO BUILD A ROBOTIC SWING

Here's how I built my robotic swing:

Words to Use		
Automatic	Function	Motor
Axle	Hub	Robot
Computer	Joint	Wires

HOW TO BUILD A ROBOTIC SWING (page 2)

This is what it looks like:

SEESAW

HERE IS A PICTURE OF MY SEESAW:

HERE ARE THE SHAPES I WILL USE TO BUILD IT:

HERE IS MY PROGRAM TO MAKE THE SEESAW MOVE:

I USED THESE PIECES TO BUILD MY SEESAW:

PIECE	HOW MANY:

TOTAL NUMBER OF PIECES:______________________

I USED THESE COLORS TO BUILD MY SEESAW:

COLOR	HOW MANY

I USED THESE SHAPES TO BUILD MY SEESAW:

SHAPE	HOW MANY

BRUSHING MY TEETH

Here's my program for brushing my teeth:

SENSORS

Here is the program for my seesaw using a motion sensor:

Try #1

My seesaw went back and forth ______ times.

Try #2

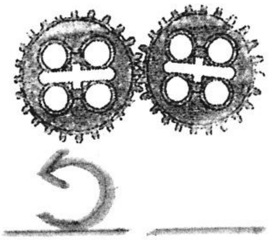
My seesaw went back and forth ______ times.

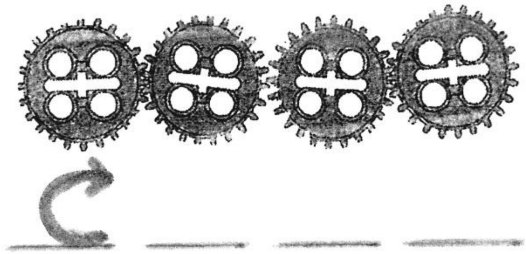
Try #3

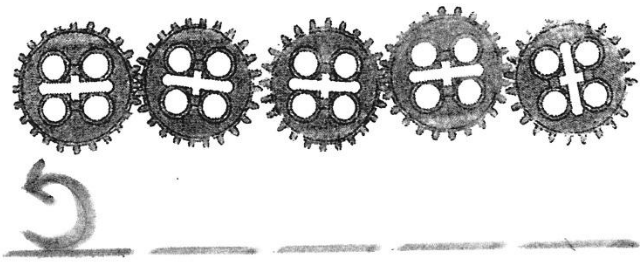
My seesaw went back and forth ______ times.

GEARS

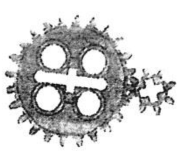
Look at the following gears. Which direction are they turning?

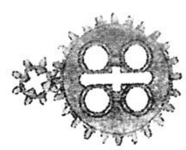


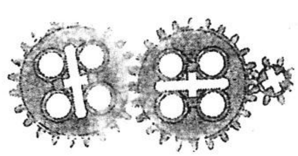




Look at the following gears. Is the last gear spinning faster or slower than the first one?







MY FINAL PROJECT

FOR MY FINAL PROJECT I WILL BUILD:

____

I PICKED THIS BECAUSE:

____

____

____

IT WILL LOOK LIKE THIS:

I WILL USE THESE SHAPES TO BUILD IT:

HERE IS MY PROGRAM:

I USED THESE PIECES TO BUILD MY FINAL PROJECT:

PIECE	HOW MANY:

I USED THESE COLORS TO BUILD MY FINAL PROJECT:

COLOR	HOW MANY

I USED THESE SHAPES TO BUILD MY FINAL PROJECT:

SHAPE	HOW MANY

MY KIT HAS THESE PIECES

PIECE	HOW MANY	PIECE	HOW MANY

PIECE	HOW MANY	PIECE	HOW MANY

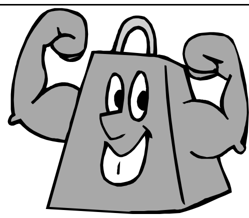
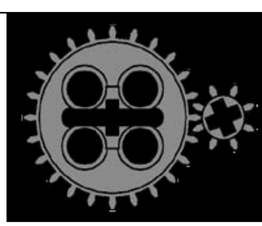
Appendix E
Engineer's License

Figure 8: Sample Engineer's License (ready to print and use)

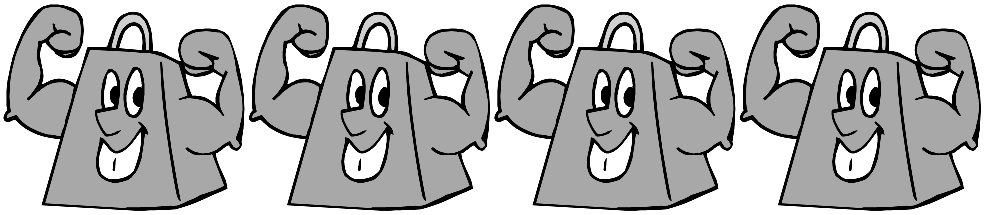
___________'s Engineer's License	
Level 1 – Sturdy Builder	Level 2 – Robot Builder
Level 3 – Programmer I	Level 4 – Programmer II
Level 5 – Programmer III	Level 6 – Gear Expert
Level 7 - Expert	

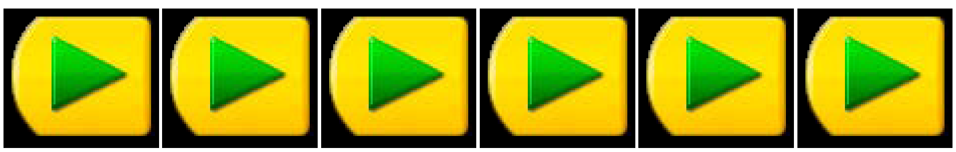
Figure 9: Key to the icons on the sample Engineer's License

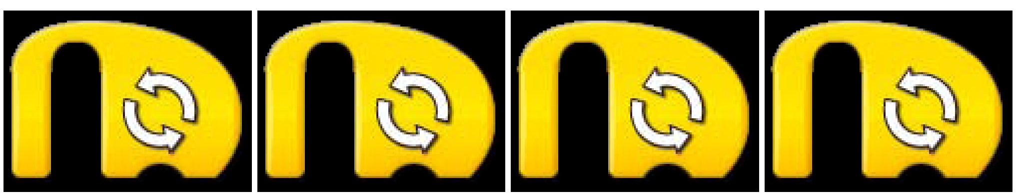
<u>Engineer's License: Key</u>

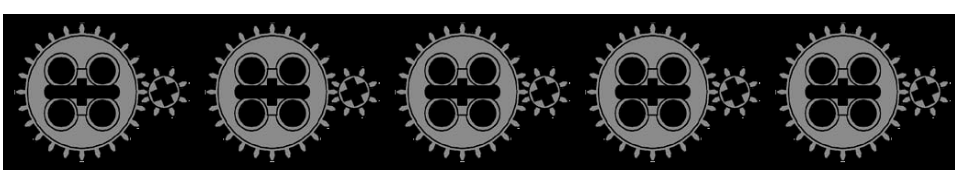
<table>
<tr><th colspan="2">Level 1: Builder</th></tr>
<tr><td></td><td>Robot stays intact.</td></tr>
<tr><th colspan="2">Level 2: Robot Builder</th></tr>
<tr><td></td><td>Robot has all attached parts. Robot is able to move.</td></tr>
<tr><th colspan="2">Level 3: Programmer I</th></tr>
<tr><td></td><td>Child picks right programming icons and puts icons in the right order</td></tr>
<tr><th colspan="2">Level 4: Programmer II</th></tr>
<tr><td></td><td>Child knows when and how to use repeats.</td></tr>
<tr><th colspan="2">Level 5: Programmer III</th></tr>
<tr><td> </td><td>Child understands sensors and how to program them.</td></tr>
<tr><th colspan="2">Level 6: Gear Expert</th></tr>
<tr><td></td><td>Child understands gears and how to use them.</td></tr>
<tr><th colspan="2">Level 7: Expert</th></tr>
<tr><td></td><td>See criteria for assessing final projects and overall levels of understandings.</td></tr>
</table>

Icons to print as stickers or cut-outs for Engineer’s Licenses:









Appendix F

WeDo™ Resources

Where Can I Get a WeDo™ Kit?

LEGO® WeDo™ robotics construction sets are available for purchase at:

http://www.LEGOeducation.us/eng/product/LEGO_education_WeDo_robotics_construction_set/2096

Where Can I Find Additional Teaching Resources?

1) The WeDo™ Basic Programming Guide is available online at:

http://aux.LEGOeducation.us/sharedimages/resources/WeDo%20Basic%20Programs.pdf

2) The LEGO® Engineering Website:

http://www.LEGOengineering.com/

3) The WeDo™ User's Guide available for download here:

http://community.LEGOeducation.us/media/p/8288.aspx

4) The TK Robotics Network Developed by the DevTech Research Group:

http://tkroboticsnetwork.ning.com/

Where Can I Find Additional Pedagogical Resources?

A list of relevant publications can be found here:

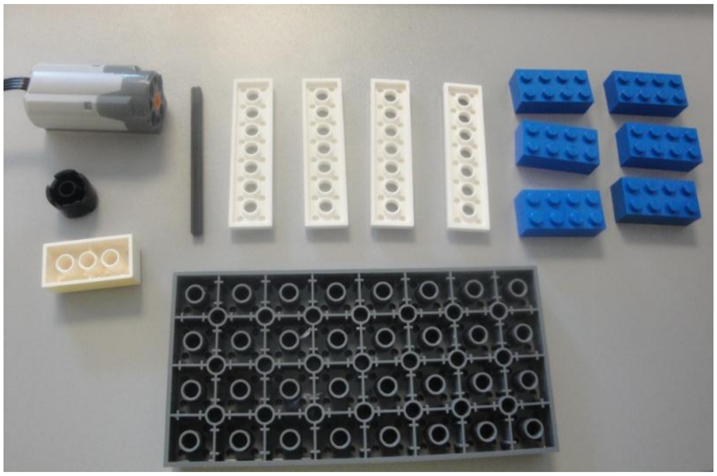
http://ase.tufts.edu/DevTech/publications/

Prof. Ber's book *Block to Robots* is available for purchase here:

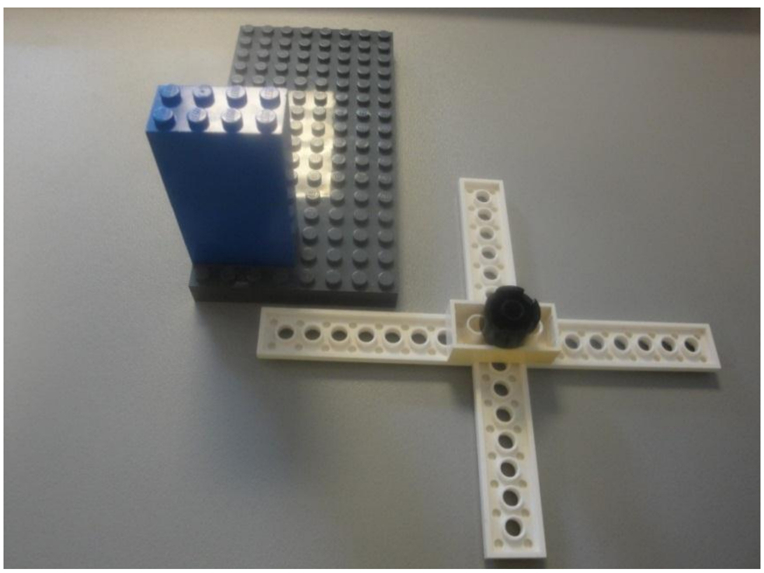
http://www.amazon.com/Blocks-Robots-Technology-Childhood-Classroom/dp/0807748471

Ferris Wheel

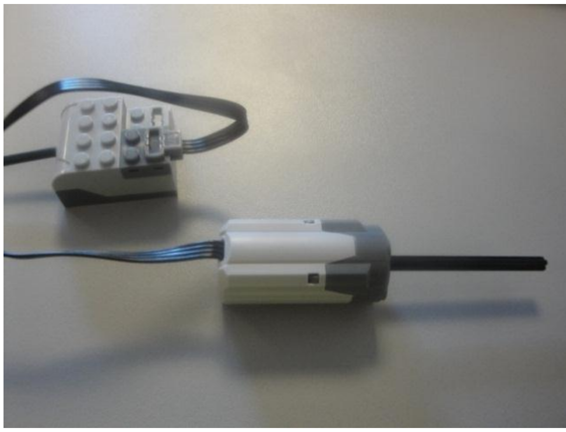
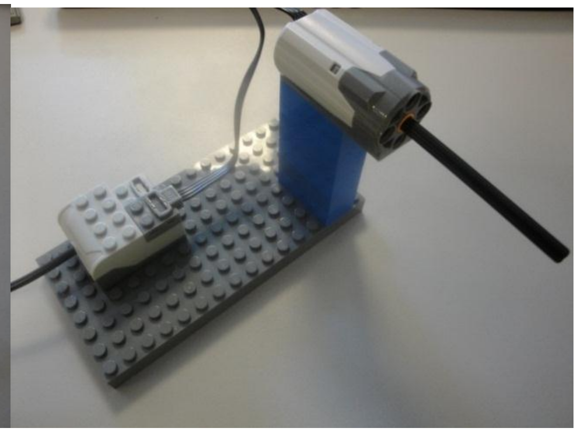
This Ferris wheel is made entirely out of parts from the LEGO® WeDo™ kit. It only requires one motor, but different variations can be implemented. Children may also decide to get creative and decorate the Ferris wheel with non-LEGO® pieces, such as arts and crafts materials.



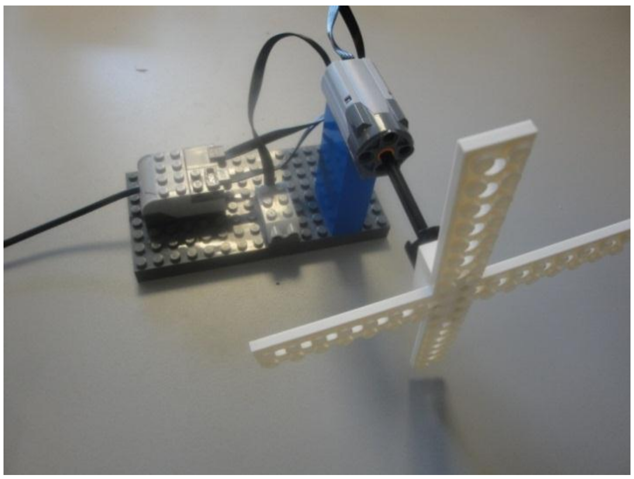
Step 1a: Parts for the Ferris wheel, axle, and motor.



Step 1b: Attach parts as shown

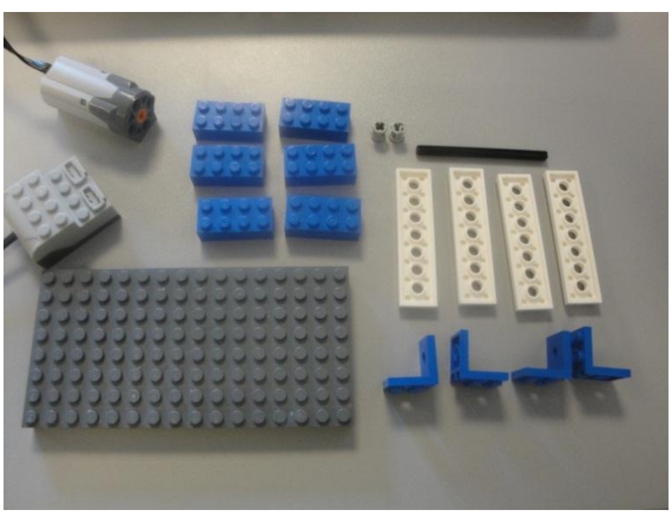



Step 2: Connect an axle to the motor, and attach the motor wire to a USB hub.

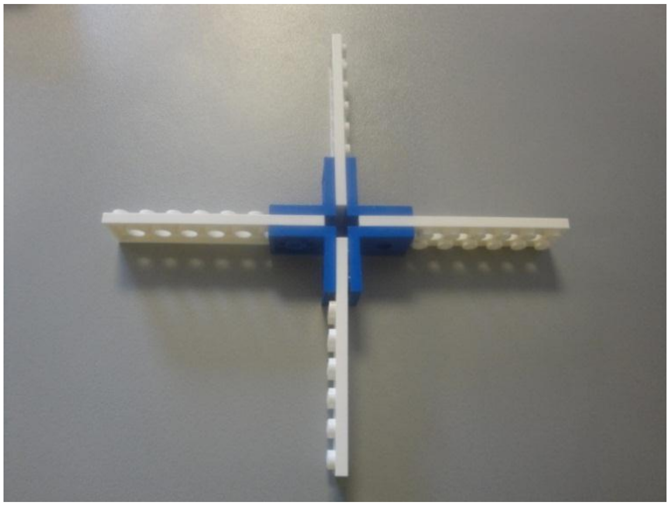


Step 3: Attach Ferris wheel to end of axle. You can also add a sensor to the structure.

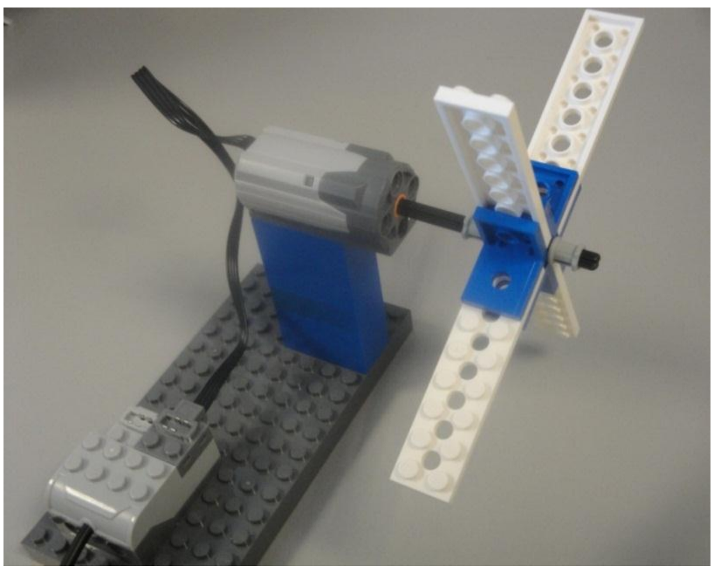
Variation: Using 4 right angle LEGO® pieces (not included in WeDo™ kit) to create Ferris wheel.



Step 1a: Parts for the variation of Ferris wheel.



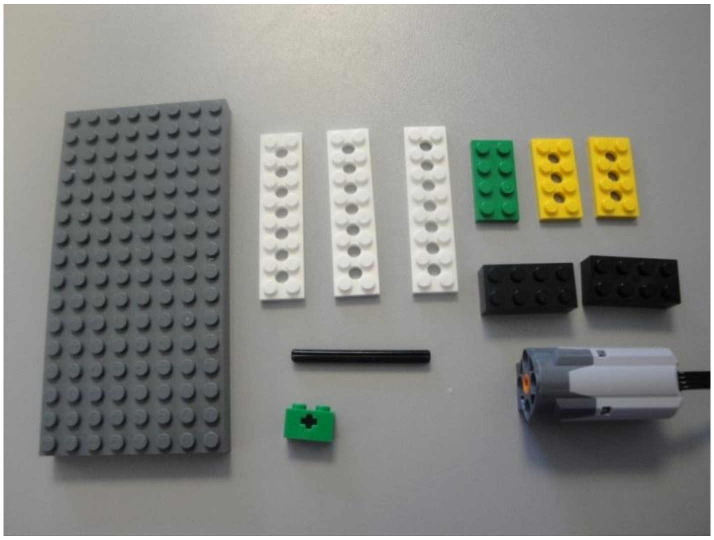
Step 1b: Attach parts as shown



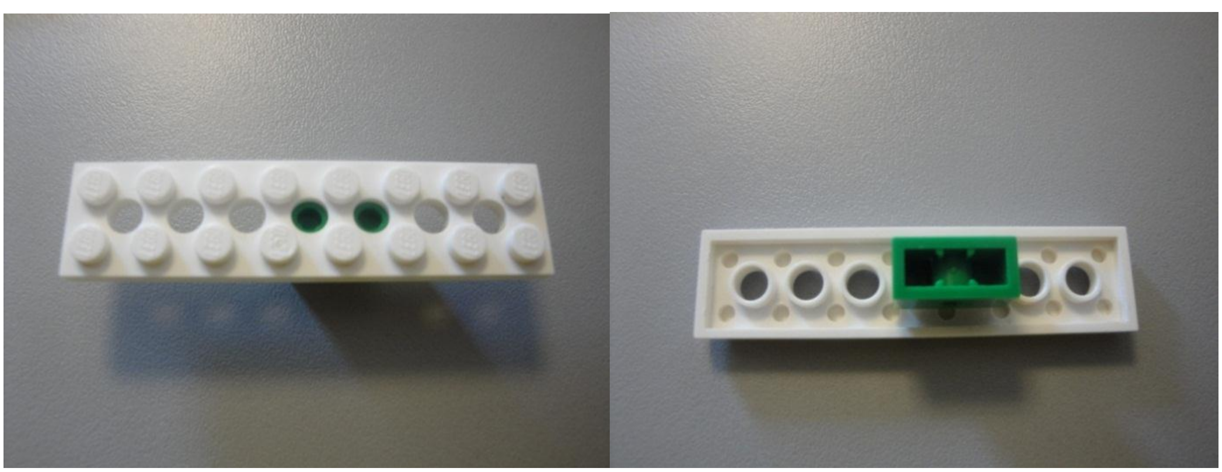
Step 2: Connect moving parts and attach Ferris wheel to axle. Use two bushings to snuggly secure the wheel and ensure that it will spin with the axle.

Seesaw

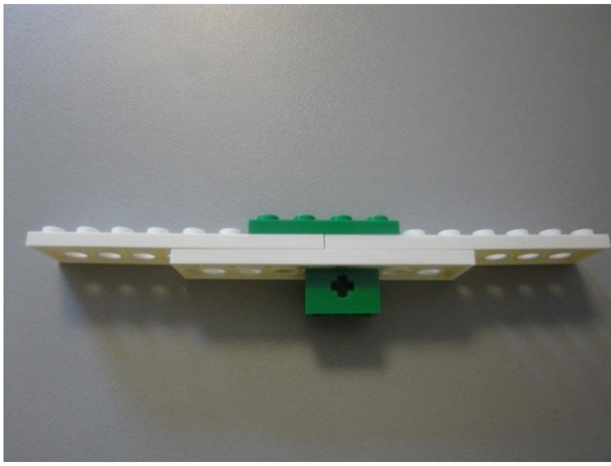
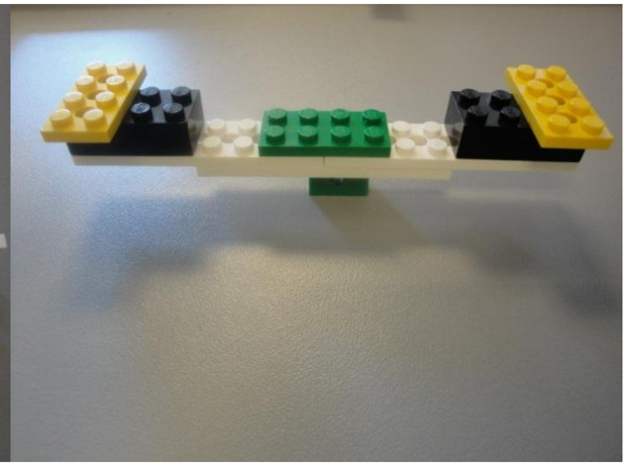
This seesaw is made entirely from the LEGO® WeDo™ kit. Just as with the Ferris wheel, many variations can be built. One of the challenges is ensuring that the seesaw is sturdy and will not break after a few turns of the axle. The instructions for the seesaw below will provide a sturdy seesaw.



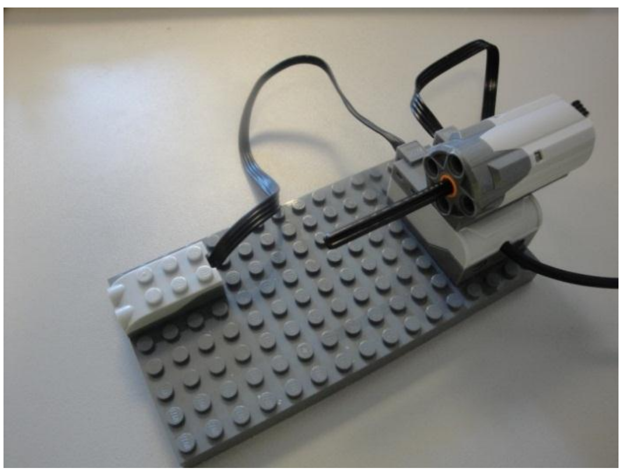
Step 1: Parts for the seesaw, axle, and motor.



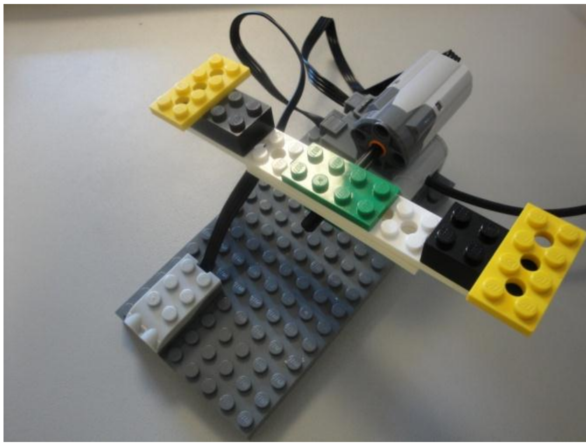
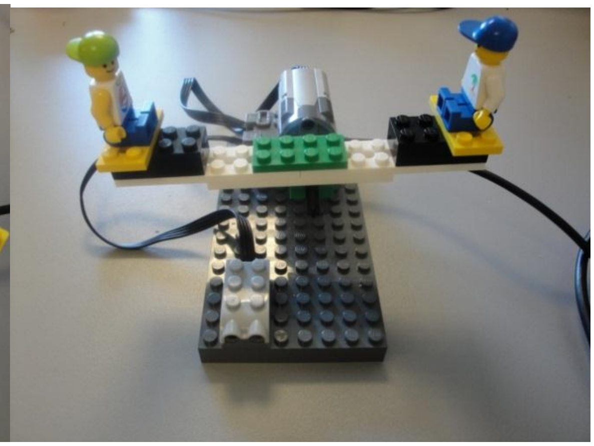
Step 2: Attach 2x1 LEGO® to flat piece as depicted. Make sure that the two studs are locked into the holes of the flat LEGO® piece for the sturdiest hold.

Step 3: Continue to construct the seesaw with more flat pieces and attach a smaller flat piece to secure the flat pieces together. Additional bricks can be added to provide more seesaw functionality.



Step 4: Attach an axle to the motor, and connect the motor wire to a USB hub. Then attach the motor to the hub, which will act as the base of the seesaw. Sensors can be added to the base of the structure or directly on the seesaw.

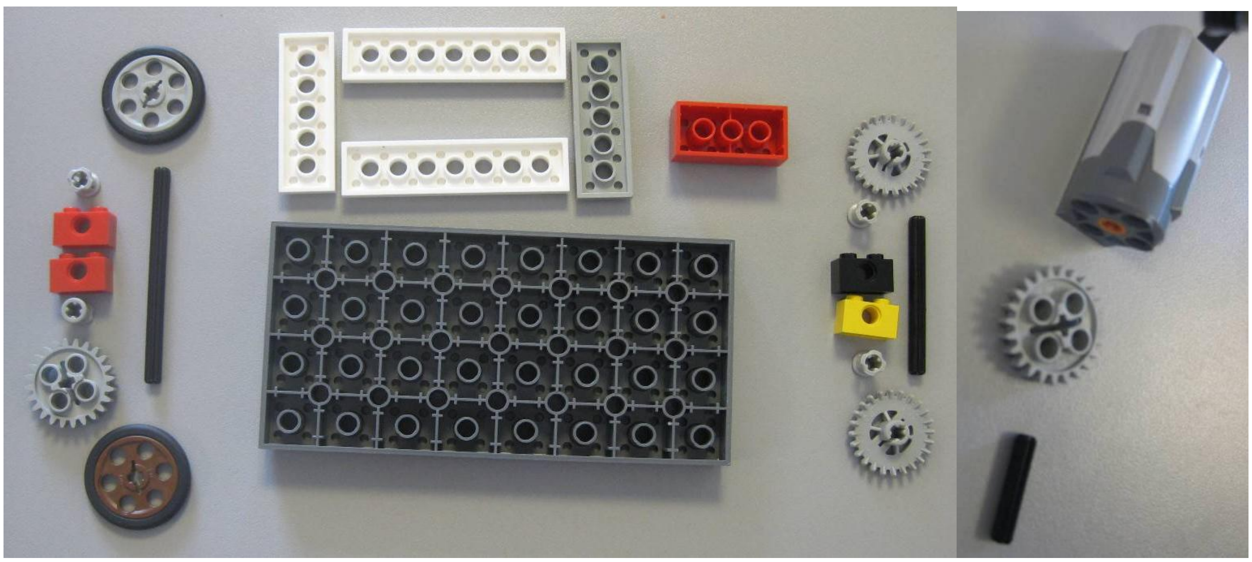
Step 5: Attach seesaw to axle.

WeDo™ Car

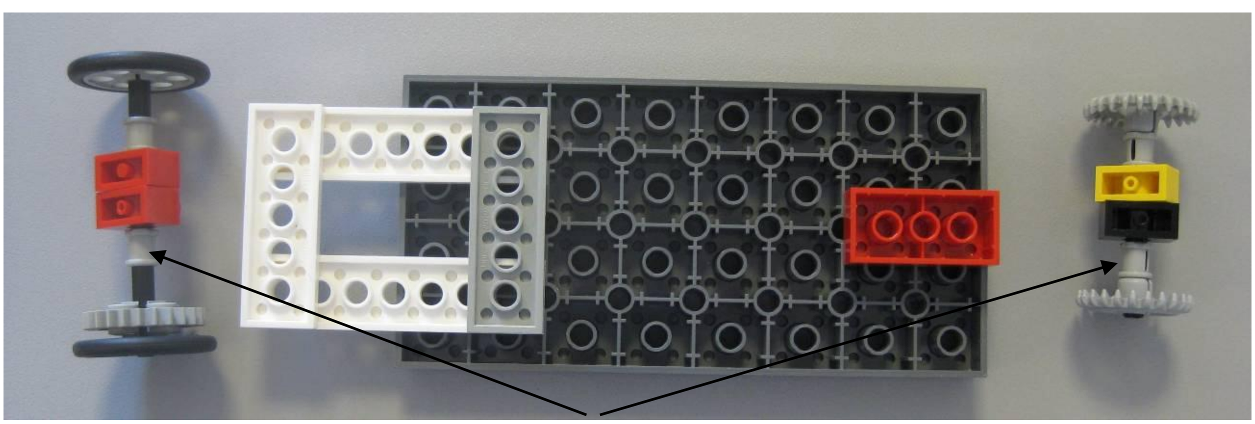
Tip: WeDo™ allows you to control up to 3 hubs with multiple motors or sensors of the same type, by labeling the motor or sensor block with a number. Press the shift key and click the motor / sensor block 1-7 times to toggle between #1-7 or no label (controls all simultaneously). You'll have to test which device is considered to be which number by WeDo™ each time you start the program.

1-motor car

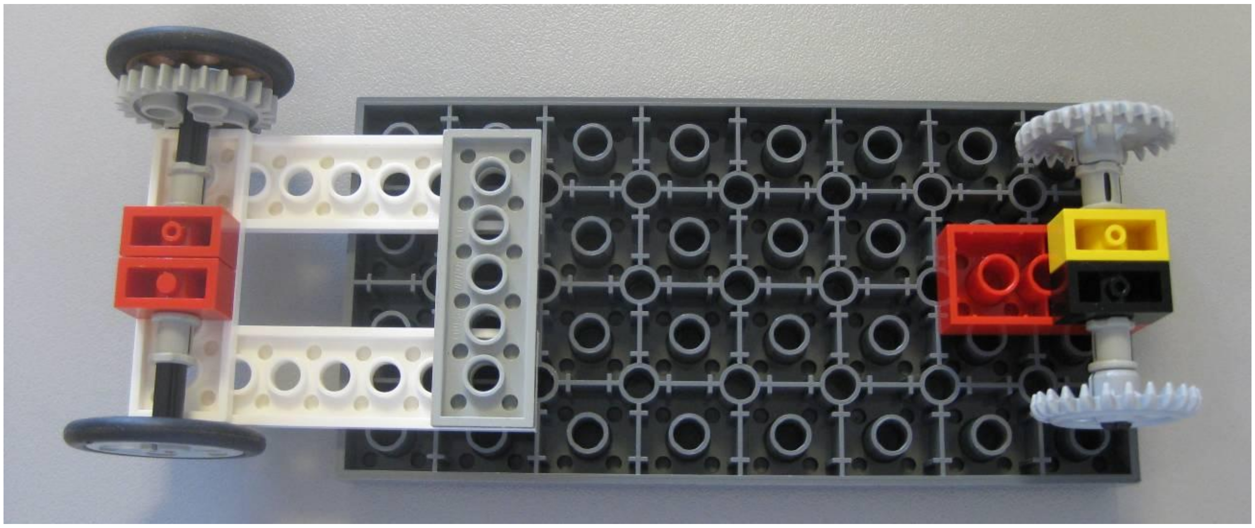
This 1-motor car is made entirely out of parts from the LEGO® WeDo™ kit. It can go forwards and backwards. Suggestions for variations including non-WeDo™ parts are provided. See instructions below for a car with 2 motors.



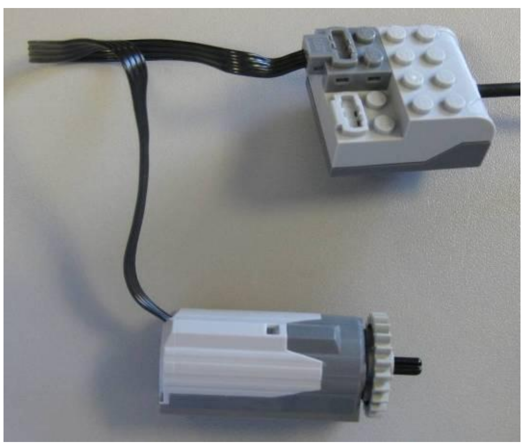
STEP 1A: **PARTS FOR THE BODY OF THE CAR, WHEELS, AXELS, AND MOTORS.**



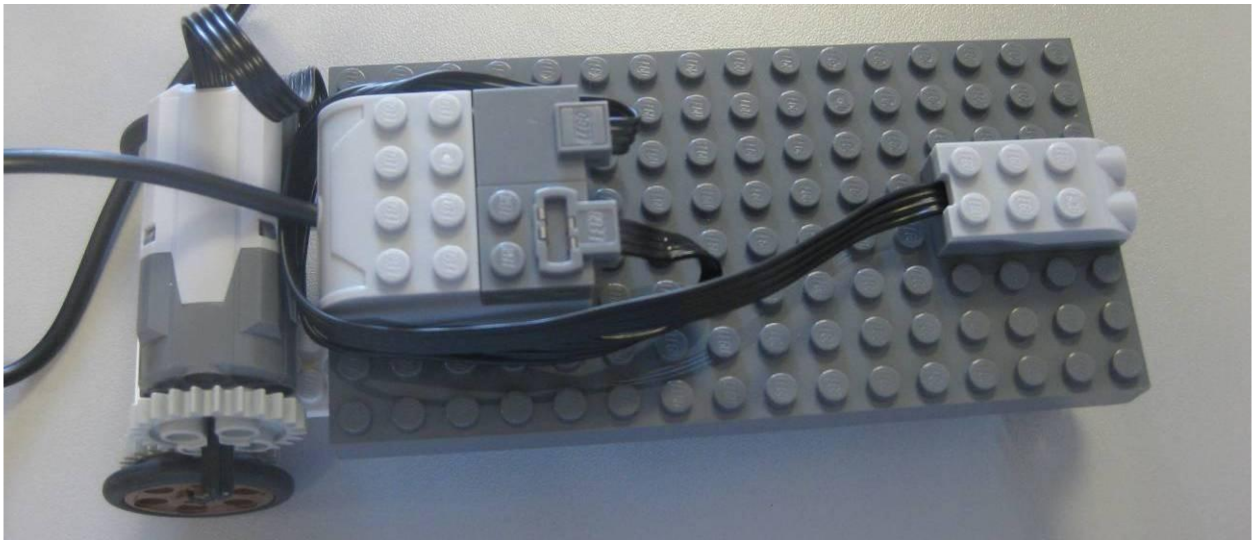
STEP 1B: **ATTACH THE PARTS AS SHOWN. THE BUSHINGS MUST BE LOOSE AGAINST THE 1X2 BRICKS SO THE AXEL SPINS FREELY.**



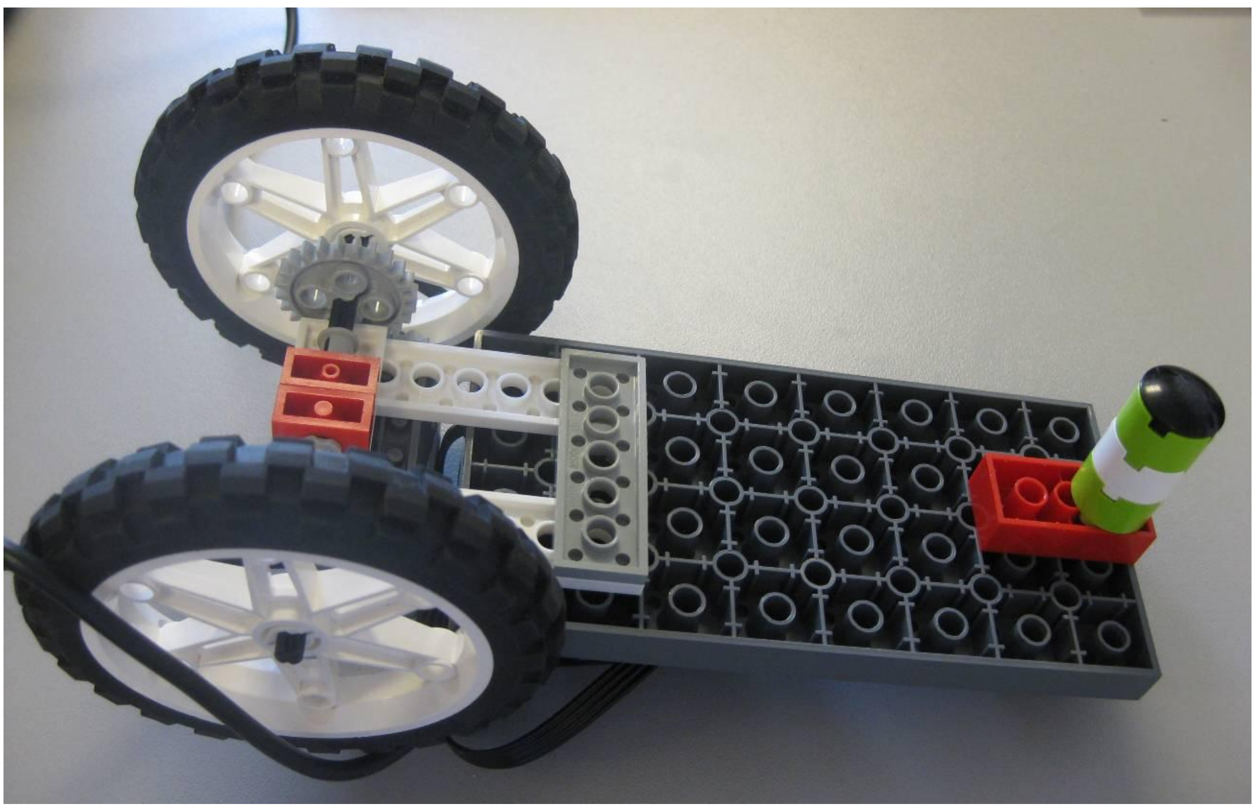
STEP 2: ATTACH ALL THREE GROUPS OF PARTS TOGETHER. MAKE SURE THE WHEELS AND AXEL STILL SPIN FREELY.



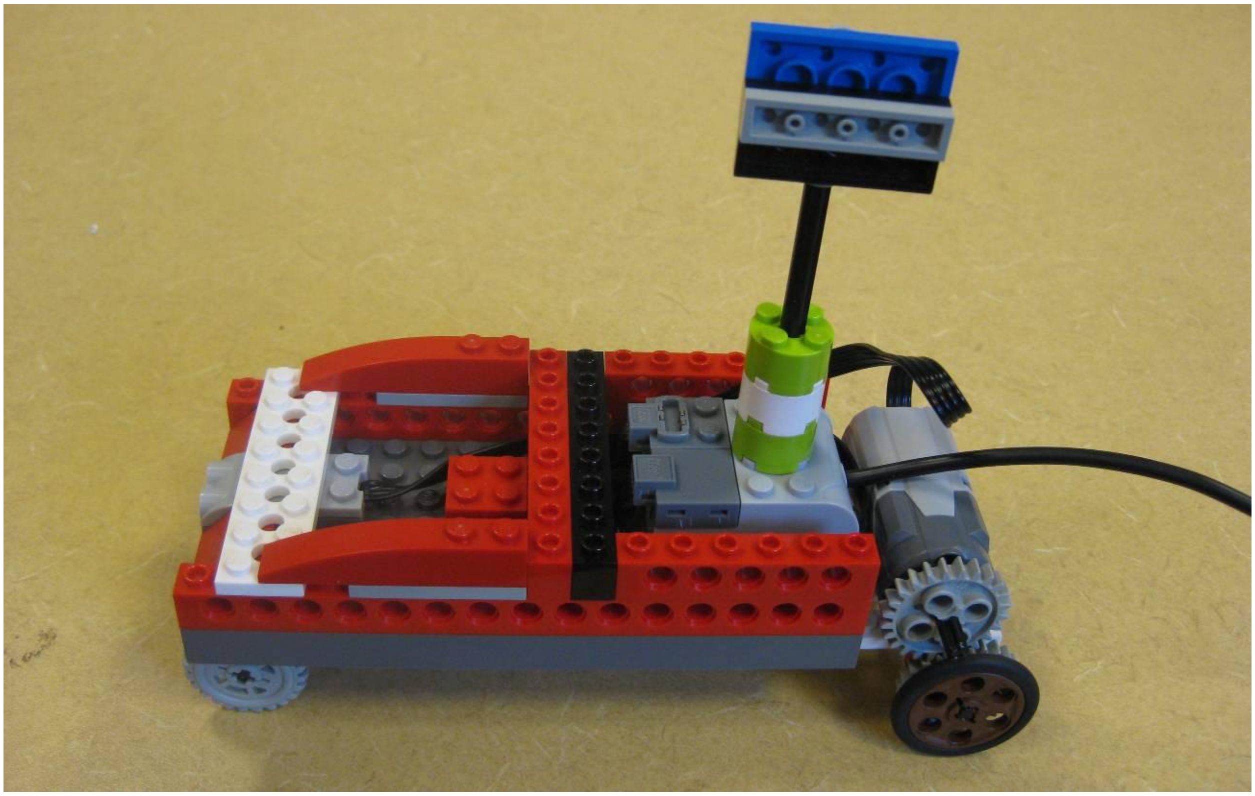
STEP 3: **CONNECT AN AXEL AND GEAR TO THE MOTOR, AND ATTACH THE MOTOR WIRE TO A USB HUB.**



STEP 4: **CENTER THE MOTOR ON THE WHITE PLATES ON THE LEFT SIDE OF THE GREY PLATE. BE SURE THE GEARS MESH. ATTACH THE HUB ANYWHERE ON THE GREY PLATE, WRAPPING WIRES OUT OF THE WAY AS NEEDED. YOU CAN ALSO ADD A SENSOR.**



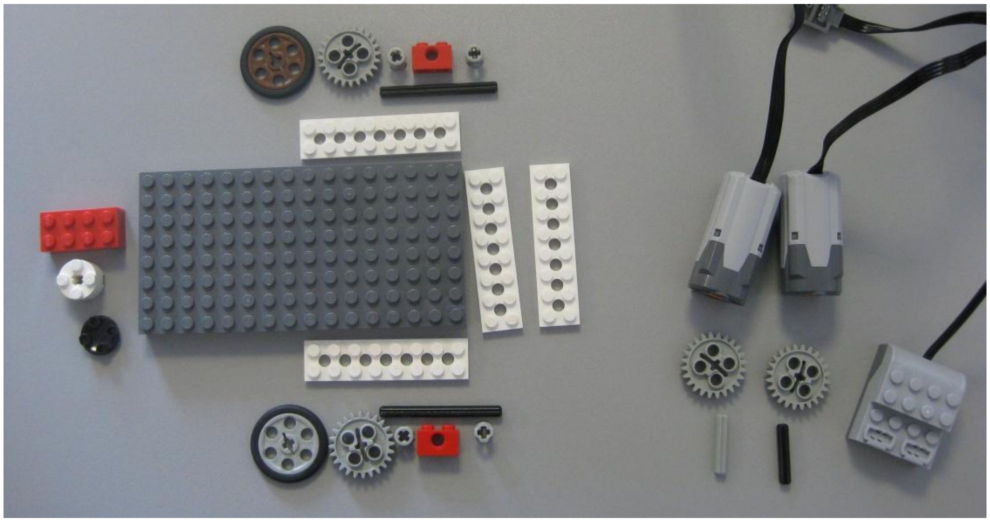
VARIATION: **A LONGER AXEL, LARGE WHEELS (BOTH LEGO® BUT NEITHER FROM WEDO™) AND/OR A SLIDER 'LEG' (FROM WEDO™) PROVIDE VARIATIONS ON THE BASIC CAR DESIGN SHOWN ABOVE.**



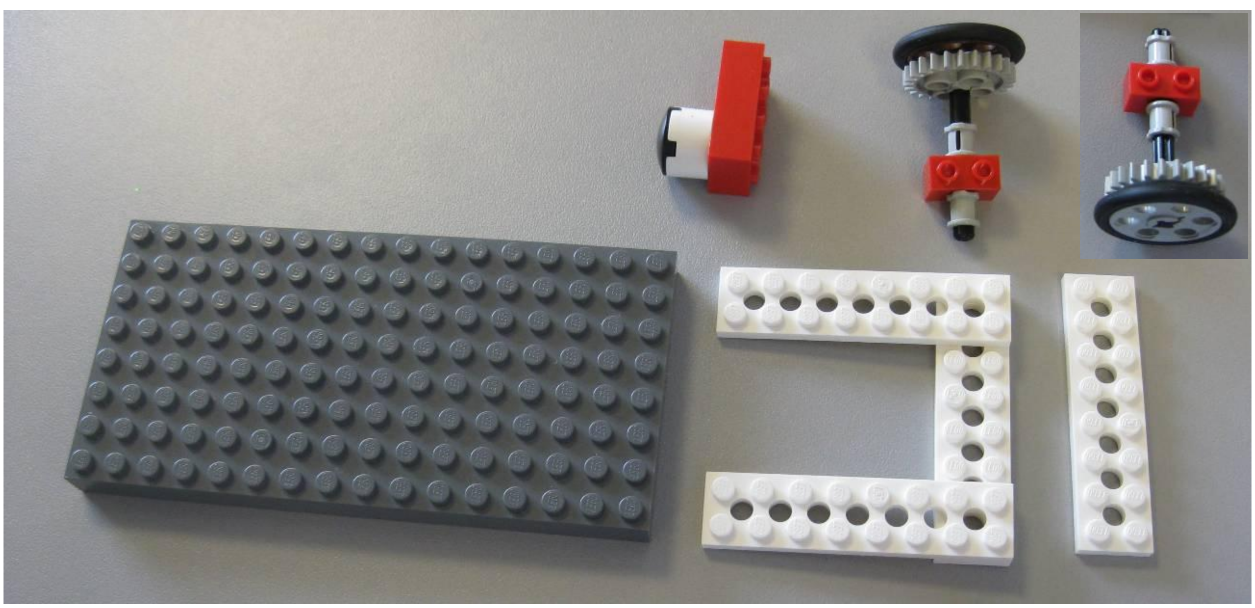
PERSONALIZE! **THIS IS A MARS EXPLORER. WHAT WILL YOUR ROBOT BE?**

2-motor car – see notes on page 1 for labeling motors to program them separately

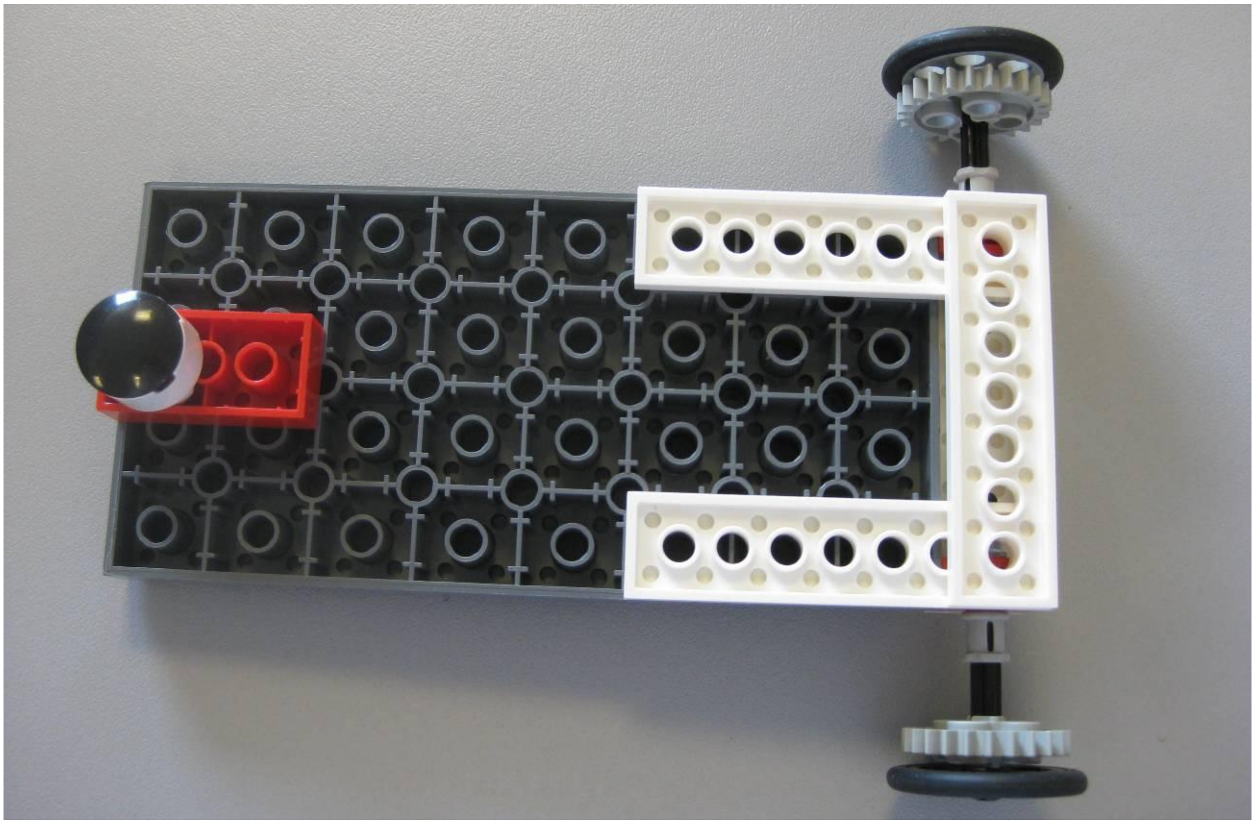
This 2-motor car is made out of parts from the LEGO® WeDo™ kit, with the addition of one extra motor and two extra gears. Use of sensors would require a second USB hub as well. ****If you have them, you could also put a set of big wheels right on the axels and avoid all the gears; see Finished photo.****



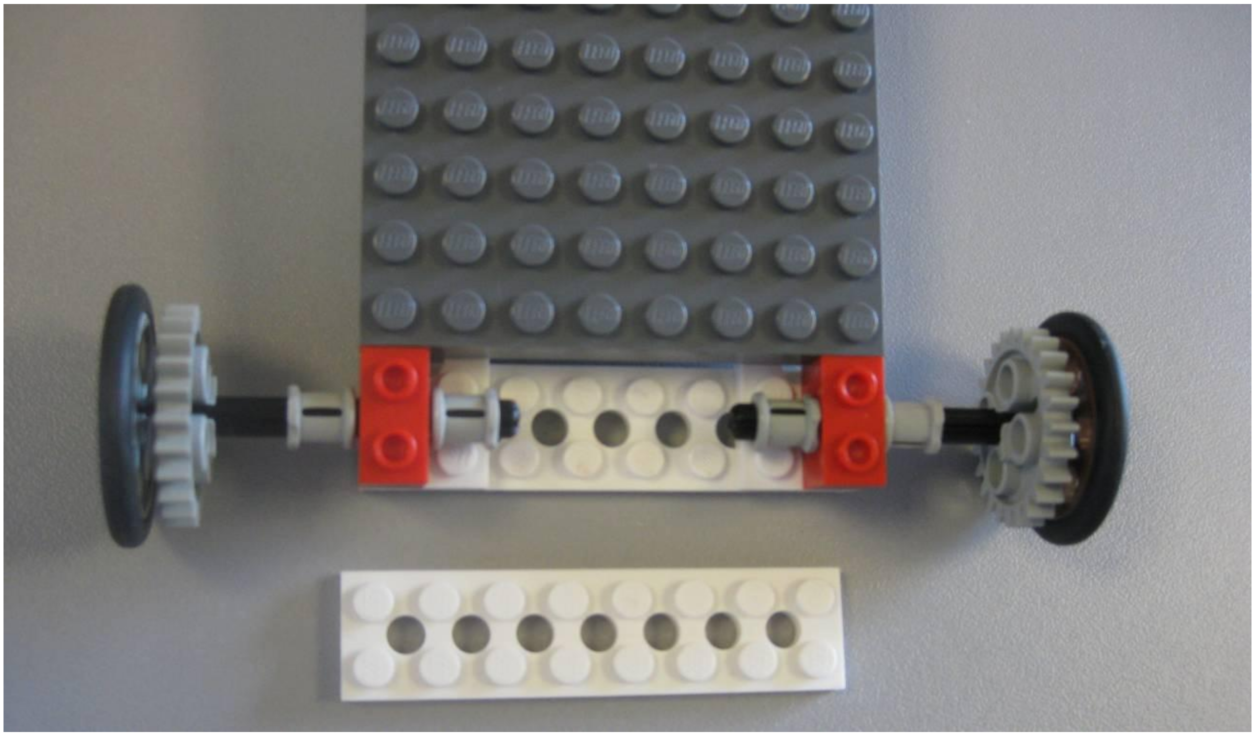
Step 1a: The parts here are similar to the parts in Step 1a for the 1-motor car. A notable exception is the use of a slider leg instead of wheels for the front of the car. Attach parts as shown.



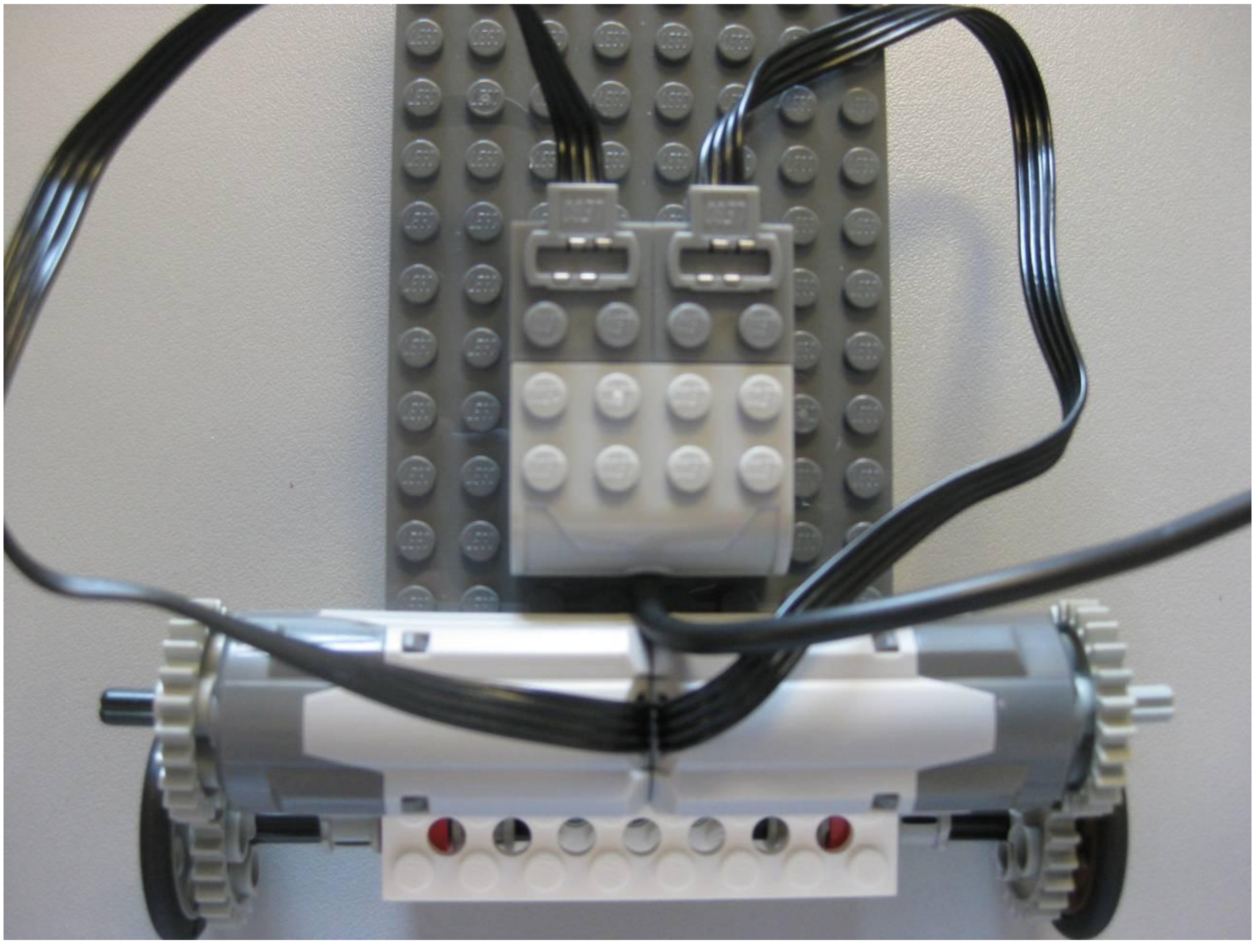
STEP 1B: **CONNECT THESE GROUPS AS SHOWN. ALSO ATTACH A SHORT AXEL AND GEAR TO EACH MOTOR (SHOWN GROUPED IN 1A).**



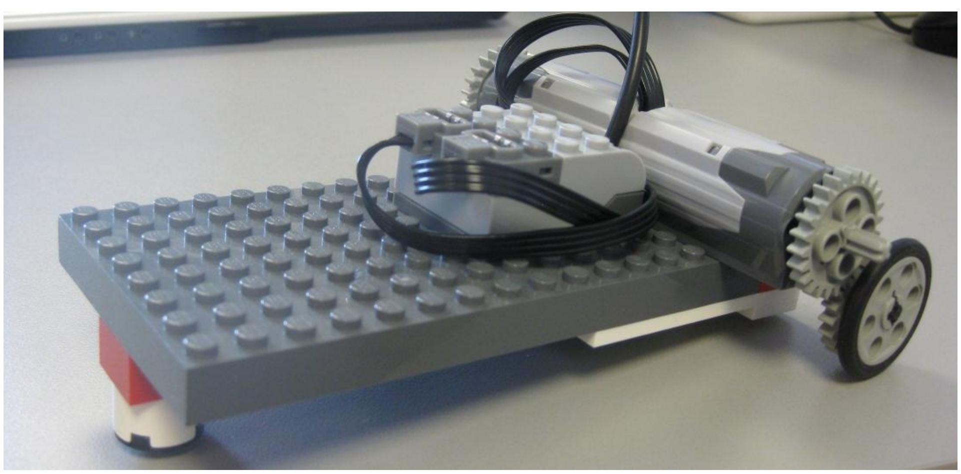
STEP 2A: **BOTTON VIEW OF ALL GROUPS OF PARTS CONNECTED.**



STEP 2B: **TOP VIEW OF ALL GROUPS OF PARTS CONNECTED. THE LOOSE 2X8 PLATE GOES ON TOP OF THE RED 1X2 BRICKS, SECURING THEM.**



STEP 3: ATTACH BOTH MOTORS ON THE LAST 2 ROWS OF PIPS ON THE LARGE GREY PLATE SO THE GEARS ON EACH SIDE LINE UP. ATTACH THE MOTOR WIRES TO A HUB AND ATTACH THE HUB ON THE GREY PLATE AS YOU WISH. SENSORS REQUIRE A SECOND HUB.



Finished! A complete, basic, 2-motor car.

List of WeDo™ Materials

Robotics materials

All robotic materials are found in the WeDo™ robotics construction sets available at:

http://www.LEGOeducation.us/eng/product/LEGO_education_WeDo_robotics_construction_set/2096

Programming materials:

- Computers installed with WeDo™ software.

Art Materials

- Various art materials including paper, scissors, markers, tape, and recyclable materials

Teaching materials:

- Posters showing robot parts
- Poster showing the Engineering Design Process
- Chart and images for "Is It a Robot?";
- Large icons for display / reference programs. They could be magnetic or felt-backed for magnetic whiteboards or felt-boards. Use whatever display surfaces are readily available;
- Design journals for final project and icons on paper for students to cut and tape / glue into their design journals;
- Engineer Licenses
- Expert Badges
- Assessment forms for each student.

Suggested "Think Like an Engineer" Pictures (Lesson 1)

Did An Engineer Make it?

Suggested “Jump for Engineers” Pictures.


Amtrak

]

Suggested "Robot or Not" Pictures (Lesson 2)

Appendix G:

Assessments

 Student's name:

Level 1: Engineering Design Process through Sturdy Building

When a child attempts this license level, also assess them according to the following scale. Use NA if not applicable.

If assessing a child based on a partially complete project, note this in the "Notes" section. In this case, assess the child based on their understanding of the core concepts below and how effectively they implemented them on the part of the project they did complete.

5	4	3	2	1	0
Complete Achievement of goal/task/ Understanding	Mostly Complete Achievement of goal/task/ understanding	Partially Complete Achievement of goal/task/ understanding	Very Incomplete Achievement of goal/task/ understanding	Did Not Complete goal/task/ understanding	Did not attempt/Other

Skill	*Achievement Level*
1. Structure is built to add and remove a toy person	5 4 3 2 1 0 NA
2. Structure remains intact while being handled and functions as it is designed to function.	5 4 3 2 1 0 NA

Notes:

Overall Debugging:

1. A. Recognizes that something is not working.	5 4 3 2 1 0 NA
2. B. Keeps original goal or changes to an acceptable alternative.	5 4 3 2 1 0 NA
3. C. Has a hypothesis of the cause of the problem.	5 4 3 2 1 0 NA
4. D. Attempts to solve the problem.	5 4 3 2 1 0 NA

Notes:

Level 2: Robot Builder

When a child attempts this license level, also assess them according to the following scale. Use NA if not applicable.

If assessing a child based on a partially complete project, note this in the "Notes" section. In this case, assess the child based on their understanding of the core concepts below and how effectively they implemented them on the part of the project they did complete.

5	4	3	2	1	0
Complete Achievement of goal/task/ Understanding	Mostly Complete Achievement of goal/task/ understanding	Partially Complete Achievement of goal/task/ understanding	Very Incomplete Achievement of goal/task/ understanding	Did Not Complete goal/task/ understanding	Did not attempt/Other

1. Knows their robot needs specific parts for specific actions and uses those parts.	5 4 3 2 1 0 NA
2. Attaches all necessary robot parts so that they work correctly (i.e. motor wire properly connected to hub, axle connected to motor, hub connected to computer).	5 4 3 2 1 0 NA
3. Knows how to program the robot to move motor	5 4 3 2 1 0 NA

Notes:

Overall Debugging:

1. A. Recognizes that something is not working.	5 4 3 2 1 0 NA
2. B. Keeps original goal or changes to an acceptable alternative.	5 4 3 2 1 0 NA
3. C. Has a hypothesis of the cause of the problem.	5 4 3 2 1 0 NA
4. D. Attempts to solve the problem.	5 4 3 2 1 0 NA

Notes:

Level 3: Programmer I

When a child attempts this license level, also assess them according to the following scale. Use NA if not applicable.

If assessing a child based on a partially complete project, note this in the "Notes" section. In this case, assess the child based on their understanding of the core concepts below and how effectively they implemented them on the part of the project they did complete.

5	4	3	2	1	0
Complete Achievement of goal/task/ understanding	Mostly Complete Achievement of goal/task/ understanding	Partially Complete Achievement of goal/task/ understanding	Very Incomplete Achievement of goal/task/ understanding	Did Not Complete goal/task/ understanding	Did not attempt/Other

1. Selects the right instructions.	5 4 3 2 1 0 NA
2. Arranges instructions in the correct order.	5 4 3 2 1 0 NA

Notes:

Overall Debugging:

1. A. Recognizes incorrect instructions or order by reading the program or watching the robot run the program.	5 4 3 2 1 0 NA
2. B. Keeps original goal.	5 4 3 2 1 0 NA
3. C. Has a hypothesis of the cause of the problem.	5 4 3 2 1 0 NA
4. D. Attempts to solve the problem.	5 4 3 2 1 0 NA

Notes:

Level 4: Programmer II

When a child attempts this license level, also assess them according to the following scale. Use NA if not applicable.

If assessing a child based on a partially complete project, note this in the "Notes" section. In this case, assess the child based on their understanding of the core concepts below and how effectively they implemented them on the part of the project they did complete.

5	4	3	2	1	0
Complete Achievement of goal/task/ understanding	Mostly Complete Achievement of goal/task/ understanding	Partially Complete Achievement of goal/task/ understanding	Very Incomplete Achievement of goal/task/ understanding	Did Not Complete goal/task/ understanding	Did not attempt/Other

1. Knows when and how to use Repeats.	5 4 3 2 1 0 NA
2. Knows when and how to use number parameters.	5 4 3 2 1 0 NA
3. Selects the right instructions.	5 4 3 2 1 0 NA
4. Arranges the instructions in the correct order.	5 4 3 2 1 0 NA

Notes:

Overall Debugging:

1. A. Recognizes incorrect instructions or order by reading the program or watching the robot run the program.	5 4 3 2 1 0 NA
2. B. Keeps original goal.	5 4 3 2 1 0 NA
3. C. Has a hypothesis of the cause of the problem.	5 4 3 2 1 0 NA
4. D. Attempts to solve the problem.	5 4 3 2 1 0 NA

Notes:

Level 5: Programmer III

When a child attempts this license level, also assess them according to the following scale. Use NA if not applicable.

If assessing a child based on a partially complete project, note this in the "Notes" section. In this case, assess the child based on their understanding of the core concepts below and how effectively they implemented them on the part of the project they did complete.

5	4	3	2	1	0
Complete Achievement of goal/task/ understanding	Mostly Complete Achievement of goal/task/ understanding	Partially Complete Achievement of goal/task/ understanding	Very Incomplete Achievement of goal/task/ understanding	Did Not Complete goal/task/ understanding	Did not attempt/Other

1. Knows how to use sensors and what they are for.	5 4 3 2 1 0 NA
2. Knows when and how to use sensor parameters.	5 4 3 2 1 0 NA
3. Selects the right instructions.	5 4 3 2 1 0 NA
4. Arranges instructions in the correct order.	5 4 3 2 1 0 NA

Notes:

Overall Debugging:

1. A. Recognizes incorrect instructions or order by reading the program or watching the robot run the program.	5 4 3 2 1 0 NA
2. B. Keeps original goal.	5 4 3 2 1 0 NA
3. C. Has a hypothesis of the cause of the problem.	5 4 3 2 1 0 NA
4. D. Attempts to solve the problem.	5 4 3 2 1 0 NA

Notes:

Level 6: Gear Expert

When a child attempts this license level, also assess them according to the following scale. Use NA if not applicable.

If assessing a child based on a partially complete project, note this in the "Notes" section. In this case, assess the child based on their understanding of the core concepts below and how effectively they implemented them on the part of the project they did complete.

5	4	3	2	1	0
Complete Achievement of goal/task/ understanding	Mostly Complete Achievement of goal/task/ understanding	Partially Complete Achievement of goal/task/ understanding	Very Incomplete Achievement of goal/task/ understanding	Did Not Complete goal/task/ understanding	Did not attempt/Other

1. Knows how to use gears and what they are for.	5 4 3 2 1 0 NA
2. Understands concepts of gearing up and gearing down.	5 4 3 2 1 0 NA

Notes:

Overall Debugging:

1. A. Recognizes that something is not working	5 4 3 2 1 0 NA
2. B. Keeps original goal or changes to an acceptable alternative	5 4 3 2 1 0 NA
3. C. Has a hypothesis of the cause of the problem.	5 4 3 2 1 0 NA
4. D. Attempts to solve the problem.	5 4 3 2 1 0 NA

Notes:

Level 6: Expert

Part 1: Assess each child along these scales when they have finished their final project.

5	4	3	2	1	0
Complete Achievement of goal/task/ understanding	Mostly Complete Achievement of goal/task/ understanding	Partially Complete Achievement of goal/task/ understanding	Very Incomplete Achievement of goal/task/ understanding	Did Not Complete goal/task/ understanding	Did not attempt/Other

1. Has a goal for the project.	5 4 3 2 1 0 NA
2. Selects the right instructions to accomplish the goal.	5 4 3 2 1 0 NA
3. Arranges instructions in the correct order to accomplish the goal.	5 4 3 2 1 0 NA
4. Knows how and when to use Repeats.	5 4 3 2 1 0 NA
5. Knows how and when to use number parameters.	5 4 3 2 1 0 NA
6. Knows how to use sensors and what they are for.	5 4 3 2 1 0 NA
7. Knows how and when to use sensor parameters with Repeats.	5 4 3 2 1 0 NA

Notes:

Overall Debugging:

1. A. Recognizes incorrect instructions or order by reading the program or watching the robot run the program.	5 4 3 2 1 0 NA
2. B. Keeps original goal.	5 4 3 2 1 0 NA
3. C. Has a hypothesis of the cause of the problem.	5 4 3 2 1 0 NA
4. D. Attempts to solve the problem.	5 4 3 2 1 0 NA

Notes:

Level 6: Expert, cont.

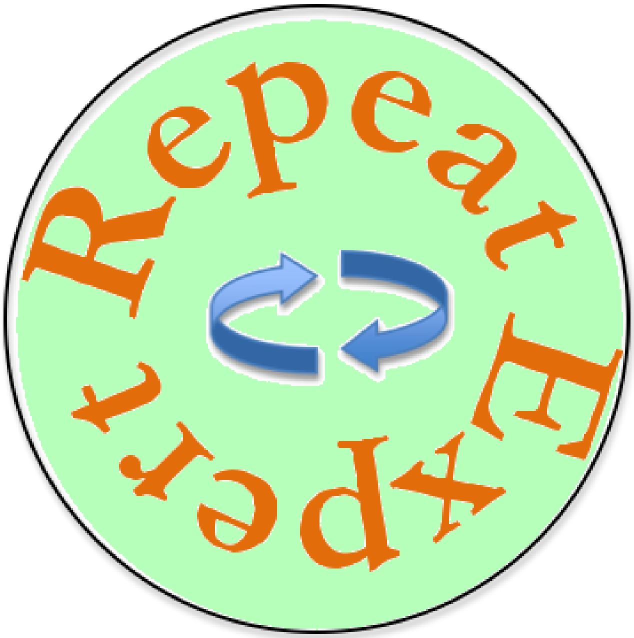
Part 2: Ask each child these questions to supplement the students' journals and the teachers' observations of and conversations with students during work and sharing times.

- What does your program tell your robot to do? How did you choose those instructions?
- What parts does your robot have (robotic and/or aesthetic)? Why did you choose them?

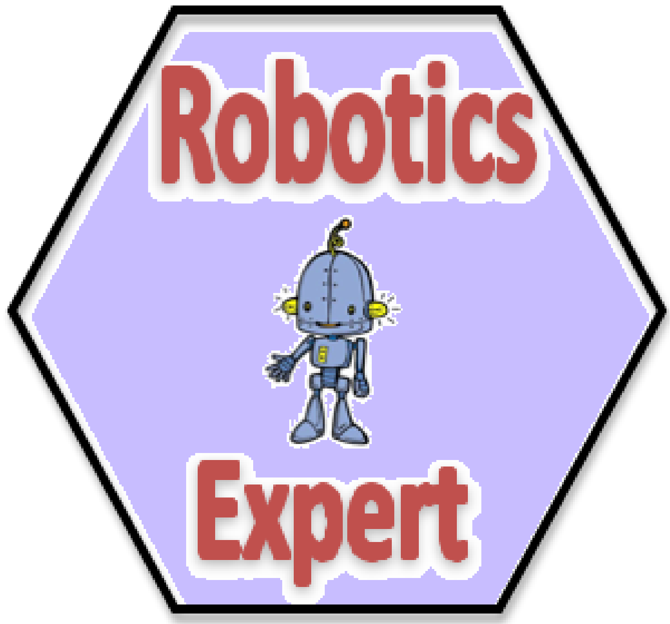
Mark the students' level of understanding of how to program a robot along the following criteria.

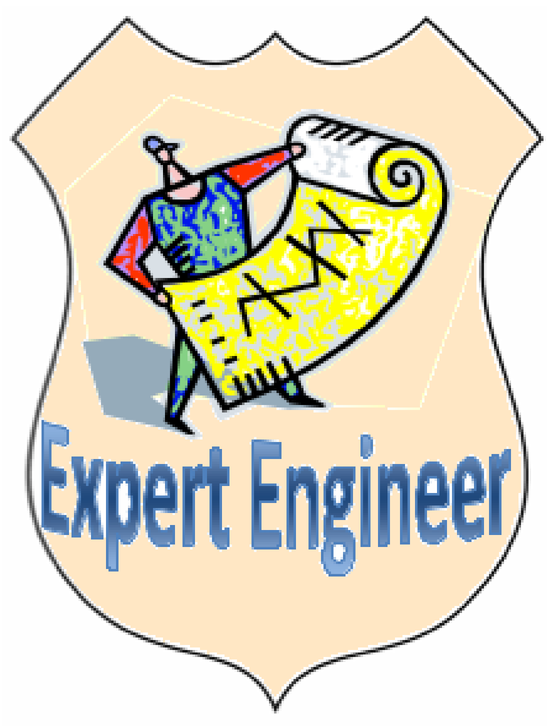
	Units:	Understands the function of individual robot parts and individual programming instructions, but not how to choose and assemble them to make a functional robot or program that accomplishes a given goal.
	Connections:	Chooses appropriate parts for the robot and instructions for the program. Puts parts together correctly and instructions in the right order. Understands that putting the parts together in certain ways creates an overall outcome. Does not see the connection between the whole program and then accomplishment of the chosen goal.
	Context:	Understands the function of each element and that the order they are put in results in a specific overall outcome. Is able to purposefully put the right instructions in the right order for the program to achieve the given goal.

Sample Expert Badges

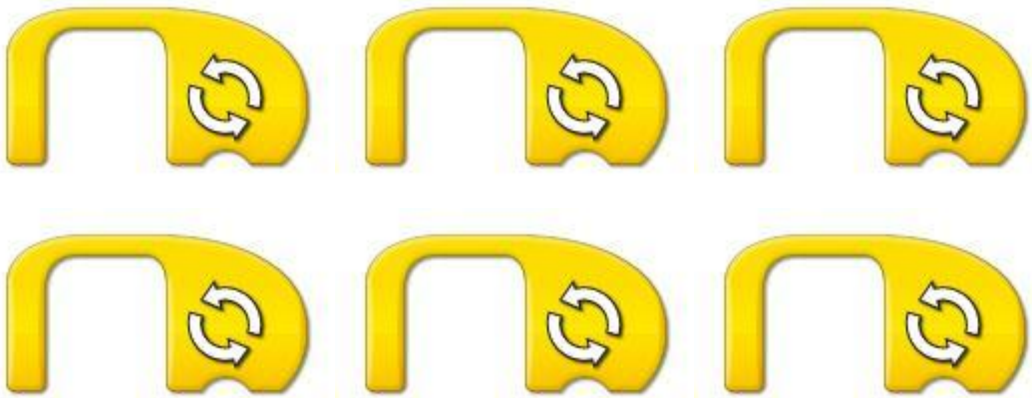



Sensors
Expert


Robotics
Expert


Expert Engineer

Sample Icons for Use in Engineering Design Journals



Appendix H

Positive Technological Development

Figure 10: *Personal Development Trajectory*

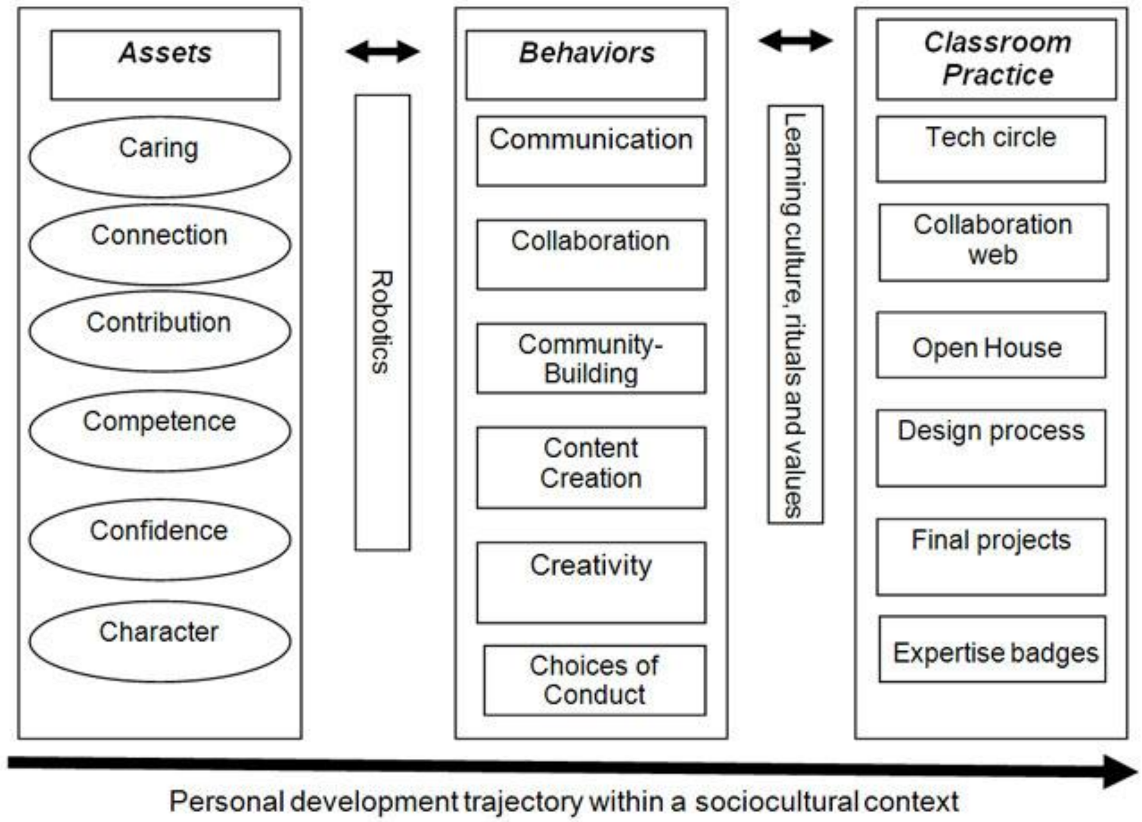


Figure 11: Sample Collaboration Web

References

Bers, M. (2008). Blocks to robots: Learning with technology in the early childhood classroom. New York, NY: Teachers College.

Common Core Standards (2011). Retrieved from http://www.corestandards.org/

Crouser, R.J. (2010). *C.h.e.r.p.* Retrieved from http://ase.tufts.edu/DevTech/tangiblek/research/cherp.asp

Crouser, R.J. (2010). *Cherp documentation* Retrieved from http://ase.tufts.edu/DevTech/tangiblek/research/Cherp%20Documentation.pdf

Darragh, J.C. (2006). *The environment as the third teacher.* Retrieved from http:// http://www.eric.ed.gov/PDFS/ED493517.pdf

International Technology and Engineering Educators Association, (2007). *Standards for technological literacy: Content for the study of technology.* Reston, VA: International Technology Educators Association.

Investigations Curriculum (2008) developed by TERC, Cambridge, MA.

New York State Learning Standards and Core Curriculum (2011). Retrieved from http://www.p12.nysed.gov/ciai/standards.html

Papert, S. (1993). Mindstorms: Children, computers, and powerful ideas. New York, NY: Basic Books.