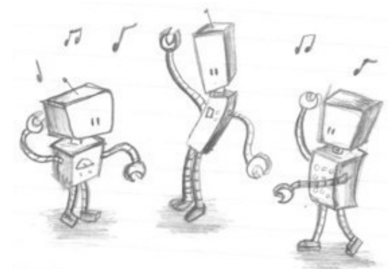
Teaching Secondary STEM Concepts With Robotics CEESA Teachers Conference Prague 2013
Dean Hester, American International School of Bucharest dhester@aisb.ro



Robotics Education Resources

European Robotics Competitions:

FIRST Lego League (FLL)
A competition for elementary and middle school students using the Mindstorms NXT robotics kit. A team of up to 10 students builds and programs a robot capable of completing multiple challenge linked to a theme. The 2012 theme was Senior Solutions in which students explored how technology could be used to assist the elderly. Teams are expected to complete a research project as part of the competition. The CEESA middle school robotics competition is an unofficial modified version of the FLL tournament.
(http://www.firstlegoleague.org)

FIRST Tech Challenge (FTC)
A competition for middle school and high school students using the Mindstorms NXT and Tetrix or Matrix robotics kits. A team of up to 10 students builds and programs a robot capable of competing in an alliance with another team's robot to score points in a game that changes every year. The 2013 game is entitled Ring It Up! and requires robots to score points by placing rings to a vertical "tic, tac, toe" rack. Team members are expected to submit an engineering log and eligible for U.S. college scholarships. The CEESA high school robotics competition is the official Eastern Europe FTC Championships.
(http://www.firstlegoleague.org) (http://www.easterneuropefirsttechchallenge.org)

VEX Robotics Competition
A competition for high school and college students using the Vex robotics kit. Teams build and program a robot capable of competing in an alliance with another team's robot to score points in a game that changes every year. The 2013 game is entitled Sack Attack and requires robot to score points by placing bean bags into goal troughs.
(http://www.vexrobotics.com/vex/competition/)

World Robotics Olympiad
A competition with levels for elementary, middle and high school students using the Mindstorms NXT robotics kit. In the regular category teams build and program a robot capable of completing multiple challenges. In the open category teams design robots based on a given theme. The 2013 theme is World Heritage. The 2014 WRO will be held in Moscow, Russia.
(http://www.wroboto.org)

Teaching Secondary STEM Concepts With Robotics CEESA Teachers Conference Prague 2013
Dean Hester, American International School of Bucharest dhester@aisb.ro
Sources for Robotics Parts:

<u>Lego Mindstorms Kit</u> Cost 478€ from Ro-Botica in Spain (www.ro-botica.com)
Advantages
Used by the *FIRST* Lego League, World Robotics Olympiad and CEESA middle school robotics competitions.
Used in many international schools.
A wide variety of parts for purchase in Europe.
Wide variety of books and websites with plans and programs.
Can be programmed in NXT-G, Labview, or Robot C.
Disadvantages
NXT-G software not included in the education kit.
3 Motor and 4 Sensor CPU Limitation
Structural limitations of plastic.

<u>Vex Dual Control Starter Kit</u> Cost 500€ from Vex Robotics in the UK (www.vexrobotics.com)
Advantages
Used by the VEX Robotics Competition.
A wide variety of parts for purchase in Europe.
Hardware only lessons are possible.
Two CPU options.
Max of 8 motors/servos and 16 sensors.
Can be programmed in Easy C or Robot C.
Disadvantages
Many of the parts are plastic.
Hole pattern design limits

<u>Tetrix Basic Education Kit</u> Cost 899€ from Ro-Botica in Spain (www.ro-botica.com)
Advantages
Used by FIRST Tech Challenge competition.
Metal parts are more durable.
Motors have more torque.
Versatile hole patterns.
Can be programmed in Labview or Robot C.
Max 4 motor and servo controllers.
Disadvantages
“Brain” comes from Mindstorms kit.
Motors do not include encoders.
Only 3 sensors without multiplexer.
Only the kits are sold in Europe.

<u>Matrix Base Robotics Kit</u> Cost 480€ from Generation Robots in France (www.generationrobots.com)
Advantages
Used by the *FIRST* Tech Challenge and World Robotics Olympiad competitions.
Prototyping connectors.
Can be programmed in NXT-G, Labview, or Robot C.
Disadvantages
“Brain” comes from Mindstorms kit.
Controller limit of 4 motors and 4 servos.
Only 3 sensors without multiplexer.
Only the kits are sold in Europe.

Teaching Secondary STEM Concepts With Robotics CEESA Teachers Conference Prague 2013
Dean Hester, American International School of Bucharest dhester@aisb.ro
Robotics Curriculum and Design Cycle Teaching Resources:

RoboMatter (www.robomatter.com)

A distributor of licenses for RobotC and curriculum for Mindstorms, Tetrix and Vex robotics for both middle and high school levels. RoboMatter also offers online courses in programming with RobotC and NXT-G. The distributor is associated with Carnegie Mellon University in the United States. An online version of the RobotC curriculum for Mindstorms and Tetrix can be found at http://www.education.rec.ri.cmu.edu/previews/robot_c_products/teaching_rc_tetrix_preview/

Computer Science Student Network (www.cs2n.org/teachers)

Free online lessons in animation, robotics, game and web design. The robotics lesson include activities based on using both NXT-G and RobotC programming languages. Teachers can track the progress of students using a learning management system with results that can be exported to a grade book.

Damien Kee (www.damienkee.com)

Author of Classroom Activities for the Busy Teacher and other Mindstorms NXT books. Lessons are written to use the Mindstorms NXT-G software.

NeboMusic (www.nebomusic.net/neborobotics.html)

Music and technology teacher that has created robotics lessons using Mindstorms NXT-G, RobotC and Python programming software.

Lucid Chart (www.lucidchart.com)

Website for creating flowcharts for the planning stage of the design cycle. Flowcharts can be created in collaboration with other students. Flowcharts can be downloaded as a pdf, png or jpeg file.

Tom's Planner (www.tomsplanner.com)

Online Gantt Chart creator for the planning stage of the design cycle. With a teacher's unlimited account students can collaborate on creating charts and download charts using free accounts.

Lego Digital Designer (http://ldd.lego.com)

Free CAD software for Windows or Mac computers for the design stage of the design cycle. Mindstorms robot designs and building instructions can be printed out or saved as a pdf file.

PTC Creo (http://www.ptc.com/company/community/download/)

Free CAD software for Windows computers for the design stage of the design cycle. Tetrix robot designs can be exported as a jpeg file. Animations can be created within the software from robot designs.

Mathematics applied when experimenting with Robots
Robotics provides a hands-on-mind-on set of lessons that reinforce and teach STEM concepts in an engaging and concrete manner. In table 1 below you will find an applied robotic application that is measurable on the left and the mathematic concept that can be taught on the right.

Table 1

Application	Mathematics Concept Demonstrated
Calculating distance traveled based on the number of rotations of an axle and the diameter of a wheel attached to the axle	Circle geometry (Diameter, Circumference) **C = π*d** Measurement Distance – centimeters Unit conversion – between systems (centimeters to inches), within systems (centimeters to meters), etc. 1 rotation = 360 degrees = X cm = Y in. (values of X and Y are calculated based on wheel size) Operations Define a procedure to convert cm into rotations
Calculating speed and acceleration using a variety of different methods (stopwatch, meterstick, software control, datalogging and graphs)	Speed equation (d=rt), variables and constants, graphs
Analyzing data collected on trial runs while conducting experiments	Central tendency (mean of 3 data values (a+b+c)/3 as a representative value) Goodness of fit – Percent error calculation: $\dfrac{\lvert theoretical\ measurement - actual\ measurement \rvert}{theoretical\ measurement}$
Predicting, based on discovered or proposed linear/proportional relationships, the distance or speed the robot will travel when changing wheel sizes/RPM of the robot/or the number of rotations of the axle	Ratios and proportions ($\frac{a}{b} = \frac{2a}{2b}$) Direct linear relationship y = ax Inverse linear relationship y = a/x Graphing, interpolation, extrapolation
Basic programming – conditional statements, calculating thresholds, passing equations in experiments	**Boolean logic**, Thresholds, Averages (means), Comparisons **(<, >, <=, >=, ==, !=)**, **Operations (+, -, ×, ÷, |x|)**
Path Planning/robot navigation– given several locations and variables write a program using the appropriate mathematical formulas to move from point to point.	Applied pre-algebra, algebra, geometry, trigonometry
Written explanation of the mathematics the student just used to solve the problem	Communication – Identify and communicate relevant math concepts used, including formulas, calculations, data, and results
Robot Mapping Scanning and mapping using the ultrasonic sensor	**Angles, Graphing, Measurement** **Polar coordinates** (very basic)**, Plotting in polar**
Robotic Calipers Calculate the size and mass of a tree based on a cylindrical model, and automated measurements.	Measurement, Mass, Volume, Density, Geometry of solids, Circle/cylinder density, Gear ratios, Physical modeling, Approximations

Table 1 continued

Application	Mathematics Concept Demonstrated
Gather data in a table, plot data on a graph, then calculate a linear regression using Excel; use the graph and equation to take future measurements	Tables, graphs, interpolation, extrapolation, linear regression, computational tools, calibration
Counting lines that the robot runs across	Variables, operations, and formulas
Predicting rates of occurrence of combinations of events on spinner wheels	Probability, compound probability

Science applied when experimenting with robots

In table 2 below Carnegie Mellon's Robotics Academy has developed a set of robotic investigations that engage students in scientific analysis where students apply scientific process investigate/research scientific concepts.

Table 2

Application	Science Concept Demonstrated
Investigation Identifying a question, formulating a possible explanation (with scaffolded guidance), collecting data to test the hypothesis, analyzing the data, drawing conclusions.	Experimental Design • hypothesis • independent variables • dependent variables • control variables • experimental conditions • multiple trials • evidence Numeric Analysis • error • %error Estimates and approximation Collecting and analyzing data collected on trial runs while conducting experiments
Examine the behavior of the Sound Sensor	• Sound and waves • Amplitude vs. frequency • Sounds vs. human perception dB/dBA via • Decibels • Data acquisition • Analysis
Programming and Modeling Light sensor Line-Tracking Behaviors	Light and reflectivity Observation and Predictions Perception of light/dark Modeling (of robot behavior, extended and refined several times) Color and perception (human vs. light sensor, reflectivity of different wavelengths)
Defining the performance envelope of the Ultrasound sensor	Measurement (distance) Ultrasound and sound waves Plotting data **Outliers error** Spatial Graph Model (Ultrasonic detection area)

Table 2 continued

Application	Science Concept Demonstrated
Designing robots - Form follows function	Mass, Speed, Equilibrium (forces in equilibrium = top speed, stall point), Measurement (mass, forces, speed), Center of Mass, Support Polygon, Balance, balanced and unbalanced forces
Analyzing robot's drive train – gears/gear ratios/compound gear ratios	**Torque, Gears/Gear Ratios/Mechanical Advantage/Simple Machines, Idler Gears**
Discover and describe how touch sensors work	**Basic circuits** Direct Current Load Conductors Switch Power Supply
Testing and describing how motors and generators work	**Magnetism** **Electromagnet** **Electromagnetic induction**
Gathering user testing data about the success of a robot's ability to communicate with people	Social Science methods (surveys, observational analysis)
Remote Control	Resonance Crystals Frequency
Engineering Design Inquiry – Design-motivated project based learning	Inquiry processes in practice, Science and Technology
Written explanation of results, design decisions, experimental results, and scientific process	Communication – Communicate relevant science process and content concepts visited during the investigations
Measuring the cooling rates of water in different containers	Thermodynamics, specific heat, thermal conductivity
Comparing the effectiveness of different cooling methods on water	Convection cooling, experimental process
Estimating/determining the concentration of insoluble material in water by measuring opacity (turbidity)	Light, refraction, solutions, solubility, water quality
Monitoring the temperature of air, soil, and water during a day-night cycle using data logging	Earth science, heat, temperature, energy, specific heat
Measuring the free-falling speeds of different objects	Gravity, air resistance, aerodynamics
Underwater-themed challenges in Aquabots Camp-on-a-Disk product	Oceanography
Nature-themed challenges in Dataloggers Camp-on-a-Disk product	Ecology/Environmental Studies

Technology Education

Robotics serves as an excellent integrator for teaching technological literacy as well as reinforcing mathematical and scientific competency. Table 3 shows fundamental lessons that can be implemented in a well designed robotic class.

Table 3

Application/Lesson	Technological Concept
Safety	Human factors • Creating a safe environment • Safety as an attitude • Legal implications • Using tools and machinery • Designing with safety in mind
Technology and Society	Robot impact on society Types of robot interactions - Communication Technology (robot-human and human-human via robot) Expected and unexpected consequences of robotic development Impact on jobs (jobs created – jobs replaced)
Career choices	Career preparation Administration Sales Marketing Engineering Programming Technician Entrepreneur
Understanding robotic systems	The concept of systems – Motors (output), Controller, Sensors (input, feedback), Program, Mechanical elements
Robot design	Engineering process model • Input-Process-Output-Feedback • Form follows function • Design tradeoffs (Drive train design, sensor integration, motor choices, remote vs. autonomous) • Testing & Revision
Controlling robotic systems	**Control** • Behaviors • Timers/Wait states • Conditional statements • Loops • Remote control • Bluetooth
Team problem solving	**Teamwork skills** • Cooperative learning • Differentiated roles and specialization
Managing a project	Project management • Time management/scheduleing • Resource allocation/budgeting • Systems analysis • Information accessing

Table 3 continued

Application/Lesson	Technological Concept
Sensors • How they work - Math and scientific principles behind sensor design • Type of feedback to expect • Designing sensor • Application	Sensor types and applications • Touch • Light • Encoder • Ultrasound • Infrared • Sound
Communications • Written • Digital multimedia • Presentation	Summative – Evaluate and present the capabilities and appropriateness of the technology to the problem presented

Communications

There are many rich opportunities to reinforce critical thinking, communication, and writing skills in robotics. Table 4 below lists several opportunities.

Table 4

Activity	Communications
Engineering Design Notebook	most questions require written explanation in addition to calculations or programming, general teamwork skills, build technical vocabulary, create logical notes and summaries
Written description of technological/scientific/ or mathematic concept	• Descriptive/Explanatory Composition: Describing behaviors, verbalize the functionality of parts of the program • Describe patterns of turning motion, compare and contrast two different types of turns • Recording data in a table, evaluation of methods, predictions, describing robot behavior, describing a proportional relationship • Verbalize troubleshooting processes, analyzing and describing an unexpected situation or observation • Make predictions using a model, extend and revise the model • Compare/contrast design choices, documenting and recording steps, explanation of why this was the best choice • Describe the effects of gears • Describe a design concept, explain the significance of center of mass to the design • Expository writing: How the machine works • Research, examine and evaluate real-world robot applications, predict the future, justify your predication • Describe a complex programming concept (multitasking) • Develop a marketing plan for a robot technology • Persuasive/Explanatory Composition: Justify a design choice
Technical report	Recording data, organizing data in tables, explaining scientific reasoning, arguing support/dissent for a hypothesis using data, extracting information from a written scenario, explaining the importance of communication within a team

ROBOTC Natural Language - NXT Quick Reference:

Robot Type Choose which robot you are using. *Default bot: none.*	`robotType();`	`robotType(rembot);`

Start Motor
Set a specific motor to a speed.
Default motor and speed: motorA, 75.
	`startMotor();`
	`wait();`
	`stopMotor();`
		`startMotor(motorC, -25);`
		`wait(0.5);`
		`stopMotor(motorC);`

Stop Motor
Stop a specific motor.
Default motor: motorA.
	`startMotor();`
	`wait();`
	`stopMotor();`
		`startMotor(motorC, -25);`
		`wait(0.5);`
		`stopMotor(motorC);`

Wait
Wait an amount of time measured in seconds.
Default time: 1.0.
	`startMotor();`
	`wait();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`wait(2.7);`
		`stopMotor(motorC);`

Wait in Milliseconds
Wait an amount of time measured in milliseconds.
Default time: 1000.
	`startMotor();`
	`waitInMilliseconds();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`waitInMilliseconds(2700);`
		`stopMotor(motorC);`

Until Touch
The robot waits for the Touch Sensor to be pressed.
Default sensor port: S1.
	`startMotor();`
	`untilTouch();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`untilTouch(S4);`
		`stopMotor(motorC);`

Until Release
The robot waits for the Touch Sensor to be released.
Default sensor port: S1.
	`startMotor();`
	`untilRelease();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`untilRelease(S4);`
		`stopMotor(motorC);`

Until Bump
The robot waits for the Touch Sensor to be pressed in and then released out.
Default sensor port and delay time: S1, 10.
	`startMotor();`
	`untilBump();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`untilBump(S4, 100);`
		`stopMotor(motorC);`

Until Button Press
The robot waits for a button on the NXT to be pressed.
Default button: centerBtnNXT.
	`startMotor();`
	`untilButtonPress();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`untilButtonPress(rightBtnNXT);`
		`stopMotor(motorC);`

Until Sonar - Less Than
The robot waits for the Sonar Sensor to read a value in cm less than the threshold.
Default threshold and sensor port: 30, S4.
	`startMotor();`
	`untilSonarLessThan();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`untilSonarLessThan(45, S1);`
		`stopMotor(motorC);`

Until Sonar - Greater Than
The robot waits for the Sonar Sensor to read a value in cm greater than the threshold.
Default threshold and sensor port: 30, S4.
	`startMotor();`
	`untilSonarGreaterThan();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`untilSonarGreaterThan(45, S1);`
		`stopMotor(motorC);`

ROBOTC Natural Language - NXT Quick Reference:

Until Dark
The robot waits for the Light Sensor to read a value less than the threshold.
Default threshold and sensor port: 45, S3.
	`startMotor();`
	`untilDark();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`untilDark(15, S2);`
		`stopMotor(motorC);`

Until Light
The robot waits for the Light Sensor to read a value greater than the threshold.
Default threshold and sensor port: 45, in2.
	`startMotor();`
	`untilLight();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`untilLight(85, S2);`
		`stopMotor(motorC);`

Until Sound - Less Than
The robot waits for the Sound Sensor to read a value less than the threshold.
Default threshold and sensor port: 50, S2.
	`startMotor();`
	`untilSoundLessThan();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`untilSoundLessThan(15, S3);`
		`stopMotor(motorC);`

Until Sound - Greater Than
The robot waits for the Sound Sensor to read a value greater than the threshold.
Default threshold and sensor port: 50, S2.
	`startMotor();`
	`untilSoundGreaterThan();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`untilSoundGreaterThan(85, S3);`
		`stopMotor(motorC);`

Until Rotations
The robot waits for a motor-encoder to reach a specified number of rotations.
Default rotations, encoder: 1.0, motorB
	`startMotor();`
	`untilRotations();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`untilRotations(2.75, motorA);`
		`stopMotor(motorC);`

Until Encoder Counts
The robot waits for a motor-encoder to reach a specified number of encoder counts.
Default counts, encoder: 360, motorB.
	`startMotor();`
	`untilEncoderCounts();`
	`stopMotor();`
		`startMotor(motorC, 50);`
		`untilEncoderCounts(990, motorA);`
		`stopMotor(motorC);`

Forward
The robot drives straight forward.
Default speed: 75.
	`forward();`
	`wait();`
	`stop();`
		`forward(50);`
		`wait(2.0);`
		`stop();`

Backward
The robot drives straight backward.
Default speed: -75.
	`backward();`
	`wait();`
	`stop();`
		`backward(50);`
		`wait(2.0);`
		`stop();`

Point Turn
The robot makes a sharp turn in place.
Default direction and speed: right, 75.
	`pointTurn();`
	`wait();`
	`stop();`
		`pointTurn(left, 50);`
		`wait(0.4);`
		`stop();`

Swing Turn
The robot makes a wide turn, activating only one drive motor.
Default direction and speed: right, 75.
	`swingTurn();`
	`wait();`
	`stop();`
		`swingTurn(left, 50);`
		`wait(0.75);`
		`stop();`

Stop
The robot halts both driving motors, coming to a stop.
	`forward();`
	`wait();`
	`stop();`
		`forward(50);`
		`wait(2.0);`
		`stop();`

ROBOTC Natural Language - NXT Quick Reference:

Line Track - for Time The robot tracks a dark line on a light surface for a specified time in seconds. *Default time, threshold, sensors:* 5.0, 45, S3.	`lineTrackForTime();` `stop();`	`lineTrackForTime(7.5, 75, S2);` `stop();`
Line Track - for Rotations The robot tracks a dark line on a light surface for a specified distance in rotations. *Default time, threshold, sensors:* 3.0, 45, S3.	`lineTrackForRotations();` `stop();`	`lineTrackForRotations(4.75, 75, S3);` `stop();`
Move Straight - for Time The robot will use encoders to maintain a straight path for a specified time in seconds. *Default time, rightEncoder, leftEncoder:* 5.0, motorB, motorC.	`moveStraightForTime();` `stop();`	`moveStraightForRotations(4.75, motorC, motorA);` `stop();`
Move Straight - for Rotations The robot will use encoders to maintain a straight path for a specified distance in encoder rotations. *Default rotations, rightEncoder, leftEncoder:* 1.0, motorB, motorC.	`moveStraightForRotations();` `stop();`	`moveStraightForRotations(4.75, motorC, motorA);` `stop();`

Tank Control
The robot is remote controlled with the right motor mapped to the right joystick and the left motor mapped to the left joystick.
Default right and left and joystick: joy1_y2, joy1_y1.
	`while(true)`
	`{`
	`tankControl();`
	`}`
		`while(true)`
		`{`
		`tankControl(joystick.joy1_x2, joystick.joy1_x1);`
		`}`

Arcade Control
The robot is remote controlled with both motors mapped to a single joystick.
Default vertical and horizontal joysticks: joy1_y2, joy1_y1.
	`while(true)`
	`{`
	`arcadeControl();`
	`}`
		`while(true)`
		`{`
		`arcadeControl(joystick.joy1_y1, joystick.joy1_x1);`
		`}`

Robot C Dance Lessons

The "Basic Forward and Back Shuffle" Dance Move

Step 1. Run the RotateForwardBack.c using the UNIQ robot on a sheet of paper and mark how far it goes forward and how far it goes back.

Step 2. Run the WaitForwardBack.c using the TANZEN robot on a sheet of paper and mark how far it goes forward and how far it goes back.

Step 3. Record the ratio of the distance the UNIQ robot drove forward divided by the distance the TANZEN robot drove forward. Record the ratio of the distance the UNIQ robot drove backward divided by the distance the TANZEN robot drove backward. What do you notice about the 2 ratios that you recorded?

$\frac{UNIQ\ Forward\ Distance}{TANZEN\ Forward\ Distance} =$ ______________ $\frac{UNIQ\ Backward\ Distance}{TANZEN\ Backward\ Distance} =$ ______________

Step 4. Use the ratios you wrote in step 3 to modify the wait times for both the forward and backward movements. Describe how you modified the wait time so the TANZEN robot travels the same distance below.

The "Improved Forward and Back Shuffle" Dance Move

The Basic Forward and Back Shuffle Move requires the TANZEN to always have the same battery charge in order to travel the same distance every time it is run but having constant battery power is not very realistic over days of performance! In the Improved Forward and Back Shuffle the robot will use it's motor encoders to ensure it moves the same distance regardless of how much battery power it has.

Step 1. Look at the RotateForwardBack.c program the UNIQ robot was using for its dance move. Why was the RobotC wait command replaced by the untilRotations command?

Step 2. Download the RotateForwardBack.c to the TANZEN robot and run the program. Did the TANZEN robot travel the same distance as the UNIQ robot while using the same program? How are the UNIQ robot wheels and TANZEN robot wheels different?

Step 3. Use the circumference formula C = π x diameter to find the circumference of a TANZEN wheel. Show your calculations below.

TANZEN Wheel Circumference = ________________________

Step 4. Use the distance the UNIQ robot travels forward and the circumference of the TANZEN wheel to calculate the number of times the TANZEN wheel must rotate.

Step 5. Modify the RotateForwardBack.c program with the wheel rotations you calculated in step 4. Run the modified program. Does the TANZEN's Forward and Back Shuffle Move match the UNIQ's dance move? If it does your TANZEN robot has mastered the Improved Forward and Back Shuffle!

The “Basic Right Swing Turn” Dance Move

Step 1. Attached the angle measurement tool to the TANZEN robot.

Step 2. Run the RotateSwingTurn.c program using the UNIQ robot. It will make a 90° swing turn by driving only the left wheel forward.

Step 3. Run the WaitSwingTurn.c program using the TANZEN robot. Did it turn 90° ?

Step 4. Use the angle measurement tool to mark the starting position of the TANZEN robot. Run the WaitSwingTurn.c program again and mark the ending position of the TANZEN robot using the angle measurement tool. What is the measurement of the angle between the starting and ending positions in degrees?

Step 5. Record the ratio of 90° divided by the angle measured in step 4 below.

Step 6. Describe how to use the ratio you wrote in step 5 to modify the wait time for the WaitSwingTurn.c program on the TANZEN robot so that it turns 90°.

The “Improved Right Swing Turn” Dance Move

Step 1. Download the RotateSwingTurn.c program the UNIQ robot used to make a 90° swing turn to the TANZEN robot and run the program. Why does the TANZEN robot not make a 90° turn? [Hint: How is the TANZEN’s wheel drive different than the UNIQ’s wheel drive?]

Step 2. Measure the distance between the wheels of the TANZEN robot and record it below. This distance is called the wheel base.

Wheel Base Distance = ______________________________

Step 3. The wheel base distance is the same as the radius of the turning circle for the TANZEN robot when it makes a swing turn. Calculate a quarter of the circumference of the TANZEN turning circle below using one fourth of the result from C = 2π x radius. Why are we calculating only one fourth of the circumference?

Step 4. The number of rotations the left wheel must turn is equal to the distance you calculated in step 2 divided by the circumference of the wheel. Use the distance found in step 2 and the wheel circumference you calculated in step 3 of the Improved Forward and Back Shuffle Move instructions to find the number of rotations needed to make a 90° swing turn.

Step 5. Modify the RotateSwingTurn.c program with the wheel rotations you calculated in step 4. Run the modified program. Does the TANZEN’s right swing turn match the UNIQ’s 90° swing turn? If it does not turn 90° make a small modification to the number of rotations so the TANZEN robot turns 90°.

The "Point Swing Turn" Dance Move

In point swing turn one wheel turns in the opposite direction of the other wheel to spin the robot around its center.

Step 1. Why is the radius for the turning circle of a point turn smaller than the radius for the turning circle of a point turn? Write the new radius for the turning circle for a TANZEN point turn below.

Step 2. Calculate a quarter of the circumference of the TANZEN turning circle below using one fourth of the result from C = 2π x radius using the new turning circle radius from step 1.

Step 3. The number of rotations the left wheel must turn is equal to the distance you calculated in step 2 divided by the circumference of the wheel. Use the distance found in step 2 and the wheel circumference you calculated in step 3 of the Improved Forward and Back Shuffle Move instructions to find the number of rotations needed to make a 90° point turn.

Step 4. Save the RotateSwingTurn.c program with a new program file name RotatePointTurn.c. Replace the swingTurn(right, 25) command in the program with a pointTurn(right,25) command. Change the number of rotations in the untilRotations command to match the number of rotations you calculated in Step 3.

Step 5. Download and run the new RotatePointTurn.c program. Use the angle measurement tool to measure how many degrees the TANZEN robot turns. Does the TANZEN make a 90° point turn? If it does not turn 90° make a small modification to the number of rotations so the TANZEN robot turns 90°.

TANZEN Robot Square Dance

Step 1. Save the RotatePointTurn.c program with a new program file name SquareDance.c. Copy the pointTurn and untilRotations command three time and insert a wait() command after each untilRotations command. Download and run the new SquareDance.c program. Does your robot turn 360°?

Step 2. Insert a forward(25) command and untilRotations(10) after each wait command. Download and run the modified program. Is the path your robot drives a square? If not modify the untilRotations command after the pointTurn command so the robot drive path is as close to a square as possible.

[optional] Use a int turncount; , turncount = 0; , turncount = turncount +1; and while(turncount<4){} control structure to shorten the program by looping around one set of pointTurn, untilRotations, wait(),forward, and second untilRotations commands four times. Done correctly the square dance can be complete using a program that is less than 11 lines!

Boom, Boom, Pow Synchronized Dance

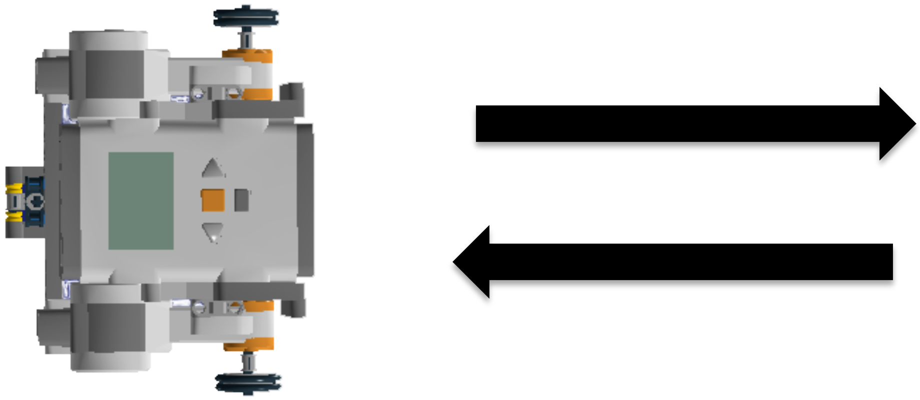
Step 1. Run the BoomBoom.c program on the UNIQ robot and watch it dance! Download the same program to the TANZEN robot and run it? What is the purpose of untilSoundGreaterThan() command in the program? Why does the TANZEN's dance moves not look like the UNIQ's moves?

Step 2. Use what you have learned from the steps for the previous TANZEN robot dance moves to modify the BoomBoom.c program so the TANZEN robot can dance in synch with the UNIQ robot.

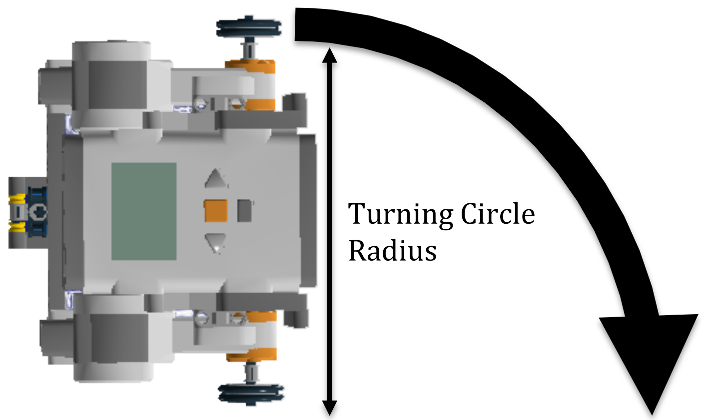
Step 3. Now that the two robots are dancing in synch try adding some more dance moves to the Boom, Boom, Pow dance routine to extend the synchronized robot dance performance! Try making up your own new dance moves! Describe any new dance moves using RobotC commands below.

TANZEN Dance Moves

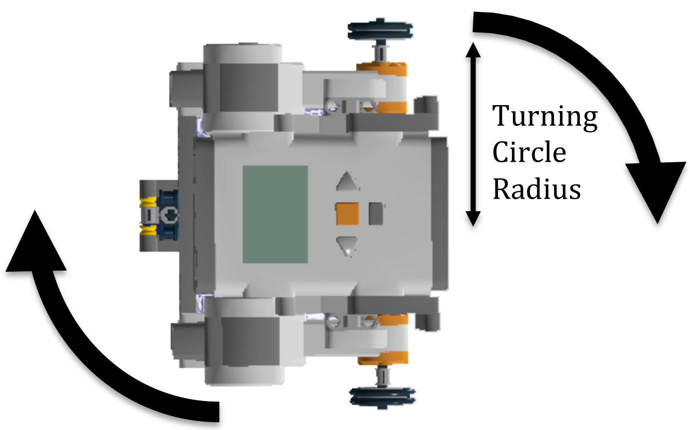
Forward and Back Shuffle

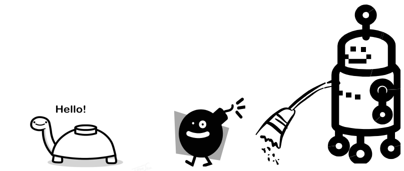


Right Swing Turn



Right Point Turn





Mind the Mines! High School Design Cycle Project Description
Due Thursday, December 13, 2012

In this project you will build and program a robotic minesweeper that will collect 12 racquetball "mines" from the tops of PVC tube "detonators" and dispose of them safely in a plastic "blast bin". An example of one possible solution is shown in the picture below;

Your mine sweeper robot will be given 3 attempts to clear as many mines as possible in 2 minutes. The success of you robot will be scored by the following rules;

Each mine must be removed from its "trigger" pedestal and transported to a designated blast bin on the board, where it can be unloaded. The mines may also be "disarmed" for partial points by simply knocking them off their pedestals. Knocking over the pedestal itself, however, will set off the mine, and incur a penalty.

- Mines disarmed but not disposed of: +2 points each
- Mines properly disposed of: +20 points each
- Pedestals knocked over: - 7 points each
- Any physical contact between a human and the robot once the round has started will result in a 10 point **"touching penalty"** being assessed against the human's team. This includes touching the robot to turn it on after the starting signal. A sound should trigger the robot to turn on.
- Mines must be unloaded automatically by the robot into the blast bin by the robot detecting the walls of the blast bin or the electrical tape surrounding the blast bin area.
- Robots must be controlled using a Bluetooth remote control connection between a laptop running RobotC and a Mindtsorms NXT CPU.
- Robots must emit a startup sound after turning on and a mine disarming sound and mine disposal sound for each mine disarmed and disposed of. Robots that do not emit the required sounds will be disqualified from making any further attempts.
- Robots must deactivate automatically 2 minutes after they have been turned on. Robots that do not deactivate after 2 minutes (putting down the gamepad is not sufficient, the robot must actually stop accepting commands) will receive a total of 0 points for the attempt.

Mind the Mines! High School Design Cycle Project

Digital Design Folder Grading Criteria *Due Thursday, December 13, 2012*

Criterion A: Investigate
Maximum 6

Investigation is an essential stage in the design cycle. Students are expected to identify the problem, develop a design brief and formulate a design specification. Students are expected to acknowledge the sources of information and document these appropriately.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student **states** the task that the Minesweeper Robot must complete. The student investigates how the task could be met by **collecting information** from a variety of sources. The student lists some specifications.
3-4	The student **describes** the task that the Minesweeper Robot must complete and the task's relevance to clearing minefields in different countries. The student investigates how the task could be met by **collecting** information about possible Minesweeper Robot or Bomb Disposal Robot designs from a variety of **documented** sources. The student lists all specifications for the task and has **described** a test **in detail to evaluate** the robot against at least one specification.
5-6	The student **describes** the task that the Minesweeper Robot must complete and the task's relevance to clearing minefields in different countries. The student investigates how the task could be met by **collecting** information about possible Minesweeper Robot or Bomb Disposal Robot designs from a variety of **broad range of documented** sources. The student lists all specifications for the task and **describes multiple** tests **in detail to evaluate** the robot against more than one design specification.

Design brief: A simple explanation of a robot that could complete the Minesweeper challenge task.

Design specifications: A detailed description of the conditions, requirements and restrictions with which the Minesweeper Robot must comply. This is a precise and accurate list of facts such as conditions, dimensions, hardware and software that are important for the designer and for the user. All appropriate solutions will need to comply with the design specifications.

Criterion B: Design
Maximum 6

Students are expected to generate several feasible designs that meet the design specifications and to evaluate these against the design specifications. Students are then expected to select one design, justify their choice and evaluate this in detail against the design specifications.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student generates **one** Minesweeper Robot design documented **with photos** in their Engineering Logbook, and makes **some attempt to justify** this against the design specifications.
3-4	The student generates **two** Minesweeper Robot designs documented **with photos** in their Engineering Logbook, **justifying** the choice of **one** design and fully **evaluating** this design against the design specifications.
5-6	The student generates **two** Minesweeper Robot designs documented **with photos** in their Engineering Logbook, with **each design evaluated** against the design specifications and **justify** their final design choice.

Criterion C: Plan
Maximum 6
Students are expected to construct a plan to create their chosen product/solution that has a series of logical steps, and that makes effective use of resources and time.
Students are expected to evaluate the plan and justify any modifications to the design.

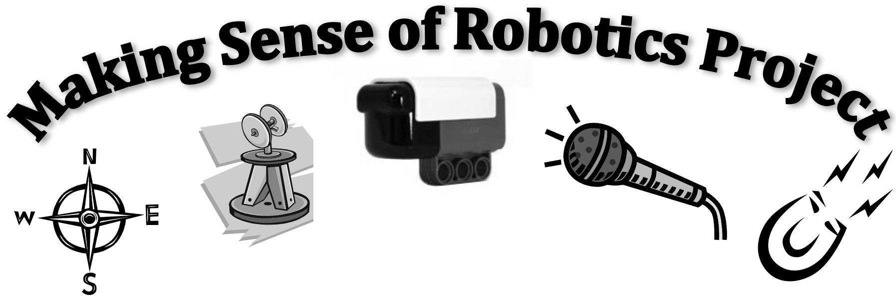
Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student produces a plan that includes a **flowchart** for the program used.
3-4	The student produces a plan that includes a **flowchart** for the program used and contains a **project timeline**. The student makes some attempt to evaluate the plan.
5-6	The student produces a plan that includes a **flowchart** for the program used and contains a **project timeline** for **all stages of the design cycle**. The student critically evaluates the plan and justifies any modifications to the design.

Criterion D: Create
Maximum 6
Students are expected to document, with a series of photographs or a video and a dated record, the process of making their product/solution, including when and how they use tools, materials and techniques. Students are expected to follow their plan, to evaluate the plan and to justify any changes they make to the plan while they are creating the product/solution.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student considers the plan and creates a Minesweeper Robot that can disarm and dispose of a mine without setting it off within 2 minutes.
3-4	The student **uses** appropriate techniques and equipment. The student follows the plan and **mentions** any modifications made, resulting in a Minesweeper Robot that can **disarm <u>or</u> dispose** either 4 mines (Critera D score: 3) or 6 mines (Critera D score: 4) in 2 minutes.
5-6	The student **competently uses** appropriate techniques and equipment. The student follows the plan and **justifies** any modifications made, resulting in a Minesweeper Robot that **can disarm <u>and</u> dispose** of either 4 mines (Critera D score: 5) or 6 mines (Critera D score: 6) in 2 minutes.

Criterion E: Evaluate
Maximum 6
Students are expected to evaluate the product/solution against the design specification in an objective manner based on testing, and to evaluate its impact on life, society and/or the environment. They are expected to explain how the product/solution could be improved as a result of these evaluations.
Students are expected to evaluate their own performance at each stage of the design cycle and to suggest ways in which their performance could be improved.

Level of Achievement	Descriptor
0	The student has not reached a satisfactory standard in any aspect listed above.
1-2	The student evaluates the Minesweeper Robot **or** his or her own performance. The student makes some **attempt to test** the Minesweeper Robot against the design specifications.
3-4	The student evaluates the Minesweeper Robot **and** his or her own performance and suggests ways in which these could be improved. The student **tests** the Minesweeper Robot to evaluate it against the design specifications.
5-6	The student evaluates the success of the Minesweeper Robot in an objective manner based on the **results of testing**, and the **possible views of the intended users of a full scale Minesweeping Robot**. The student provides an evaluation of his or her own performance **at each stage of the design cycle** and suggests improvements. The student provides an appropriate evaluation of the **impact** of the robotic minesweepers on life and society.



In this project you will be exploring how to design a robot using the advanced sensors built by HiTechnic. Each team will choose a HiTechnic sensor from the list below and investigate how it works and how to write RobotC code so a robot can use the sensor values. The robot must display sensor values and use those values to control motors or servos. The robot can be constructed from either Lego parts or a combination of Lego and Tetrix parts.

The design specifications for this project must include;

1. The use of a sound sensor to activate the robot by voice or a hand clap
2. The use of one of the HiTechnic sensors;

   NXT Gyro Sensor: The NXT Gyro Sensor returns the number of degrees per second of rotation as well as indicating the direction of rotation.

   NXT Compass Sensor: The NXT Compass Sensor is a digital compass that measures the earth's magnetic field and outputs a value representing the current heading.

   NXT Magnetic Sensor: The NXT Magnetic Sensor will enable you to build robots that can detect magnetic fields.

   NXT Acceleration Sensor: The HiTechnic Accelerometer / Tilt Sensor measures acceleration in three axes.

   NXT IRSeeker: The NXT IRSeeker is a multi-element infrared detector that detects infrared signals from sources such as the HiTechnic IRBall soccer ball, infrared remote controls and sunlight.

3. At least one motor or servo that is controlled using the sensor values.
4. Sensor values must be displayed on the NXT's screen.
5. The robot must respond to sensor changes for exactly 4 minutes after the sound sensor activation and then display the message "I am done"

As in your previous project you will be graded based on your digital design folder, which must include sections for the investigate, plan, design, create and evaluate stages of your project. Your digital design folder must have a description of how the sensor hardware and RobotC driver works. The evaluate section must also contain a description of a real world application for the same type of sensor. **The project deadline is Friday, January 27th at 3:00 p.m. with no extensions!**

Making Sense of Robotics
Digital Design Folder Grading Criteria *Due Friday January 27, 2011*

Criterion A: Investigate
Maximum 6

Investigation is an essential stage in the design cycle. Students are expected to identify the problem, develop a design brief and formulate a design specification. Students are expected to acknowledge the sources of information and document these appropriately.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The design specifications include all of the design brief tasks. The student investigates how the chosen HiTechnic sensor operates.
3-4	The design specifications include all of the design brief tasks and additional tasks using the sensor. The student investigates how the HiTechnic sensor hardware works and how to write RobotC programs that use the sensor. The student **describes** at least one test to **evaluate how the robot uses the HiTechnic sensor**.
5-6	The design specifications include all of the design brief tasks and additional tasks using the sensor. The student investigates how the HiTechnic sensor hardware works and how to write RobotC programs that use the sensor. The student **describes multiple tests to evaluate how the robot uses the HiTechnic sensor**.

Design specifications: A detailed description of the conditions, requirements and restrictions with which the robot must comply. This is a precise and accurate list of facts such as conditions, dimensions, hardware and software that are important for the designer and for the user. All appropriate solutions will need to comply with the design specifications.

Criterion B: Design
Maximum 6

Students are expected to generate several feasible designs that meet the design specifications and to evaluate these against the design specifications. Students are then expected to select one design, justify their choice and evaluate this in detail against the design specifications.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student generates **one robot design** documented **with photos** in their Engineering Logbook, and makes **some attempt to justify** this against the design specifications.
3-4	The student generates **two robot designs** documented **with photos** in their Engineering Logbook, **justifying** the choice of **one** design and fully **evaluating** this design against the design specifications.
5-6	The student generates **two robot designs** documented **with photos** in their Engineering Logbook, **each evaluated** against the design specifications and **justify** their final design choice.

Criterion C: Plan
Maximum 6
Students are expected to construct a plan to create their chosen product/solution that has a series of logical steps, and that makes effective use of resources and time.
Students are expected to evaluate the plan and justify any modifications to the design.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student produces a plan that includes a **flowchart** for programs used.
3-4	The student produces a plan that includes a **flowchart** for all programs used and contains a **prototype timeline**. The student makes some attempt to evaluate the plan.
5-6	The student produces a plan that includes a **flowchart** for all programs used and contains a **prototype timeline** for the **entire design cycle**. The student critically evaluates the plan and justifies any modifications to the design.

Criterion D: Create
Maximum 6
Students are expected to document, with a series of photographs or a video and a dated record, the process of making their product/solution, including when and how they use tools, materials and techniques. Students are expected to follow their plan, to evaluate the plan and to justify any changes they make to the plan while they are creating the product/solution.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student considers the plan and **creates a robot that activates by sound** and completes **one other design brief task either partially or completely.**
3-4	The student **uses** appropriate techniques and equipment. The student follows the plan and **mentions** any modifications made, resulting in a **robot that activates by sound** and completes **one design brief task completely and another design brief task either partially or completely.**
5-6	The student **competently uses** appropriate techniques and equipment. The student follows the plan and **justifies** any modifications made, resulting in a robot **that activates by sound** and completes **two design brief tasks completely and another design brief task either partially or completely.**

Criterion E: Evaluate
Maximum 6
Students are expected to evaluate the product/solution against the design specification in an objective manner based on testing, and to evaluate its impact on life, society and/or the environment. They are expected to explain how the product/solution could be improved as a result of these evaluations.
Students are expected to evaluate their own performance at each stage of the design cycle and to suggest ways in which their performance could be improved.

Level of Achievement	Descriptor
0	The student has not reached a satisfactory standard in any aspect listed above.
1-2	The student evaluates the robot **or** his or her own performance. The student makes some **attempt to test** the robot against the design specifications.
3-4	The student evaluates the robot **and** his or her own performance and suggests ways in which these could be improved. The student **tests** the robot to evaluate it against the design specifications.
5-6	The student evaluates the success of the robot in an objective manner based on the **results of testing**. The **student describes possible uses of the type of HiTechnic sensor in real world applications.**. The student provides an evaluation of his or her own performance **at each stage of the design cycle** and suggests improvements.

Personal Assistant: Smart Housekeeping

Introduction to Mobile Robotics > End of Project Activities > Smart Housekeeping

The Assignment:

The dog! That darn dog! For the third time this week, the housekeeping staff at the Mini mansion has quit because their adorable 200-pound St. Bernard has trashed the place again. Mr. Mini is losing hair and sleep over the problem. He can't bear to part with the dog, but he's having more and more trouble finding people who are willing to keep the place clean for any amount of money!

Mrs. Mini, however, has heard about your Personal Assistant robot through a friend, and wants to know if the robot could perhaps be adapted to serve as a robotic housekeeper for the Minis and their dog. The robot would have to do a few basic chores, like taking out the trash but mostly, it just needs to be able to keep up with the dog!

The Details:

Your robot must be able to perform the following tasks on a game board modeled after the Mini mansion:

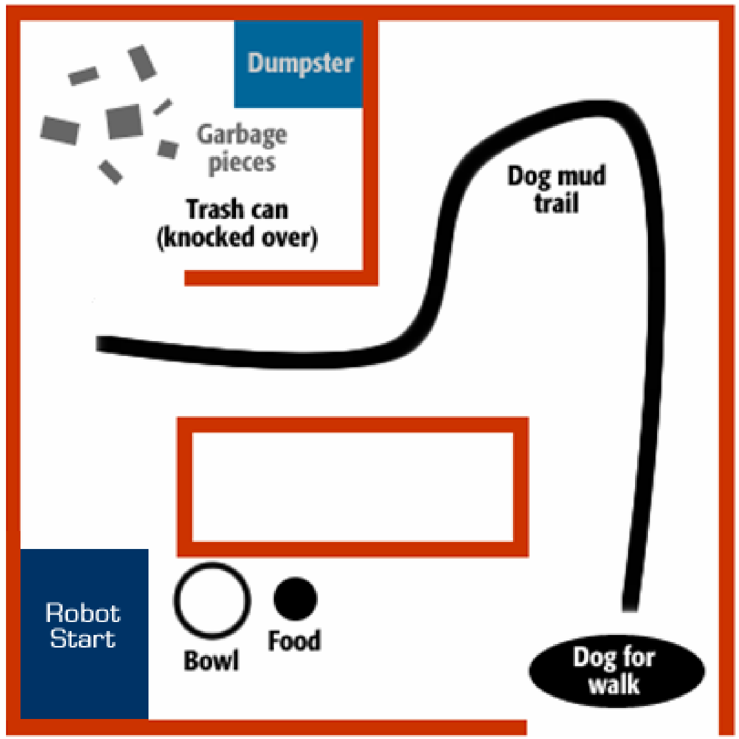
- Stay in low-power mode until the barking dog indicates that the robot will be needed again. Sigh.
- Move through the house cleaning up after everything the dog has done this time.
    - The dog has probably knocked over the trash so the robot must push all the trash over into the dumpster area.
    - The dog has also tracked mud through the house the robot must follow the trail as far as it goes, cleaning it up along the way (fortunately, the bottom of the robot automatically cleans any spot the robot passes over).
    - After all the mischief, the dog is waiting at the door to be walked the robot needs to let the dog out and walk it one full lap around the yard.
    - The dog also ate all its food the robot should fill the food bowl with the food that is right next to it.
- After this, the robot must then return to its waiting spot, and wait for the next time it's needed.

Some additional notes and requirements:

- The robot must have a manual shutoff button so that someone could easily turn the robot off at any time. The dark grey button on the NXT does not count.
- The robot has to be able to do everything to take care of the house and the dog, but not necessarily all in one run, or using only one program. Each task or few tasks can be done with a different program, just so long as everything gets done!

<u>This is a team assignment.</u> You should collaborate with your team members to do any and all of the work for this assignment, and you should distribute the work in a fair and effective way. All work done, however, must be your own group's. If you have any questions about whether something is allowable, ask your teacher *in advance*.

Gameboard Layout



Ask your teacher for instructions to set up the game board.

Tips:

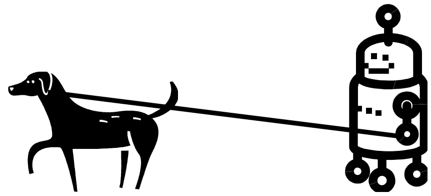
- Your robot will be expected to complete all of the four different tasks on competition day.
- Concentrate on developing a solution step by step. Don't try to get it all at once or things will get out of hand very quickly.
- Try using Move blocks to control the robot. It will make your programs much shorter and smaller than using Motor blocks.
- Teamwork is crucial. The challenge is too much for one person to get done alone in time. Try dividing up tasks for different team members.
- Test and refine your design as many times as possible. The robot must be ready for whatever comes its way.

Deadlines:

Progress check 1:

Progress check 2:

Final competition:



PROTOTYPE THIS *Mindstorms Robo Sitter!*
Digital Design Folder Grading Criteria *Due Friday November 18, 2011*

Criterion A: Investigate
Maximum 6
Investigation is an essential stage in the design cycle. Students are expected to identify the problem, develop a design brief and formulate a design specification. Students are expected to acknowledge the sources of information and document these appropriately.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student **states** all of the tasks the Robositter must complete. The student investigates how Robositter tasks could be met by **collecting** information about possible Mindstorms designs from a variety of sources. The student lists some specifications.
3-4	The student **describes** the Robositter tasks and their relevance to a full scale robot. The student investigates how Robositter tasks could be met by **collecting** information about possible Mindstorms designs from a variety of **documented** sources. The student lists all specifications for the tasks and has **described** a test to **evaluate** the robot against at least one design specification.
5-6	The student **describes** the Robositter tasks and their relevance to a full scale robot. The student investigates how Robositter tasks could be met by **collecting** information about possible Mindstorms designs from a variety of sources. The student lists all specifications for the tasks and **describes multiple** tests to **evaluate** the robot against more than one design specification.

Design specifications: A detailed description of the conditions, requirements and restrictions with which the Robositter must comply. This is a precise and accurate list of facts such as conditions, dimensions, hardware and software that are important for the designer and for the user. All appropriate solutions will need to comply with the design specifications.

Criterion B: Design
Maximum 6
Students are expected to generate several feasible designs that meet the design specifications and to evaluate these against the design specifications. Students are then expected to select one design, justify their choice and evaluate this in detail against the design specifications.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student generates **one Robositter** design documented **with photos** in their Engineering Logbook, and makes **some attempt to justify** this against the design specifications.
3-4	The student generates **two Robositter** designs documented **with photos** in their Engineering Logbook, **justifying** the choice of **one** design and fully **evaluating** this design against the design specifications.
5-6	The student generates **two Robositter** designs documented **with photos** in their Engineering Logbook, **each evaluated** against the design specifications and **justify** their final design choice.

Criterion C: Plan
Maximum 6
Students are expected to construct a plan to create their chosen product/solution that has a series of logical steps, and that makes effective use of resources and time.
Students are expected to evaluate the plan and justify any modifications to the design.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student produces a plan that includes a **flowchart** for programs used.
3-4	The student produces a plan that includes a **flowchart** for all programs used and contains a **prototype timeline**. The student makes some attempt to evaluate the plan.
5-6	The student produces a plan that includes a **flowchart** for all programs used and contains a **prototype timeline** for the **entire design cycle**. The student critically evaluates the plan and justifies any modifications to the design.

Criterion D: Create
Maximum 6
Students are expected to document, with a series of photographs or a video and a dated record, the process of making their product/solution, including when and how they use tools, materials and techniques. Students are expected to follow their plan, to evaluate the plan and to justify any changes they make to the plan while they are creating the product/solution.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student considers the plan and creates a Robositter that completes **one task either partially or completely.**
3-4	The student **uses** appropriate techniques and equipment. The student follows the plan and **mentions** any modifications made, resulting in a Robositter that completes **one task completely and a second task either partially or completely.**
5-6	The student **competently uses** appropriate techniques and equipment. The student follows the plan and **justifies** any modifications made, resulting in a Robositter that completes **two tasks completely and a third task either partially or completely.**

Criterion E: Evaluate
Maximum 6
Students are expected to evaluate the product/solution against the design specification in an objective manner based on testing, and to evaluate its impact on life, society and/or the environment. They are expected to explain how the product/solution could be improved as a result of these evaluations.
Students are expected to evaluate their own performance at each stage of the design cycle and to suggest ways in which their performance could be improved.

Level of Achievement	Descriptor
0	The student has not reached a satisfactory standard in any aspect listed above.
1-2	The student evaluates the Robositter **or** his or her own performance. The student makes some **attempt to test** the Robositter against the design specifications.
3-4	The student evaluates the Robositter **and** his or her own performance and suggests ways in which these could be improved. The student **tests** the Robositter to evaluate it against the design specifications.
5-6	The student evaluates the success of the Robositter in an objective manner based on the **results of testing**, and the **possible views of the intended users of a full scale Robositter**. The student provides an evaluation of his or her own performance **at each stage of the design cycle** and suggests improvements. The student provides an appropriate evaluation of the **impact** of the Robositter on life and society.

Design Cycle Project 1: *SUMO ROBOTS!*
Digital Design Folder Grading Criteria *Due Friday May 31, 2011*

Criterion A: Investigate
Maximum 6

Investigation is an essential stage in the design cycle. Students are expected to identify the problem, develop a design brief and formulate a design specification. Students are expected to acknowledge the sources of information and document these appropriately.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student **states** all of the tasks the Sumo Robot must complete. The student investigates how Sumo Robot tasks could be met by **collecting** information about possible Mindstorms Sumo Robot designs from a variety of sources. The student lists some specifications.
3-4	The student **describes** the Sumo Robot tasks and their relevance to winning the sumo robot competition. The student investigates how Sumo Robot tasks could be met by **collecting** information about possible Sumo Robot Mindstorms designs from a variety of **documented** sources. The student lists all specifications for the tasks and has **DESCRIBED A TEST TO EVALUATE** the robot against at least one design specification.
5-6	The student **describes** the Sumo Robot tasks and their relevance to winning the sumo robot competition. The student investigates how Sumo Robot tasks could be met by **collecting** information about possible Sumo Robot Mindstorms designs from a variety of sources. The student lists all specifications for the tasks and **DESCRIBED MULTIPLE TESTS TO EVALUATE** the robot against more than one design specification.

Design specifications: A detailed description of the conditions, requirements and restrictions with which the Sumo Robot must comply. This is a precise and accurate list of facts such as conditions, dimensions, hardware and software that are important for the designer and for the user. All appropriate solutions will need to comply with the design specifications.

Criterion B: Design
Maximum 6

Students are expected to generate several feasible designs that meet the design specifications and to evaluate these against the design specifications. Students are then expected to select one design, justify their choice and evaluate this in detail against the design specifications.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student generates **one Sumo Robot** design documented **with photos** in their Engineering Logbook, and makes **some attempt to justify** this against the design specifications.
3-4	The student generates **two Sumo Robot** designs documented **with photos** in their Engineering Logbook, **justifying** the choice of **one** design and fully **evaluating** this design against the design specifications.
5-6	The student generates **two Sumo Robot** designs documented **with photos** in their Engineering Logbook, **each evaluated** against the design specifications and **justify** their final design choice.

Criterion C: Plan
Maximum 6
Students are expected to construct a plan to create their chosen product/solution that has a series of logical steps, and that makes effective use of resources and time.
Students are expected to evaluate the plan and justify any modifications to the design.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student produces a plan that includes a **flowchart** for programs used.
3-4	The student produces a plan that includes a **flowchart** for all programs used and contains a **prototype timeline**. The student makes some attempt to evaluate the plan.
5-6	The student produces a plan that includes a **flowchart** for all programs used and contains a **prototype timeline** for the **entire design cycle**. The student critically evaluates the plan and justifies any modifications to the design.

Criterion D: Create
Maximum 6
Students are expected to document, with a series of photographs or a video and a dated record, the process of making their product/solution, including when and how they use tools, materials and techniques. Students are expected to follow their plan, to evaluate the plan and to justify any changes they make to the plan while they are creating the product/solution.

Level of Achievement	Descriptor
0	The student has not reached a standard described by any of the descriptors given below.
1-2	The student considers the plan and creates a Sumo Robot that **wins one or two matches.**
3-4	The student **uses** appropriate techniques and equipment. The student follows the plan and **mentions** any modifications made, resulting in a Sumo Robot that **wins three or four matches.**
5-6	The student **competently uses** appropriate techniques and equipment. The student follows the plan and **justifies** any modifications made, resulting in a Sumo Robot that **wins five or six matches.**

Criterion E: Evaluate
Maximum 6
Students are expected to evaluate the product/solution against the design specification in an objective manner based on testing, and to evaluate its impact on life, society and/or the environment. They are expected to explain how the product/solution could be improved as a result of these evaluations.
Students are expected to evaluate their own performance at each stage of the design cycle and to suggest ways in which their performance could be improved.

Level of Achievement	Descriptor
0	The student has not reached a satisfactory standard in any aspect listed above.
1-2	The student evaluates the Sumo Robot **or** his or her own performance. The student makes some **ATTEMPT TO TEST** the Sumo against the design specifications.
3-4	The student evaluates the Sumo Robot **and** his or her own performance and suggests ways in which these could be improved. The student **TESTS** the Sumo Robot to evaluate it against the design specifications.
5-6	The student evaluates the success of the Sumo Robot in an objective manner based on the **RESULTS OF TESTING**, The student provides an evaluation of his or her own performance **at each stage of the design cycle** and suggests improvements.