Teaching ROBOTC®
for LEGO® MINDSTORMS®
Product Overview
Why Robotics/STEM Education

We can’t predict what the hot new technology will be in five years, but we can confidently predict that it will include computer programming, electronic embedded systems, engineering design, and mathematics. If you believe these things, then you need to know that robotics has the ability to teach these concepts. At the same time, robotics teaches 21st century skill sets like time management, resource allocation, teamwork, problem solving, and communications.

Think about this...

- Approximately 98% of all the 32-bit microprocessors currently in use worldwide are used in embedded systems; in other words they are being used in robotic smart technologies.
- By the year 2010, it is forecasted that 90% of the overall program code developed will be for embedded computing systems; to innovate and compete globally we will need more people that know how to program.
- Robotics Technology is a hundred billion dollar emerging industry that has moved from being an industry that could potentially employ thousands of people to an integral part of all industries. Robotics will impact the economy the same way that mass production impacted the industrial revolution and the computer impacted the information age.
- Science and Engineering (S&E) occupations are projected to grow by 26% from 2010 to 2020, twice as fast as the overall job market during that period (S&EI 2008) yet we have fewer students pursuing S&E careers.

The Teaching ROBOTC for LEGO MINDSTORMS Curriculum

If you are reading this, then you are probably considering teaching using the NXT and ROBOTC. I understand the dilemma that you are facing, I taught in the Pittsburgh Public School System for 27 years before coming to Carnegie Mellon. You have a limited budget, you have a thousand things to do every day that don’t involve teaching, and then you also teach five or six sections of students. (Sometimes with multiple preps!) This curriculum is designed for a teacher with no programming background that is interested in teaching programming and engineering. This curriculum also supports a teacher that knows how to program, but has students of various skill levels and wants to allow them to move at their own pace. The teaching materials that we’ve developed use a high level of multimedia and have been tested with hundreds of students; they work. ROBOTC is the best programming software available for use with the NXT if you consider the percentage of teams that use ROBOTC software and made it to the finals of the FTC competition. You have one of the hardest jobs in the world; you teach. This training tool will make it easier.

Math is the language of science, engineering and technology

Many teachers see robotics as a way to teach STEM education. We’ve seen that robotics does provide unique opportunities for teachers to place engineering design, scientific process, technological literacy and mathematics in contexts that students find engaging and understand. Across the nation, many schools and community-based organizations are using robotics to address STEM competencies. Yet, our research is finding that many teachers miss key STEM “teaching moments” that robotics enables. Often, robotics teachers will allow students to be haphazard in their design process and avoid mathematics when possible (e.g., using guess-and-check strategies). This methodology leads to weak solutions and reduces student learning.

At the Robotics Academy we believe that:

- Math is the language of STEM and if you can’t do math, then you won’t be able to compete for a STEM job.
- Mathematics needs to be carefully thought out by the teacher and foregrounded for the student. The focus of the math instruction must be centered on addressing specific mathematics concepts (not general) and the mathematics in the lesson must be made explicit not implicit.
- For students’ STEM understanding to move beyond parroting the teacher’s words, ideas, and solutions, and to develop deep understanding, students need the opportunity to struggle with the problem, be able to defend their decisions, and explain their answer in their own words.

The moral of the story is “require your students to do the math.” To learn more about teaching with robotics visit our web site. Carnegie Mellon is committed to helping teachers teach robotics. If there is something that we can help you with, then please contact us. If you see opportunities to make our teaching products stronger, then please contact us.

Have a great year.

Robin Shoop,
Director, Carnegie Mellon Robotics Academy
### Teaching ROBOTC for LEGO MINDSTORMS

Teaching is a craft and every teacher does it differently. This curriculum is designed to teach “engineering process” and “programming”. The Robotics Academy has developed this curriculum to help teachers to teach and students to learn those competencies. The Robotics Academy is committed to helping teachers use robotics to teach science, technology, engineering, and mathematics. Teachers can find additional NXT resources to teach robotics and engineering at our NXT site, which is continually upgraded by the Robotics Academy, go to www.nxteacher.com.

The scope and sequence below is designed to help you to quickly find resources to teach your class. The number of days at the right will depend on the number of challenges and labs that your students complete.

<table>
<thead>
<tr>
<th>Welcome to Robotics Class</th>
<th>1-2 days</th>
</tr>
</thead>
<tbody>
<tr>
<td>What are the goals of this class?</td>
<td></td>
</tr>
<tr>
<td>What is the definition of a robot? - SPA handout - page 37</td>
<td></td>
</tr>
<tr>
<td>What is the definition of engineering? - Videos and handouts - page 10 &amp; 11</td>
<td></td>
</tr>
<tr>
<td>What is the definition of programming? - Videos and handouts - page 15</td>
<td></td>
</tr>
<tr>
<td>What is the definition of a system?</td>
<td></td>
</tr>
<tr>
<td>What does it mean to manage a project? - Videos and handouts - page 10</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class organization rules</th>
<th>1-2 days</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grading/Rubrics for Evaluation – page 13</td>
<td></td>
</tr>
<tr>
<td>Lab Procedures – pages 13 &amp; 14</td>
<td></td>
</tr>
<tr>
<td>Keeping an Engineering Journal – page 10</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Safety</th>
<th>2-3 days and then ongoing</th>
</tr>
</thead>
<tbody>
<tr>
<td>General Safety handout – page 8</td>
<td></td>
</tr>
<tr>
<td>Safety Checklist handout – page 9</td>
<td></td>
</tr>
<tr>
<td>Safety Quiz – page 9</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Introduction to the NXT Hardware</th>
<th>2-3 days then ongoing</th>
</tr>
</thead>
<tbody>
<tr>
<td>The NXT Controller – page 18</td>
<td></td>
</tr>
<tr>
<td>NXT Sensors – page 17</td>
<td></td>
</tr>
<tr>
<td>NXT Parts Identification – page 17</td>
<td></td>
</tr>
<tr>
<td>Building your first robot – page 21</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Introduction to ROBOTC Software</th>
<th>1 days then ongoing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Download Firmware video – page 21</td>
<td></td>
</tr>
<tr>
<td>Download Your First Program lesson video – page 21</td>
<td></td>
</tr>
<tr>
<td>Introduction to ROBOTC 2.0 Software lesson video – page 15</td>
<td></td>
</tr>
<tr>
<td>Programming Quizzes and handouts – page 22</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Introduction to Programming</th>
<th>1-2 days</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thinking About Programming lesson video – page 15</td>
<td></td>
</tr>
<tr>
<td>ROBOTC Programming Syntax lesson video – page 16</td>
<td></td>
</tr>
<tr>
<td>Behaviors/pseudocode handout – page 15</td>
<td></td>
</tr>
<tr>
<td>Whitespace/Comments/ reserved words handouts – page 16</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Movement – NXT Forward/Backward/Turning</th>
<th>8-10 days</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labyrinth Challenge – page 27</td>
<td></td>
</tr>
<tr>
<td>Moving Forward lesson videos and handouts – pages 28 &amp; 29</td>
<td></td>
</tr>
<tr>
<td>Motor Power Engineering Lab – page 30</td>
<td></td>
</tr>
<tr>
<td>Speed and Direction lesson videos and handouts – pages 29 &amp; 30</td>
<td></td>
</tr>
<tr>
<td>Turning Engineering Lab – page 30</td>
<td></td>
</tr>
<tr>
<td>PID videos and handout – page 31</td>
<td></td>
</tr>
<tr>
<td>Synchronized Motors lesson video and handout – page 32</td>
<td></td>
</tr>
<tr>
<td>Synchronized Motors Engineering Lab – page 32</td>
<td></td>
</tr>
<tr>
<td>Introduction to Encoders lesson video and handouts – page 32</td>
<td></td>
</tr>
</tbody>
</table>
Teaching ROBOTC for LEGO MINDSTORMS

Sensing .............................................................................................................................. 20-25 days
  The Obstacle Course Programming Challenge – page 34

Touch Sensor – pages 35 through 37 ........................................................................... 3-5 days
  While Loop lesson video – page 36
  While Loop reference handout – page 37
  Sense-Plan-Act Algorithm reference handout – page 37
  Boolean Logic lesson videos and handouts – pages 36 & 37
  Touch Sensor Programming Challenges – page 37
  Touch Sensor Quiz – page 37

Ultrasonic Sensor – pages 38 & 39 .............................................................................. 2-3 days
  Ultrasonic lesson video – page 39.
  Calculating Thresholds handout – page 39
  Random Numbers reference handout – page 39
  Ultrasonic Programming Challenges – page 39
  Ultrasonic Sensor Quiz – page 39

Encoders/the LEGO Smart Motors – pages 40 & 41 .................................................. 3-5 days
  Encoder lesson videos – page 41
  Encoder Engineering Lab – page 41
  Motor Encoder reference handout – page 41
  Encoder programming Challenge – page 41

Light Sensor – pages 42 & 43 ...................................................................................... 3-5 days
  Light Sensor lesson videos – page 43
  Light Sensor Challenges – page 43
  Light Sensor Quiz – page 43

Light Sensor/Line Tracking – pages 44 through 46 ..................................................... 3-5 days
  Line Tracking lesson videos – page 45
  Timer video – page 45
  If-else Statement reference handout – page 46
  Switch Case reference handout – page 46
  Line Tracking Programming Challenges – page 46
  Line Tracking Quiz – page 46

Sound Sensor – pages 47 .............................................................................................. 2-3 days
  Sound Sensor lesson videos – page 47
  Sound Sensor reference handout – page 47
  Sound Sensor Quiz – page 47

Variables and Functions – page 48 ............................................................................ 10-15 days
  The Warehouse Programming Challenge – page 49

Automatic Threshold Calculations – pages 50 through 52 ...................................... 3-5 days
  Automatic Threshold lesson videos – page 51
  Values and Variables lesson videos – page 51
  Variables and the Debugger lesson video – page 51
  Variables reference handout – page 52
  Text to Display lab – page 52
  Automatic Calculations Programming Challenge – page 52
  Automatic Thresholds Quiz – page 52

Variables and Functions/Counting – pages 53 through 55 ...................................... 3-5 days
  Line Counting lesson videos – page 54
  Quick Tap Programming Challenges – page 55
  Line Counting Quiz – page 55
Teaching ROBOTC for LEGO MINDSTORMS

Variables and Functions/Patterns of Behaviors - pages 56 through 58 .......................................................... 3-5 days
- Variables and Functions lesson videos – page 57
- Global Variables reference handout – page 58
- Functions reference handout – page 58
- Functions Programming Challenges – page 58
- Functions Programming Quiz – page 58

Debugging – page 59 ............................................................................................................................................ 2 days
- Debugging lesson videos – page 59

Remote Control – page 60 .................................................................................................................................... 4-10 days
- Remote Control Soccer Programming Challenge – page 61

Remote Control Basics – pages 62 & 63
- Remote Control lesson videos – page 63
- Remote Control reference handout – page 63
- Remote Control Buttons lab – page 63
- Remote Control Programming Challenge – page 63

Using Bluetooth - pages 64 & 65
- Using Bluetooth lesson videos – page 65
- USB Bluetooth Adaptor reference handout page 65

How Robotics Addresses Standards - pages 76 through 85
- National Science Education Standards - pages 76 through 79
- National Council of Teachers of Mathematics - pages 80 & 81
- International Technology Education Association - pages 82 & 83
- Reading, Writing, Listening, and Presenting - pages 84 & 85

Additional Robotics Academy LEGO Robotics Resources - pages 86 & 87
Getting Started

The Teaching ROBOTC for LEGO MINDSTORMS is a comprehensive guide that teaches how to program the NXT Mindstorms hardware system as it helps student develop engineering competencies.

Scaffolded Approach to Learning

This curriculum uses a scaffolded approach to teach students how to program the NXT system. Students begin by learning to use, troubleshoot, and program the LEGO MINDSTORMS NXT robot system while advancing to more complex programming concepts. Once students understand how to program and troubleshoot the NXT system, there are Programming Challenges, Mini-challenges, and Engineering Challenges to practice what you’ve learned.
Getting Started/Fundamentals

The curriculum is divided into two sections: “Getting Started” and “Programming and Engineering.”

The Getting Started Lessons are found in the “Fundamentals” and “Setup” sections.

The Programming and Engineering Lessons are found in the Movement, Sensing, Variables, and Remote Control sections.

The Fundamentals Unit

The Fundamentals Unit is divided into six Lesson Sets: Safety, Project Management, Assessment Rubrics, Introduction to Programming, and NXT hardware.
Safety

Any course that involves moving parts, handling and processing materials and students requires safety training. Safety begins with the development of a safe attitude. Most accidents can be avoided if a student develops a safe and conscientious attitude. The safety lesson begins by challenging a student’s general beliefs about safety and concludes with a safety inspection of the robotics lab.

Safety is an Attitude - A one page handout that defines what safety is and what safety is not, and concludes with statements that support the fact that most accidents are preventable with the development of a safe attitude.

General Lab Safety - A four page handout that spells out general safety rules, describes features of a safe classroom, safe storage, material handling, disposal of materials, tools and equipment, and ends with a list of definitions of terms that students may not know.

Safety Checklist - A three page handout that contains a safety checklist, rules to consider when you are moving things around the lab, and a one page safety poster.

Electrical Safety - A two page handout that describes safety rules when working with electricity and common causes of electrical accidents, including defective equipment, unsafe practices, and lack of electrical knowledge.

Power Tool Safety - A one page handout that sets rules and expectations for when students use power tools in the robotics lab.
Safety Tests and Answers - Three different safety quizzes designed to check students' understanding of the importance of safety.

Robotics Lab Safety Inspection Sheet - Helps students to understand that they need to monitor the robotics classroom for safety.
Project Management

Many schools compete in robotic competitions; other schools are using MINDSTORMS to teach engineering. The links on this page provide students with resources that teach how to manage projects and solve engineering design problems.

*Project Planning Video* - The five minute Project Planning Video uses a combination of humor and examples to describe what a well planned project looks like.

*Engineering Process Video* - The five minute Engineering Process Video highlights the importance of research, planning, developing prototypes, and iterative testing when solving engineering design problems.

*Engineering Design Notebook* - Each student is required to keep an engineering design notebook. This two page handout describes what is kept in the notebook and what a daily log is.
Getting Started/Fundamentals/Project Management continued

Project Planning Documents

*Engineering Process Reference* - This three page set of handouts describes steps that engineers use to solve problems, provides a set of definitions for the word “engineering”, and describes the iterative nature of design.

*Team Building PDF* - The Team Building Lesson Set consists of four documents that describe guidelines that teams should use for the first team meeting, general ground rules for any team meeting, things to consider when building teams, and a description of roles on a robotics team: Project Management, Programming, Engineering, Documentation and Communications.

*Understanding the Problem PDF* - One of the keys to solving any problem is “Understanding the Problem”. This three page set of handouts consists of: Defining the Problem, Technical Research, and Creating a Design Specification.

*Brainstorming PDF* - Another key problem-solving step involves meetings where people brainstorm together to develop potential solutions. This three page set of handouts is broken into: a Brainstorming Primer, Things to Think About, and Brainstorming Tips. Each handout can be used individually or as part of a set.

*Planning Your Time PDF* - Time management is a crucial skill to develop. This four page handout uses a simple activity, planning a birthday party, to describe a critical skill set that everyone should learn: how to manage time.

*Design Reviews PDF* - Engineers conduct design reviews on a regular basis. This two page handout describes how to conduct weekly team design reviews, as well as preliminary and detail design reviews.
Getting Started/Fundamentals/Project Management continued

Project Planning Documents

Organizational Matrix Ideas PDF - This three page handout graphically shows three methods of organizing projects.

Recording Progress PDF - The recording progress tools offer the project manager three solutions that can be used to help team members to document the team’s progress toward the project goals.

Gantt Chart PDF - A Gantt chart provides a graphical illustration of a schedule to help plan, coordinate, and track specific tasks in a project. This one page handout is designed to teach students how Gantt charts work.

PERT Chart PDF - A PERT chart is a tool that graphically illustrates when parts of the project become due. The advantage of the PERT chart is that it shows which things must be completed in sequence and which things need to be completed simultaneously. This one page handout is designed to teach students how PERT charts work.

Preparing for a Competition PDF: Robotic competitions offer unique opportunities to teach students about time management, resource allocation, teamwork, and problem solving, all within a context that they find challenging but fun. The Preparing for a Competition handout is designed to support robotics teams as they plan for the competition.
Assessment Rubrics

Timely assessment is paramount in today's educational environment. A clear expectation of what is being assessed is a key to training students. Traditional assessments are provided in the curriculum; ie quizzes. The assessments found in this section are assessment rubrics for project-based learning. There are many other tools that a teacher may use, but this section provides some examples. Rubrics allow all stakeholders to see what is being measured.

**Writing Criteria Rubric** - Writing is a process and good writing requires several steps: brainstorming, outlining, prewriting, and editing. This is a simple rubric that check for those steps.

**Engineering Journal Rubric** - Explains to students what is expected in their engineering journals.

**Presentation Rubric** - Helps students determine what a good presentation should include.

**Request for Proposal Rubric** - Helps students to determine what is being evaluated in their RPF submission.

**Work Habit Evaluation** - This is a great tool for students to use to develop strong work habits.

**Workplace Competencies Rubric** - This rubric helps students to develop the skills that are valued by industry.

**Internal Design Rubric** - This evaluation tool helps students understand the expectations and preparation needed for an internal design review.

**External Design Review** - This evaluation tool helps students understand the expectations and preparation needed for an external design review.
Introduction to Programming

Thinking About Programming Video Set - The “Programmer and Machine” and “Planning and Behaviors” videos explain to students the role of the programmer and the machine, and how the programmer must learn to think like a machine in order to program robots. The videos are designed to explain programming concepts to beginners.

Behaviors and Flowcharts & Pseudocode Helper Pages - These are reference PDFs that students can use as study guides. The helper pages are designed to be guides to the topics.
Getting Started/Fundamentals/Introduction to Programming

**Thinking about Programming ROBOTC Rules Parts 1 & 2** - This video set builds the foundational knowledge that students must develop to begin to program using any syntax-based programming language. This video set introduces syntax concepts in a sequential and logical manner.

**Whitespace, Comments, and Reserved Words Helper Pages** - These are reference PDFs that students can use for reference or as study guides.

**Thinking About Programming and ROBOTC Programming Quiz** - One page quiz designed to check students' understanding of introductory programming topics.
NXT Hardware

NXT Sensors Slide Show - The Sensors Slide Shows introduce new learners to the MINDSTORM sensors that are part of the base set: the smart motors, touch, light, ultrasonic and sound sensors. Students will learn how they work and what type of feedback they record.

NXT Parts Identification - Graphically introduces students to the parts and how they work.

TECHNIC Hardware Primer - The TECHNIC hardware primer is designed to show students how LEGO TECHNIC parts can be used to build subassemblies that they can use in their robot designs.
Getting Started/Fundamentals/NXT Hardware

NXT Hardware

Using the NXT Slide Show - Assumes that the user has never used an NXT. There are 11 sections of the slide show that include: Introduction to the NXT, NXT Guide, Running a Program, Viewing Sensor Values, Introduction to Deleting, Deleting Programs 1, 2 & 3, Returning to the Main Menu, Changing the Volume, and Turning Off the NXT. This resource is a very valuable teaching tool for students just beginning to use the NXT.
Getting Started/Setup/NXT Setup

**NXT Setup Page**

*Build the REM Robot* - All of the ROBOTC for MINDSTORMS lessons use the “Robot Educator Model” plus attachments. This page provides building instructions in both a printable and digital format.

*Download Firmware Video and Quiz* - Students will learn what firmware is and how to download it to their robots. There is also a quiz designed to check student understanding.

*Download Sample Program and Quiz* - Students will learn to download their first ROBOTC program and test it on their NXT. The lesson also includes a PDF helper page as well as a quiz designed to check student understanding.
Getting Started/Setup/NXT Setup continued

NXT Setup Page Resources Continued

![Robot Educator Model Building Instructions](image)

![Setup Download Firmware](image)

![Setup Download Sample Program](image)

![Running a Program](image)

![Download Firmware](image)

![Download Sample](image)
The Introduction to Programming Curriculum is divided into four instructional units: Movement, Sensing, Variables and Functions, and Remote Control. Each unit contains a number of Lesson Sets designed to teach a particular concept. Each unit is organized around a "Unit Programming Challenge". The solution for the Unit Programming Challenge is taught as students move through the individual Lesson Sets.

The student will find other exercises designed to reinforce their programming skills or designed to give them a deeper understanding of the NXT hardware.
Introduction to Programming/Movement/Programming Challenge

The Movement Unit

The Movement Unit is taught using three Lesson Sets and a programming challenge. The Lesson Sets begin using sample code that is already included in ROBOTC. The first Lesson Set, Moving Forward, teaches students in a very lockstep manner what each line of code does while introducing them to moving motors for specific amounts of time. The second Lesson Set, Speed and Direction, explains motor power levels and how to reverse polarity. The second Lesson Set includes several “engineering labs” that the students will complete. The engineering labs place students in the role of engineer where they run their robots, measure results, iteratively test the results to determine reliability, and then extrapolate from their data set to predict new robot behaviors. The third Lesson Set, Improved Movement, begins to teach very important lessons about PID control, motor synchronization, and setting motor targets.

The Movement Unit also includes several programming challenges where students are challenged to solve simple movement programming challenges.

It will be important to remind students that although the initial work may seem easy, that the skills that they learn in the movement unit are foundational pieces that they must understand before they move to the sensing unit.
Introduction to Programming/Movement/Programming Challenge

Labyrinth Programming Challenge
Each Programming Unit (Movement, Sensing, Variables and Remote Control) contains a programming challenge that is designed to place the learning into an interesting context. In the “Movement” Lesson Set the programming challenge is the “Labyrinth Challenge”.

In this challenge, students will learn:
• Behavior based programming logic
• How to program their robots to accurately move forward, backward and turn
• The syntax rules related to programming using ROBOTC

Labyrinth Challenge Teacher Resources
Each programming challenge comes with a PDF that explains the rules to the challenge as well as a video solution. These Labyrinth challenge resources are pictured on the left.
Introduction to Programming/Movement/Moving Forward

The Moving Forward Lesson Set

Each Lesson Set is designed to teach a related set of programming concepts, and is designed to give students confidence opening a program and understanding what the individual lines of code mean. Every set is supported with the following video and print resources:

Program Dissection Video - Students are given a line by line description of the code used in the first sample program.

Timing Video - Timing offers the least accurate way of programming a robot to move from point to point. It is also the simplest method of programming. Students are introduced to wait-states.

Sumo-Bot Challenge - This fun challenge involves developing a programming/mechanical solution to push as many cans as possible out of a ring.
Additional Moving Forward Lesson Set Resources

Pictured at the left are two additional resources that accompany the Moving Forward Lesson Set.

The Moving Forward Quiz

The Moving Forward PDF - Each set of videos is accompanied by a PDF version of the script. The PDF version includes lots of pictures and text to describe everything that is shown in the videos. The PDF can be used either to accompany the video instruction or as a study guide.

Introduction to Programming/Movement/Speed and Direction

The Speed and Direction Lesson Set consists of two video lessons, two engineering challenges, three open-ended programming challenges, and a quiz. The resources are shown and explained in on the next page.
Introduction to Programming/Movement/Speed and Direction cont.

Motor Power Lesson
The first video lesson, Motor Power, teaches students how to change the power level on the robot. Students learn that as they change the power level, they are in effect changing the robot’s speed. They will also complete an Engineering Lab named “NXT Wait States/Power Level Investigation”. In this investigation, students program their robots at a variety of power levels and keep the amount of time the robot runs constant. In the investigation students identify if there is a proportional relationship between power levels and speed.

Turn and Reverse Lesson
In the “Turn and Reverse” lesson, students learn about polarity. The Lesson Set includes a 10 page PDF, shown at the right, designed to complement the videos. It also includes 3 open ended programming challenges. Two of the programming challenges are shown below: Robo-Slalom and Line Painter Bot. Students complete the lesson with an Engineering Challenge, where they are asked to iteratively test robot turns given a specific power level and a target angle and then use that data to predict what the “wait state” will be to complete other turns.
Introduction to Programming/Movement/Improved Movement

Improved Movement Lesson Set
The first half of the Improved Movement Lesson Set teaches students about PID speed control as well as how to use the Motor and Sensor Setup Window, the Debug Windows, and the NXT Devices window. The following resources are available to teach PID control:

*The Principles of PID Video* - This video teaches students what PID (Proportional Integral Derivatives) is and how it uses a closed loop control system to regulate speed control.

*The PID Control Video* - This video teaches students how to use ROBOTC's debugger to see the NXT’s speed regulation. Students will use the debugger windows to watch the internal feedback mechanisms built into the system.

*PID Speed Control Reference Page* - This two page handout explains PID, and shows how to monitor PID using ROBOTC's debugging window.
Introduction to Programming/Movement/Improved Movement cont.

Synchronized Motors Lesson Set
ROBOTC enables the programmer to synchronize the robot’s wheel movements. The Lesson Set contains the following resources:

Synchronized Motors Video - Teaches students how to use ROBOTC reserved words to control their robots.

Synchronized Motor Reference Page - Can be used as a handout for a study guide or as a reference page.

Synchronized Motors Engineering Lab - Students perform a set of iterative experiments where they test robot performance and collect data designed to promote understanding of robot synchronization.

nMotorEncoder Command and Target Distances Lesson Set
Encoders are used by the robot to count how many times a wheel on the robot turns. Students are introduced to encoders in this lesson allowing them to achieve better control of their robot.

The Target Distance Video - Teaches students how encoders work and gives them a naive understanding of the nMotorEncoderTarget function.

Improved Movement Lesson Set PDF - This PDF complements the videos used to teach the Improved Movement Lesson Set.
Introduction to Programming/Sensing

Sensing

The Sensing Unit consists of six Lesson Sets for the NXT. The lessons are designed to be taught in a sequential order from top to bottom. The lessons start with the touch sensor because of its simplicity, then move to the ultrasonic and sound sensors. If students skip one of the Lesson Sets, then they run the risk of missing a programming concept that was taught in a prior lesson. Do the lessons in sequence.
The Obstacle Course Programming Challenge

Students are required to use feedback from all sensors to complete the obstacle course. There is more than one way to solve this programming challenge, and individual students will develop their own programming style. For beginners, it will be important for the teacher to reinforce breaking robot behaviors down into their simplest parts.

The Obstacle Course Video - Shows a video solution to the programming challenge.

The Obstacle Course Handout - PDF that describes the challenge in terms of robot behaviors.
Introduction to Programming/Sensing/Touch Sensor

Sensing - Touch Sensor

The Wall Detection Lesson Set provides a comprehensive set of materials designed to introduce students to sensors. Students begin to solve the obstacle course by learning how a touch sensor works, how to configure the touch sensor using the motors and sensors setup windows, how to name sensors, and how to build a new program from scratch. As they learn about touch sensors, they will also learn about while loops, structures, boolean logic, conditional statements, ROBOTC reserved words, and how to use ROBOTC’s built-in debugger windows to see sensor feedback values. Students will have access to the following video and print resources:

**The Touch Versus Timing Video** - This video introduces the touch sensor and explains how it will be used to help solve the obstacle course challenge.

**The Configuring Sensors Video** - This video shows students how to configure sensors using ROBOTC’s Motors and Sensors Configuration Menu. It also shows students how to use the debugger screens to see sensor feedback.

**The While Loop Video** - Teaches students about conditional statements and how they control the while loop structure.

**The Putting It All Together Video** - This video is a step-by-step tutorial that begins with a blank program, then builds a program starting with configuring the sensors through setting up conditional statements and controlling motors.

**Touch and Light Sensor Attachment** - Instructional slide show that shows how to connect the sensor to the robot.

**Sense Plan Act Reference Page** - One page handout that describes the Sense-Plan-Act algorithm.

**While Loop Reference Page** - One page handout that describes the “while loop” structure.

**Boolean Logic Videos 1 & 2** - The first video teaches students how conditional statements work. The second teaches them about logical operators. There is also a Boolean Logic 3 page PDF that complements the videos.

**Robot Programming Challenges** - This Lesson Set includes three open-ended programming challenges that can be solved by students: the Can Bot Challenge, Robo500 Challenge (Level 1) and the RoboMower Challenge (Level 1).
Sensing - Ultrasonic Sensor

The Wall Detection Ultrasonic Sensor Lesson Set builds on what students learned in the Wall Detection Touch Sensor Lesson Set. This Lesson Set teaches students how ultrasonic sensors work, how to measure distance using sound, how to use thresholds to program the ultrasonic sensor, and how to re-configure the motors and sensor setup window. The lesson is also designed to review how programming loops work.

The Sonic Sojourn Video - This video describes how the ultrasonic sensor works.

The Threshold Reference Page - This handout describes what thresholds are, how they are used in programming, and how to calculate them for the ultrasonic sensor.

Random Numbers Reference Page - This handout describes how to write programs that use random number values using ROBOTC.

The Ultrasonic Sensor PDF Lesson - A printed version of the video designed to complement the video instruction.

Ultrasonic Programming Challenges - This Lesson Set includes three different programming lessons: The Robo500 Challenge (Level 2), The Table Bot (Level 1) Challenge, and the RoboMower Challenge (Level 2). The challenges are designed so that a teacher can use the same "setup" for multiple challenges. Some challenges get progressively more difficult, while other challenges are designed so that the student completing it is able to quickly solve the same challenge with a different sensor.

Ultrasonic Quiz - Designed to check students understanding of the concept.
Introduction to Programming/Sensing/Ultrasonic Sensor

Sensing - Ultrasonic Sensor Resources
Sensing - Encoders

In the Forward for Distance Lesson Set, students learn advanced features of the nMotorEncoder function, allowing them to make very precise movements. The NXT smart motors have encoders built into them that allow the robot to move point-to-point in a very accurate manner. In the first video, students learn a simple method of moving point-to-point. In the second and third video set, students learn how to use the nMotorEncoderTarget function.

Forward for Distance Video - This lesson teaches students about the encoders built into the NXT smart motors, and the nMotorEncoder function.

Advanced Target Distance Part 1 Video - This lesson teaches students about the difference between the nMotorEncoder function and the nMotorEncoderTarget function. It also explains how the nMotorEncoderTarget function is programmed. Students will also learn about a new function that watches the motors “run state”.

Advanced Target Distance Part 2 Video - This lesson shows students how to write the nMotorEncoderTarget program.

Boolean Logic Videos - Comparison operators and logical operators are key elements that all students must know. This video set is included at several spots throughout the curriculum for student review.

Encoders Reference Page - The encoders reference page is a handout that can be used as a reference or study guide.

Turning with Encoder Engineering Lab - The Encoder Engineering Lab has students calculate encoder counts to turn specific angles. This data will help them with all programming moving forward.

The Robo500 (Level3) Programming Challenge - Students have seen this challenge before. Now they will be able to apply what they’ve learned in the encoder lessons to solve the problem more perfectly.
Introduction to Programming/Sensing/Encoder

Sensing - Encoder Resources
Introduction to Programming/Sensing/Light Sensor

Sensing - Light Sensor

The Forward Until Dark Lesson Set introduces students to how the light sensor works, how to troubleshoot programs that use the light sensor, and how to calculate threshold values for light sensors. The Lesson Set has the following resources:

The Light Sensor Video - The light sensor video reviews how the touch and ultrasonic sensors work and then describes how the LEGO light sensor works. The video also describes things to consider to optimize how the sensor works.

The Thresholds 201 Video - This lesson describes how to calculate thresholds for the light sensor. It also demonstrates the physical steps that a programmer needs to do to calculate a light sensor threshold value.

The Wait for Dark Video - This lesson begins with a prior program, then adapts the program to use the light sensor. Students will learn how to configure the Motors and Setup Configuration menu for the light sensor, and then are given directions that allow them to complete the program.

The Threshold and Random Numbers Reference Pages - These handouts are the same handouts used in the previous lesson and are useful to students still learning these foundational programming concepts.

The Forward Until Dark Printable PDF - This document is a printed version of the three video lessons in the Forward Until Dark Lesson Set.

The Programming Challenges - There are five programming challenges with this lesson: The Robo500 Challenge (Level4), the Table Bot Challenge (Level 2), the Line Runner Challenge (Level1), the Minesweeper Challenge, and the Firefly Challenge (Level1). All students may not complete all programming challenges.

Forward Until Dark Lesson Set Quiz - A quiz to check student's understanding.
Introduction to Programming/Sensing/Light Sensor

Sensing - Light Sensor Resources
Sensing - Light Sensor LineTracking

The Line Tracking Lesson Set breaks line tracking, a seemingly complex behavior, into a set of simple robot behaviors. It introduces students to a new structure, the if-else structure. Students learn to use multiple sensor feedback when they first use feedback from internal timers and light sensors to control how far their robot moves. Once they solve that problem, they then use feedback from the NXT smart motors and the light sensor to determine how far their robot travels. At the end of this Lesson Set, students will have all of the tools that they need to complete the obstacle course challenge. This Lesson Set uses the following video resources:

**The Line Tracking (Basic) Video** - This video teaches students how line tracking works, and how to break complex behaviors into simple behaviors. It also introduces the students to a naive way to solve the line track programming problem using while loops. This solution works, but has serious limitations.

**Line Tracking (Better) Video** - This lesson teaches students how to use feedback from multiple sensors with their robot. They discover that their naive solution using only “while loops” doesn’t always work. The video explains if/else structures, explaining the code as it talks students through the solution.

**The Line Track Timer Video** - This video teaches students how to use the internal timers built into the NXT to control how long the robot does something. This is a very valuable tool that the student will be able to use in many other situations.

**The Line Tracking Rotation Parts 1 & 2 Videos** - Teaches the student to use feedback from the encoders built into the NXT smart motors to determine when to stop tracking a line.
Introduction to Programming/Sensing/Light Sensor/Line Tracking

Sensing - Light Sensor Line Tracking Video Resources

If-Else Reference Page - Students will use the if-else structure from this point forward in their programming career. This is a one page reference sheet that shows them how it works.

Switch Case Statement - Another option for programmers that have to control multiple conditions is the switch case. This resource is a multi-page handout that students can reference when using switch cases.

Timer Reference Page - The timer reference page demonstrates specific reserved words used in ROBOTC to control the internal timers built into the NXT.

Line Tracking Programming Challenges - This Lesson Set includes three open-ended challenges designed to support the lesson: the MouseBot Challenge, the Minefield Challenge (Level 1), and the Robocci Challenge (Level 2). All students may not complete every programming challenge.

Line Track Quiz - The line track quiz is included to check student's understanding.
Introduction to Programming/Sensing/Light Sensor/LineTracking

**Sensing - Light Sensor LineTracking Print Resources**

### Line Tracking Basic Lesson

For the student to master the line tracking lesson, we do something by following a wire using the robot. The teacher will show the students how to use the light sensor to track the line. The students will then be asked to work in groups to move around the classroom.

### If-else Statement

The if-else statement is used to select a path. If the statement is true, the robot will move forward. If the statement is false, the robot will move backward. The students will then be asked to program the robot to move forward or backward based on the input.

### Switch Case

The switch case statement is used to select a path. The students will then be asked to program the robot to move forward or backward based on the switch case.

### Timers

Timers are used to control the duration of actions. The students will then be asked to program the robot to move forward or backward based on the timer.

### Line Tracking Quiz

1. Why was the program written using a timer?

2. What is the purpose of the light sensor?

3. How can you control the direction of the robot?

### Mindfield Level 1

**Challenge Description:**

To complete the mindfield level 1, you must navigate the robot through the course without touching any obstacles. The level is divided into two main parts: the start line and the finish line. The robot must pass through both parts to complete the level.

### Outback Level 1

**Challenge Description:**

To complete the outback level 1, you must navigate the robot through the course without touching any obstacles. The level is divided into two main parts: the start line and the finish line. The robot must pass through both parts to complete the level.

### Robocat Level 2

**Challenge Description:**

To complete the robocat level 2, you must navigate the robot through the course without touching any obstacles. The level is divided into five main parts: the start line, the first obstacle, the second obstacle, the third obstacle, and the finish line. The robot must pass through all parts to complete the level.
Introduction to Programming/Sensing/Sound Sensor

In the Speed Based on Sound Lesson Set, students are challenged to map the values of the sound sensor to the motor powers so that they can control the motors speed based on sound. This lesson reinforces the idea that a programmer needs to continually update the sensor value if they want to use the sensor value to control something real time. This is a very important concept that beginning programmers often miss. In this lesson, students also practice using the internal timers and the reserved words that are associated with it. The Lesson Set includes: two videos, a 9 page PDF that aligns with the video, and a quiz.
Sensing - Variables and Functions

The Variables and Functions Unit contains four Lesson Sets: Automatic Thresholds, Line Counting, Patterns of Behaviors and Debugging. Each of these Lesson Sets teaches a major programming concepts. Automatic Thresholds introduces the concept of variables, variable types, how to name variables, and how to manipulate them. The Line Counting Lesson Set builds on the variable concept and teaches students how they can manipulate the value of variables. Patterns of Behaviors introduces functions, how to pass parameters in functions, and how functions can be used to simplify programs. And finally, the Debugging Lesson Set, offers two lessons on how to use ROBOTC’s debugger to troubleshoot programs.

The previous programming units, Movement and Sensing, introduce basic robot programming concepts. These lessons begin to introduce basic programming concepts!
Sensing - Variables and Functions/Programming Challenge

The Warehouse Programming Challenge asks students to develop a customizable program that allows them to program their robot to move to any spot in the warehouse. As students move through the Automatic Thresholds, Line Counting and Patterns of Behavior Lesson Sets they will gain the skills needed to solve the programming challenge.

There are two resources available to introduce the programming challenge: the Warehouse Challenge PDF and the Warehouse Challenge Solution Video.
Sensing - Variables and Functions/Automatic Thresholds

The Automatic Thresholds Lesson Set introduces a variables; a very large concept in computer programming. The step-by-step guided lesson will allow students to solve this programming problem, but the concept will need to be revisited multiple times before students become competent using variables.

**Values & Variable Video** - This video introduces variables, variable types, and how to name variables.

**Variables and Thresholds Video** - This video reviews how to calculate thresholds for the light sensor, the programming problem that students will solve in this Lesson Set, and defines the initial variables used to solve this problem.

**Programming with Variables Video** - This video walks students through the automatic threshold calculation problem beginning with configuring the light and touch sensor and continuing through the initial setup of the problem.

**Variables and the Debugger Video** - This video shows students how to use ROBOTC’s debugger to find the problem in their initial programming solution. They will discover how fast computer processors work, and be shown a solution.

**Threshold Calculations Video** - This video shows students how to do calculations using variables and how to display values to the NXT display screen.

**The Boolean Logic Video Set** - Students have seen this set in an earlier lesson; this is a good time to review.

**Variable and Data Types Reference** - This is a multiple page handout that students can use as a study guide.

**Threshold Values Reference** - One page reference guide that students can use to review threshold values.

**Display Text Reference** - This multiple page lesson shows students step-by-step how to display text on their screens.

**Firefly Challenge (Level2)** - This challenge requires students to find a light value and report the value to their screen.

**Automatics Thresholds PDF** - A 36 page PDF that covers what was taught in the video lessons in a print format.
Programming/Variables and Functions/Automatic Thresholds

Sensing - Variables and Functions/Automatic Thresholds - Print Resources

ROBOTC Display Test Features

Firefly Bot Level 2

Thresholds

Side Mounted Touch

Sensor Attachment
Programming/Variables and Functions/Counting

Sensing - Variables and Functions/Counting

The Line Counting Lesson Set introduces students to using variable to store values so that their robots can count. It appears that this would be a very simple task to implement, but there are multiple small problems to consider. Students will learn more about processors and implications of processing speed that they may not have thought about. Students will learn to setup a counter variable, when to count, how to count, and how to stop counting in a program.

**The Counting Video** - This video introduces students to reassigning values to variables allowing the robot to count.

**Line Counting Part 1 Video** - This video helps student to identify the problem as well begin to write their code.

**Line Counting Part 2 Video** - This video introduces students to using the “breakpoint” function built into ROBOTC’s debugger allowing students to see how fast the robot’s processor loops.

**Line Counting Part 3 Video** - Students are shown how to create a set of variables and turn them on and off helping them to count or not count at the correct spots in their program.

**Line Counting Part 4 Video** - The last video in the Lesson Set shows students how they can control when the robot stops.

**Line Counting PDF Lesson Set** - A printed version of everything that was found in the Line Counting video lessons.

**Programming Challenges** - Students will begin to separate at this point. There are five programming challenges included in this Lesson Set. The programming challenges are more difficult at this point and not every student will complete every challenge. The challenges are: PipeBot Challenge (Level1), Quick Tap Challenge, The Pipe Bot Challenge (Level2), the Line Runner Challenge (Level 2), and the Auto Attendance Challenge.

**Line Counting Lesson Set Quiz** - A quiz designed to check students understanding of these concepts.
Programming/Variables and Functions/Counting
Sensing - Variables and Functions/Counting - Video Resources
Programming/Variables and Functions/Counting

Sensing - Variables and Functions/Counting - Printed Resources
Sensing - Variables and Functions/Patterns of Behaviors

In the Patterns and Behaviors Lesson Set students learn to program using functions. Functions are very powerful organizers used by programmers that program robots using behavior based programming strategies.

**Behaviors Video** - Introductory video that explains the value of programming using functions.

**Creating and Using Functions Video** - In this video students learn the relationship between task main and a function. They learn to declare functions and use them in the task main section of their program.

**Variables and Functions Part 1 Video** - In this video students learn how to take code from other programs and turn them into functions and then they will take those functions and use them in an easier to read program. They also learn about the scope of variables and how to create global variables.

**Variables and Functions Part 2 Video** - In this video students continue to write the solution to the “warehouse” challenge using their newly found function tool.

**Variables and Functions Part 3 Video** - In this video students learn to pass parameters making their functions much more powerful.

**Functions Reference Guide** - This is a multipage handout that shows students how functions work.

**Global Variables Reference Guide** - This handout explains what a global variable is and how it can be used.

**Programming Challenges** - This Lesson Set include two advanced programming challenges: the Robot Acceleration Challenge, and the Minefield Challenge (Level2).

**The Patterns of Behaviors Lesson Set Handout** - This is a 40 page PDF that complements the videos in this Lesson Set.
Programming/Variables and Functions/Patterns of Behaviors

Sensing - Variables and Functions/Patterns of Behaviors - Video Resources
Programming/Variables and Functions/Patterns of Behaviors

Sensing - Variables and Functions/Patterns of Behaviors - Print Resources

Patterns of Behavior Behaviors

Patterns of Behavior does

Global Variables

Functions

Challenge

Mindfield Level 2

Robot Acceleration Challenge

Sensing - Variables and Functions/Patterns of Behaviors

Teaching ROBOTC® for LEGO® MINDSTORMS®

© 2009 Carnegie Mellon Robotics Academy
Sensing - Variables and Functions/Debugging

The Debugging Lesson Set consists of a set of videos designed to highlight ROBOTC’s debugging screen. The first video, Debugging Techniques has the student open up a sample program that shows them how to open up the expert mode to use the higher level debugging tools available through ROBOTC. Students will learn how to use the step into, step over, step out, breakpoint, and clear all features found on the debug start menu. The second video set, Printing to Screen, highlights the advantage of using the on screen NXT display so that you can see values as you troubleshoot your code.
Programming/Remote Control

The Remote Control Unit teaches students how Bluetooth works, how to troubleshoot Bluetooth, and how to program their remote control unit. ROBOTC allows two Logitech remote control units to control one robot. All of the lessons in this curriculum are designed for one remote control unit. For more information on controlling one robot with two remote control units refer to the built in help in ROBOTC.
Programming/Remote Control/Programming Challenge

The Remote Control Robot Soccer Programming Challenge provides a fun engineering/programming challenge that students of all levels enjoy. In the Remote Control programming unit, students will learn how to program the joysticks and buttons on their Logitech remote control units to produce desired behaviors like "kick the ball" or "drive straight". They already know how to program using the LEGO sensors, but there are a whole array of other third party sensors available that make the robot soccer challenge a great engineering/programming challenge. Two sensors that come to mind quickly are the compass sensor and the IR sensor. The soccer challenge is a very good challenge at all levels middle school through adult hobbyist.

The following resources are available to preview the challenge:
Remote Control Basics

The Remote Control Basics Lesson Set teaches students everything that they need to know to combine autonomous and remote control programming. This is a very powerful addition to the LEGO’s already powerful NXT hardware. This Lesson Set contains two Remote Control Engineering Labs where students will be guided step-by-step toward a successful implementation of a combination of remote control and autonomous programming. The following resources are available to support this Lesson Set:

**How Remote Control Communications Works Video** - This video describes the communications between the Logitech Remote Control, the computer, and the NXT based robot.

**Using Remote Control Video** - In this lesson, students learn about include files, the new functions that are made available via the include file, and how the new functions map to the Logitech remote control.

**Improving Remote Control Video** - This lesson teaches students how to improve the joystick control via variable math and creating a dead zone on their remote control.

**Using Buttons Video** - This lesson teaches students how to program the buttons on the Logitech controller.

**Install the Joystick Controller Reference** - This is a reference PDF that helps to install and troubleshoot the Logitech remote controller installation.

**Remote Control Basics Engineering Lab** - This lab provides step-by-step instructions designed to help a student to learn how to program the Logitech remote controller joysticks to control the NXT.

**Remote Control Buttons Engineering Lab** - This lab provides step-by-step instructions on how to write a program that enables the Logitech remote controller buttons to control the NXT.

**Gripper Building Instructions** - A set of instruction for students to use to build a LEGO gripper.
Using Bluetooth
The Using Bluetooth Lesson Set is designed to help a first time user connect their computer to their NXT. There are four resources available in this section that have multiple components designed to ensure success.

*The Setting Up Bluetooth Video Set* - This Lesson Set focuses on answering the following questions: What is Bluetooth? How do I rename my NXT so that it is easy for Bluetooth to recognize? and How do I ensure that my NXT is visible to my computer?

*The Connecting Via Bluetooth Video* - This video demonstrates step-by-step how to connect via Bluetooth using ROBOTC.

*The Troubleshooting Bluetooth Video Set* - This troubleshooting guide consists of six things to consider if your Bluetooth connection isn't working.

*Installing your Bluetooth Adaptor Reference Guide* - A one page handout that describes the Bluetooth adaptor.
Programming/Remote Control/Using Bluetooth

Using Bluetooth Resources

Setting up Bluetooth

Setting up Bluetooth

Setting up Bluetooth

Using Bluetooth

Using Bluetooth

Troubleshoot Bluetooth

Troubleshoot Bluetooth

Troubleshoot Bluetooth

Troubleshoot Bluetooth

Step #1: Rebooting the Computer

Step #2: Pulling the Battery out of the NXT

Step #3: Moving the USB Bluetooth Adapter
How Robotics Aligns with Standards

This section describes how robotics as a content area aligns with National Science, Mathematics, Technology, and Communications Standards. Below you will see the format that we use to show how robotics education aligns to national science, technology, engineering, and mathematics (STEM) standards.

<table>
<thead>
<tr>
<th>Standard</th>
<th>Robotics Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>On the left is a description of the standard or particular point of the standard that is addressed through robotics.</td>
<td>On the right is a description of how robotics in general and this curriculum in particular addresses this standard.</td>
</tr>
</tbody>
</table>

Science Standards Addressed
From the National Science Education Standards (NSES)

Systems, Order, and Organization

The natural and designed world is complex; it is too large and complicated to investigate and comprehend all at once.

A system is an organized group of related objects or components that form a whole.

The goal of this standard is to think and analyze in terms of systems.

Science assumes that the behavior of the universe is not capricious, that nature is the same everywhere, and that it is understandable and predictable.

Prediction is the use of knowledge to identify and explain observation, or changes, in advance. The use of mathematics allows for greater or lesser certainty of predictions.

Order is the behavior of units of matter, objects, organisms or events in the universe – can be described mathematically.

Types and levels of organization provide useful ways of thinking about the world

Robots are excellent examples of systems, with many heterogeneous components interacting in organized, methodical ways to achieve results as a whole that they could not have achieved separately.

Examples include:
- Navigation systems (e.g., sensor tells the robot where it is, programmable controller tells the robot how to interpret this information, motors move in order to achieve the desired result)
- Sensing systems (electrical, mechanical, and programming elements of a sensor)
- Power & transmission systems (motor, axle, gear, wheel)
- Manipulator systems
- Lifting systems, vision systems, etc.

Each system can be broken down into subsystems.

Robotics technology is built upon a series of behaviors that can be measured mathematically and are understandable and predictable.

There are many examples that are easy for students to manipulate and understand:
- Gears and mechanical advantage
- Sensors and electronic control
- Wheel diameter and its effect on distance traveled
- Rotation sensor readings and robot path planning
**Science Standards continued**  
From the National Science Education Standards (NSES)

### Evidence, Models and Explanation

Evidence consists of observations and data on which to base scientific explanations. Using evidence to understand interactions allows individuals to predict changes in natural and designed systems.

Models are tentative schemes or structures that correspond to real objects, events, or classes of events that have explanatory power. Models help scientists and engineers understand how things work. Models take many forms, including physical objects, plans, mental constructs, mathematical equations and computer simulations.

Scientific explanations incorporate existing scientific knowledge and new evidence into logical statements. Terms like “hypothesis,” “model,” “law,” “theory,” and “paradigm” are used to describe various scientific explanations.

The investigations included in this curriculum allow students to collect evidence to investigate scientific principles. Robots physically demonstrate many scientific concepts to make them more clear and understandable.

Examples include:
- Electronics and basic circuitry, which can be demonstrated using touch sensors and the NXT power supply
- Gear trains, which demonstrate the ability to mathematically predict mechanical advantage and speed.
- Light sensors, which can detect infrared as well as visible light

### Constancy, Change and Measurement

Although most things are in the process of becoming different – changing – some properties of objects and processes are characterized by constancy; the speed of light, the charge of an electron, the total mass plus energy of the universe.

Energy can be transmitted and matter can be changed. Nevertheless, when measured, the sum of energy and matter in the system, and, by extension, the universe, remains the same.

Mathematics is essential for accurately measuring change.

Different systems of measurement are used for different purposes.

Scale includes understanding that different characteristics, properties, or relationships with a system might change as its dimensions are increased or decreased.

Rate involves comparing one measured quantity with another measured quantity, for example, 60 meters per second.

Robots rely on the use of many innate constants in their basic operation. Ultrasonic sensors, for instance, calculate distance based around an assumed value for the speed of sound.

In calculating the distance a robot travels per spin of its motor, fundamental mathematical relationships govern the elements of change and constancy between the different factors involved. For example, the ratio between the diameter and circumference of the wheel is constant ($C=\pi d$). On the other hand, a robot doesn’t always need to use the same wheels – they can change – yet, no matter what the size of the wheel, the distance traveled per turn of the wheel remains proportional.

Measurement is fundamental to all aspects of robotics, from matching dimensions of parts to ensure that they can connect properly, to measuring how far your robot went, to measuring how well a prediction matched a result.
### Science Standards continued
From the National Science Education Standards (NSES)

#### Evolution and Equilibrium

<table>
<thead>
<tr>
<th>Evolution is a series of changes, sometimes gradual and sporadic, that accounts for the present form and function of objects, natural systems and designed systems. The general idea of evolution is that the present arises from materials and forms of the past.</th>
<th>Every robot design has a story. As they build and modify their robot designs, students can trace the evolution of their creation as they adapt it in different ways that allow it to complete different tasks, building upon lessons learned from their previous designs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equilibrium is a physical state in which forces and changes occur in opposite and off-setting directions. For example, opposite forces are of the same magnitude, or off-setting changes occur at equal rates.</td>
<td>Equilibrium appears in many different forms as a design factor that students will encounter in designing their robots. For example, a robot’s top speed is an equilibrium point between the physical force of friction and the force generated by the motor.</td>
</tr>
</tbody>
</table>

#### Form and Function

<table>
<thead>
<tr>
<th>Form and function are complementary aspects of objects, organisms, and systems in the natural and designed world.</th>
<th>When designing robots, form always follows function.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Whether the design decision involves using large versus small wheels, making the motor power high versus low, or selecting the sensing device the robot will use, all decisions are based on what the robot is expected to do: its function. All of these decisions will affect the final shape of the robot: its form.</td>
<td></td>
</tr>
</tbody>
</table>

#### Science as Inquiry – Content Standard “A”

<table>
<thead>
<tr>
<th>As a result of activities in all grades, all students should develop:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Abilities necessary to do scientific inquiry</td>
</tr>
<tr>
<td>• Understanding about scientific inquiry</td>
</tr>
<tr>
<td>Students should be engaged in activities that:</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>• Begin with a question</td>
</tr>
<tr>
<td>• Allow them to perform an investigation</td>
</tr>
<tr>
<td>• Gather evidence</td>
</tr>
<tr>
<td>• Formulate an answer to the original question</td>
</tr>
<tr>
<td>• Communicate the investigative process and results</td>
</tr>
<tr>
<td>The guided investigations in Robotics Engineering are targeted at specific relevant questions about robotics technologies and concepts that lead to rich exploratory experiences.</td>
</tr>
<tr>
<td>Some investigations focus on specific portions of the inquiry process, such as evidence-gathering or hypothesis evaluation. Others begin with a question and seek an answer using general inquiry processes.</td>
</tr>
<tr>
<td>Explanation and evaluation are primary abilities applied in answering questions, not simply calculations or summarization.</td>
</tr>
</tbody>
</table>
### Science Standards continued
From the National Science Education Standards (NSES)

#### Physical Science – Content Standard “B”

<table>
<thead>
<tr>
<th>As a result of activities in all grades, all students should develop an understanding of:</th>
<th>Robotics is able to demonstrate many applied physical concepts. Here are a few examples:</th>
</tr>
</thead>
</table>
| • Properties and changes of properties in matter  
• Motions and forces  
• Transfer of energy | • Mechanical advantage (gears)  
• Basic circuitry (sensor operation)  
• Digital and analog electronics (sensors)  
• Light (lamp, light sensor)  
• Sound (ultrasonic, sound sensors)  
• Speed (motors)  
• Friction (robot movement) |

By using simple objects, such as rolling balls and mechanical toys, students can move from qualitative to quantitative descriptions of moving objects and begin to describe the forces acting on the objects.

Understanding of energy will include light, heat, sound, electricity, magnetism, and the motion of objects.

#### Science and Technology – Content Standard “E”

| As a result of activities in all grades, all students should develop:  
• Abilities in technological design  
• Understandings about science and technology | Robotics is the premier example of the marriage of science and technology, especially as related to the solving of problems or human needs. |
| --- | --- |

Students should begin to differentiate between science and technology.

In the middle school years, scientific investigations can be completed by activities in which the purpose is to meet a human need, solve a problem, or develop a product rather than explore ideas about the natural world.

Every investigation students conduct with the robot is motivated by the need to advance the performance of the robot in order to meet performance criteria, connecting the “need to know” with the “ability to do”.

Robotics is able to demonstrate many applied physical concepts. Here are a few examples:

• Mechanical advantage (gears)  
• Basic circuitry (sensor operation)  
• Digital and analog electronics (sensors)  
• Light (lamp, light sensor)  
• Sound (ultrasonic, sound sensors)  
• Speed (motors)  
• Friction (robot movement)
## Mathematics Standards Addressed

*From the National Council of Teachers of Mathematics (NCTM)*

### Numbers and Operations

- Understand numbers, ways of representing number, relationships among numbers and number systems.
- Understand meaning of operations and how they relate to one another.
- Compute fluently and make reasonable estimates.

Robotics uses numbers and operations in nearly all lessons, for example:

- Calculating distance with rotational sensors (equations, equalities)
- Gears, gear ratios and speed (ratios and proportions)
- Light sensors and threshold (inequalities)
- Wheel circumference, radius and diameter (geometric relationships)

### Algebra

- Represent and analyze mathematical situations and structures using algebraic symbols.
- Use mathematical models to represent and understand qualitative relationships.
- Analyze change in various contexts.

Robotics lessons that involve algebra include the following:

- Conditional statements (inequalities)
- Programming sensors and thresholds (inequalities)
- Measuring turns (equalities, solving equations)
- Gears and speed (ratios, direct and indirect proportionality)
- Passing parameters in functions

### Geometry

- Precisely describe, classify, and understand relationships among types of two and three-dimensional objects using their defining properties.
- Specify location and describe spatial relationships using coordinate geometry and other representational systems.

Robotics situations involving geometry include:

- Wheel rotations and circumference (diameter, circumference)
- Identifying locations in order to program a robot to move from point to point (connected path segments)
- Interlocking gears and gear ratios (discrete combinations of radii)

### Measurement

- Understand measurable attributes of objects and the units, systems, and processes of measurement.
- Apply appropriate techniques, tools and formulas to determine measurements.

Understanding the significance and meaning of measurements are central to the understanding of robotics:

- Distance the robot travels (linear measurement, meter stick)
- Amount a motor turns (angular measurement)
- Directional change of the robot (angular measurement, protractor)
- Speed of the robot (rate measurement, meter stick, built-in timer)
- Physical quantities measured by sensors (touch, sound, light, distance)
- Detectable region of a sensor (ultrasonic sensor, meter stick, 2D graph paper)
**Mathematics Standards** continued
From the National Council of Teachers of Mathematics (NCTM)

**Problem Solving**

- Build new mathematical knowledge through problem solving.
- Solve problems that arise in mathematics and other contexts.
- Apply and adapt a variety of appropriate strategies to solve problems.
- Monitor and reflect on the process of problem solving.

In the lessons, there are both guided and open-ended design problems that involve designing, building, and programming needed to create autonomous robots.

- How do I get a robot to move a certain distance? (solved through measurement and the verification and use of a proportionality relationship)
- What does the sound sensor measure? (solved by graphing the sensor readings with tones of varying volume and pitch, then seeing which one indicated an orderly relationship)

**Reasoning and Proof**

- Recognize reasoning and proof as fundamental aspects of mathematics.
- Make and investigate mathematical conjectures.
- Develop and evaluate mathematical arguments and proofs.
- Select and use various types of reasoning and methods of proof.

Reasoning in robotics comes in many different forms, including the following:

- Experimental reasoning, proof using measurements and physical evidence (Wheels and Distance)
- Reasoning using equations, proof by solving (Measured Turns)
- Reasoning about graphs, proof by observing trends (Frequency and Amplitude)

**Communications**

- Organize and consolidate their mathematical thinking through communications.
- Communicate their mathematical thinking coherently and clearly to peers, teachers, and others.
- Use the language of mathematics to express mathematical ideas precisely.

Activities and Engineering Labs requires documentation that allows students to reflect on what they have accomplished or experienced, and describe it or some aspect of it in their own words to someone else. Emphasis is placed upon explaining reasoning in addition to showing calculations.

- The Engineering Design Challenge includes opportunities for students to communicate with their peers and teachers what they have learned and accomplished.

**Connections**

- Recognize and use connections among mathematical ideas.
- Understand how mathematical ideas interconnect and build on one another to produce a coherent whole.
- Recognize and apply mathematics in contexts outside of mathematics.

One of the strongest features of using robotics to teach math, science, engineering, technology and communications is its ability to make links between multiple disciplines. Students are able to take what they know and connect it to what they are learning, synthesizing new knowledge as they continue.
# Technology Standards Addressed
From the International Technology Education Association (ITEA)

## The Nature of Technology

1. Students will develop an understanding of the characteristics and scope of technology.
2. Students will develop an understanding of the core concepts of technology.
3. Students will develop an understanding of the relationships among technologies and the connections between technology and other fields of study.

All robotics activities provide excellent hands-on exposure to technology in use and development.
- Robotics activities feature linkages to advanced technologies that allow students to connect their designs to real-world needs and solutions
- Successful robot operation revolves around the application of systems concepts to make sensors, actuators, and other components work together
- Design processes take into account goals, resources, and trade-off factors to achieve optimal results
- Technology exists in proper context alongside applications in science, math, and engineering
- Several different technologies (e.g. desktop computer, USB/Bluetooth peripheral interface, mobile robotics controller, electromechanical sensors and actuators) are routinely used together in the operation of the MINDSTORMS robot system, and all are necessary for it to work

## Technology and Society

6. Students will develop an understanding of the role of society in the development and use of technology.

Robotics Engineering Design Challenges are linked to real world problems that use similar technologies to accomplish tasks that fulfill a social and/or economic need in the real world. For example:
- For instance the robot mining project that is part of this curriculum addresses a real problem that mining industries face daily.
- Some robot activities focus specifically on Human-Robot Interaction (HRI), an emerging field dealing specifically with psychological and design issues relating to the use of robots in human environments.

## Design

8. Students will develop an understanding of the attributes of design
9. Students will develop an understanding of engineering design
10. Students will develop an understanding of the role of troubleshooting, research and development, invention and innovation, and experimentation in problem-solving.

Students gain first-hand experience with developing a functional robotic system in many activities, including:
- The Warehouse Programming Challenge
- The Robot Mining Challenge
- Teacher assigned robot problems
Technology Standards continued
From the International Technology Education Association (ITEA)

Abilities for a Technological World

11. Students will develop the ability to apply the design process
12. Students will develop the ability to use and maintain technological products and systems

Students will apply design processes continually while working with and developing the robot. Here are some basic examples:
- Solving engineering design problems
- Robotics Competitions

In the course of working with the robot, students will be responsible for the maintenance of their robots:
- Mechanical soundness (the robot needs to be kept in good enough condition to perform its tasks daily)
- Organizing information (students must keep good records to know how to use systems they initially designed days or weeks earlier)
- Troubleshooting (robots have problems—often—and students must be able to identify and solve these issues as they arise)

Students will work with many important technologies as part of the operation of the NXT system:
- Electronic microcontrollers (NXT)
- Desktop/laptop computer and software (NXT Programming Software, word processor for write ups, spreadsheets for data graphs)
- Peripheral interfaces (USB or Bluetooth wireless)
- Electromechanical systems (touch, light, rotation, sound, ultrasonic sensors)
- Electromechanical actuators (Interactive Servo Motors)

The Designed World

16. Students will develop an understanding of and be able to select and use energy and power technologies
17. Students will develop an understanding of and be able to select and use information and communications technologies
18. Students will develop an understanding of and be able to select and use transportation technologies
19. Students will develop an understanding of and be able to select and use manufacturing technologies

The NXT robot is an excellent example and integrator of many different designed technologies working together as a coordinated system.
- Power sources (battery technologies – rechargeable Lithium-Ion vs. disposable alkaline)
- Vehicle systems (all the robot’s systems must work together in order to make it mobile, a viable platform for transportation of goods or as a platform to perform other work)
- Manufacturing and prototyping (robot must be built and modified using appropriate materials, plans and tools)
- Structural soundness and stability concepts are integral to the design of the robot’s physical form.
- Communication between system components (desktop to NXT, sensors to NXT, NXT to motors)
- Communication technologies (USB vs. Bluetooth)
## Reading, Writing, Listening, and Presenting
Communications skills developed through robotics

Engineering does not exist in a vacuum; it is highly interdisciplinary and highly social. Teamwork is a central foundation of Engineering, and communication is essential to smooth functioning of any engineering team. Students will find that highly developed communication skills are an absolute necessity for success.

### Communications skills applied when working with Robots

<table>
<thead>
<tr>
<th>Situation or Activity</th>
<th>Communications Concepts Applied</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintain Engineering Design Notebook</td>
<td>Organization of information</td>
</tr>
</tbody>
</table>
| Reach consensus on which of several student-proposed designs the team will build | Teamwork and group communication skills  
  • Running and participating in meetings  
  • Building consensus |
| Compose a compelling proposal to convince a (virtual) sponsor that their robot’s development is worth funding | Formal persuasive composition  
  Integrate self-conducted research into a piece that is not purely expository  
  Technical writing  
  • Explaining technical decisions and implementations to an audience that is not necessarily technically inclined |
| Document the team’s progress and accomplishments daily | Documentation and accounting for time, resources, and progress |
| Undergo review and integrate feedback from experts | Review and feedback processes  
  Learning to accept and respond to criticism |
| Choose from a variety of representations to best illustrate and communicate a point | Use many different formats of both technical and non-technical information, across different media:  
  • Graphs  
  • Charts  
  • Tables/Matrices  
  • Photographs  
  • Sketches  
  • Timelines  
  • PERT and Gantt Charts  
  • Multimedia presentation  
  • Text |
| Use Bluetooth to communicate between two NXTs | Electronic communication paradigms (as opposed to human)  
  Networking and connected devices |
| Establish a standard “language” for communicating between two NXTs | Electronic communication  
  Basic principles of communication  
  • Necessity of shared language  
  • Encoding and interpretation |
| Programming the robot | Communicate instructions explicitly to a robot using a “foreign” language |
**Reading, Writing, Listening, and Presenting continued**
Communications skills developed through robotics

<table>
<thead>
<tr>
<th>Situation or Activity</th>
<th>Communications Concepts Applied</th>
</tr>
</thead>
<tbody>
<tr>
<td>Various interim deliverables intended for either internal or external use</td>
<td>Examples</td>
</tr>
<tr>
<td></td>
<td>• Descriptive/Explanatory Composition: Describe behaviors, verbalize the functionality of parts of the program</td>
</tr>
<tr>
<td></td>
<td>• Expository writing: How the machine works</td>
</tr>
<tr>
<td></td>
<td>• Persuasive/Explanatory Composition: Justify a design choice</td>
</tr>
<tr>
<td></td>
<td>• Record data in a table, evaluation of methods, predictions, describing robot behavior, describing a proportional relationship</td>
</tr>
<tr>
<td></td>
<td>• Verbalize troubleshooting processes, analyzing and describing an unexpected situation or observation</td>
</tr>
<tr>
<td></td>
<td>• Describe a design concept</td>
</tr>
<tr>
<td></td>
<td>• Compare/contrast design choices, document and record steps, explain why the group took a certain approach</td>
</tr>
<tr>
<td></td>
<td>• Research, examine and evaluate real-world robot applications</td>
</tr>
<tr>
<td></td>
<td>• Describe a complex programming concept</td>
</tr>
<tr>
<td></td>
<td>• Develop a marketing plan for a robot technology</td>
</tr>
</tbody>
</table>
The Carnegie Mellon Robotics Academy develops...
Classroom-proven tools for teachers that foreground STEM concepts through hands-on minds-on robotics activities

**Robotics Engineering Vol. I**
*Introduction to Mobile Robotics* CD-Rom
A standards-mapped, STEM-based curriculum with engaging activities, extensive resources, and complete teacher support...
- 18 lessons supported by video tutorials
- Building instructions & programming assistance
- Explanatory animations for each sensor
- Worksheets and data tables for each lesson
- Teacher notes and implementation suggestions
- Handouts and introductory Powerpoint presentations
- Quizzes, answer keys and evaluation rubrics
- Aligns to Math, Science, and Technology standards

**Robotics Engineering Vol. II**
*Guided Research* CD-Rom / DVD
A learning continuum (with Vol. I) that features research, teamwork, and ‘real world’ engineering problem solving...
- Automated mining, patrol robot, and automated tree measurer projects where students learn advanced programming concepts
- 3 guided research and engineering challenges that build on the projects: mine mapping, creating a sentry system, & tree surveying
- Advanced programming videos on loops, switch blocks, data hubs, displaying real time data, storing variables, and Bluetooth
- Teacher notes & implementation suggestions, lesson handouts, evaluation rubrics, quiz answers. Aligns to Math, Science, Technology, and Communications standards

**NXT Video Trainer 2.0 DVD**
A self-paced guide for students & educators that teaches software programming for NXT-G through basics & beyond...
- Focused on introductory programming including motors, sensors, and decision-making
- Self-guided video lesson structure with regular ‘check your understanding’ questions
- Programming lessons paired with STEM investigations
- Classroom-ready with printable worksheets, teacher guide, and step-by-step video directions

**Student learning that contextualizes math, science, technology and engineering**

**ROBOTC Software**

Specifically designed to program educational robots. Includes a user-friendly interface with basic and advanced programming options. Based on industry-standard C code and compatible with multiple robot hardware platforms.

**ROBOTC Curriculum**

Teaching ROBOTC for LEGO MINDSTORMS
Curriculum includes over 45 short videos, over 300 pages of documentation, 20 programming challenges, and quizzes to check student understanding. Leads new programmers step-by-step into the world of industry-standard C-programming.

**Online Professional Development**
A live online course that teaches you how to program NXT-based robots and how to use robotics as an organizer to teach STEM...

Course includes:
- USB headset (yours to keep afterward)
- NXT Video Trainer CD with classroom license ($225 value)
- Online access to supplemental lessons
- Technical support
- Live access to instructors
- 24/7 access to class forums / message boards

Carnegie Mellon Robotics Academy

www.education.rec.ri.cmu.edu • 412 681.7160 • Ten 40th St., Pittsburgh, PA 15201
The Carnegie Mellon Robotics Academy develops classroom-proven tools for teachers that foreground STEM concepts through hands-on minds-on robotics activities.

**ROBOTC Curriculum**
For TETRIX and LEGO MINDSTORMS CD-Rom
A multimedia curriculum that leads new programmers step-by-step into the world of industry-standard "C" programming...
- Over 50 short videos help new users "out of the starting gate".
- Set-up section covers firmware, building your first bot, & more
- Students learn the role of a programmer, and what syntax is
- Lessons on autonomous control of a robot's speed & direction
- Challenges augment lessons with engaging scenarios
- Extensive coverage of sensor hardware and feedback
- Students learn how to use variables and create functions
- Aligns to Math, Science, and Technology standards

**Robotics Engineering Vol. I**
Home School CD-Rom
Single License for Home School use only
A standards-mapped, STEM-based curriculum with engaging activities, extensive resources, and complete teacher support...
- 18 lessons supported by video tutorials
- Building instructions & programming assistance
- Explanatory animations for each sensor
- Worksheets and data tables for each lesson
- Teacher notes and implementation suggestions
- Handouts and introductory Powerpoint presentations
- Quizzes, answer keys and evaluation rubrics

**Science & Datalogging Investigations & Experiments** CD-Rom
Introduces new programmers & scientists to the data logging capabilities of the NXT, including the temperature sensor...
- Six inquiry-focused STEM lessons and projects
- Teacher materials included for each lesson
- Over 40 short videos
- Accompanying worksheets, quizzes, and checks for understanding
- Building instructions for all programs
- Models designed to be built quickly by students in the classroom

**Deep Space Terraformers**
Terraformers Camp-on-a-Disk CD takes campers on a mission to make a distant planet habitable for humans. Videos and animations help campers build robots, learn programming, tackle challenges, and hone their robotic and project planning skills.
Terraformers is perfect for a one-week robotics camp or a 9-to-12 week after-school program. It provides extensive resources for a Robocamp director, including a step-by-step camp guide, registration forms, awards sheets, gameboard plans, themed props, and more.

**RCX**
Robocamps-on-a-Disk
A live online course that teaches you how to program NXT-based robots and how to use robotics as an organizer to teach STEM...
Course includes:
- USB headset (yours to keep afterward)
- NXT Video Trainer CD with classroom license ($225 value)
- Online access to supplemental lessons
- Technical support
- Live access to instructors
- 24/7 access to class forums / message boards

**Online Professional Development**
A live online course that teaches you how to program NXT-based robots and how to use robotics as an organizer to teach STEM...

www.education.rec.ri.cmu.edu • 412 681.7160 • Ten 40th St., Pittsburgh, PA 15201