HOW TO MAKE YOUR ROBOT SMARTER

A Project

Presented to

The College of Graduate and Professional Studies

Department of Curriculum, Instruction, and Media Technology

Indiana State University

Terre Haute, Indiana

In Partial Fulfillment

of the Requirements for CIMT 620

Instructional Design

Professor: Dr. Timothy Boileau

---

by

Haisong Ye

December 12, 2012

Contents

List of Tables

List of Figures

HOW TO MAKE YOUR ROBOT SMARTER

Project A: Learning Context Analysis

Introduction

Switch and loop control flows are the fundamental of computer programing. In this project, it will teach 4th grade students to use switch and loop control flows in their earlier robotic projects. The goal is to teach the learners how to use basic control flows to solve real problem.

Needs Assessment

Needs assessment plan. I will use interview and content analysis to get information. During the interview, I will ask 7 questions (as show in table 1). After the interview, I will collect all the instructional materials which be mentioned in interview question 6 and conduct a content analysis.

Table 1

Interview questions

Interview question	Note
1. What is your grade? (4th / 5th grade)	General information.
2. Do you like your NXT? What will you plan to do with your NXT? What have you done in your project?	Warm up questions. Record how they answer these questions.
3. Do you have your own computer and NTX 2 bricks to do the project?	Hardware support. (it's for learning environment analysis)
4. Have you used switch block in your project? If yes, a. Can you give me one example which you used switch block? If no, b. How would you program your robot to let it turn around when it hit the wall?	Gather information on the SWITCH block. The interviewees may have used or never used switch blocks.

5. Have you used loop block in your project? If yes, a. Can you give me one example which you used loop block? If no, b. How would you program your robot to repeat saying HELLO and HAVE A NICE DAY?	Gather information on the LOOP block. The interviewees may have used or never used loop blocks.
6. Now, I will give you a problem. You will use the robot car on the table and program it to solve the problem.	In this question, we have designed 3 problems. 1st is for the interviewees who only know SWITCH block, 2nd is for those who only know LOOP block, and the last one is for those who know both SWITCH and LOOP blocks. While they are programming, observe their steps. The interviewees who never used SWITCH and LOOP blocks will skip this question.
7. Where did you learn all those things from?	Gather information about existing instructional materials.

Needs assessment report. According to the interview, I got following information.

What should be? For this specific module, learners should be able to

1. Describe the purpose of switch flow with example.
2. Describe the purpose of loop flow with example.
3. Recognize the **Loop** block in toolbox.
4. Recognize the **Switch** block in toolbox.
5. Identify a **Loop** block or a **Switch** block in a program.
6. When given conditions which are need decision making in a program, choose the correct control method and other parameters in a **Switch** block.

7. When given a situation which is need to repeat action(s) in a program, choose the correct control method and parameters in a **Loop** block.
8. When given a compound problem which need decision making and repeating actions in a program, solve the problem with a **Switch** and a **Loop** blocks.
9. When given a real problem, solve the problem with a Switch and a Loop blocks along with prior skills.

How have the identified goals been already being achieved? A total of 12 interviewees (60%) have used the switch block; 14 interviewees (70%) have used the loop block; 6 of interviewees (30%) have used both the Switch and the Loop blocks. However, only 2 interviewees (10%) can show their capability to solve real problem (in question 6) by using the Switch or the Loop block.

Determine the gaps. According to the learning tasks, students could only demonstrate the basic concepts of switch and loop control flows (tasks No.1~5), but couldn't use the two blocks fluently (tasks No.6~9) in problem solving. Therefore, tasks No.6~9 are the gaps.

Prioritize gaps. The most important gap is task No.9, and it is the most challenge one. However, tasks No.6~8 are the fundamental for task No.9.

Determine the need for instruction. Tasks No.6~8 can be improved by using strategies for learning procedures; Task No. 9 can be improved by using strategies for problem-solving. So, all gaps are appropriate for design and development of instruction.

Define goal statement. Given a real problem, learners will be able to use the Switch and the Loop blocks in NTX-G to program a NXT robot and solve the problem.

The target learners are 4th graders who interested in robotic programming and have basic arithmetic skills. The tools are LEGO® NXT2 bricks and NTX-G software.

Description of Learning Environment[1]

Existing curricula. There is a related curriculum, Information Literary, for elementary students. There are existing instructional materials, including tutorials and books from LEGO® Company and third parties. However, most of them are too difficult to understand, particularly for 4th graders. Students who use LEGO® tutorials could walk through the tutorial. They could finish the tutorial, and make a wonderful robotic machine. But when they were facing real problem, they didn't have any ideas. Therefore, the existing instructional materials can only teach concepts and rules, but not problem solving. The new instruction will teach problem solving skills.

Users of the module. A total of 18 interviewees (90%) have great ideas about what their robot should be like, and all of the interviewees showed their capability of creative thinking. So, they have motivation in learning how to program a robot machine.

A total of 10 interviewees (50%) who never use SWITCH or LOOP block also showed their logical thinking capability. They can describe the problem in general language correctly, so they have the potential to master the two control flows.

The instruction needs some arithmetic skills. A fourth grader should have achieved following standards (Indiana Department of Education, 2009) in third grade:

- Students solve problems involving addition and subtraction of whole numbers. They model and solve simple problems involving multiplication and division.

[1] This instruction will be a self-instructional module, therefore no instructor is needed.

- Students select appropriate symbols, operations, and properties to represent, describe, simplify, and solve simple number and functional relationships.
- Students describe and compare the attributes of plane and solid geometric shapes and use their understanding to show relationships and solve problems.
- Students choose and use appropriate units and measurement tools for length, capacity, weight, temperature, time, and money.
- Students make decisions about how to approach problems and communicate their ideas.

Therefore, a 4th grader already has the basic mathematic skills for the new instruction.

Available media. This instruction need a computer and a set of LEGO® NXT2 bricks, and most target learners have them, because they have been ready for the competition.

Community climate. Science, Technology, Engineering, and Math (STEM) education will help produce the workforce needed to compete in a global marketplace. More and more researches are focused on STEM education. Robot designing is a comprehensive field in STEM education. It integrates mathematics, physical, engineering and computer technology. This instruction will help student improve their computer programming skills, and also helpful for STEM education.

Project B: Learner Analysis

This project analyzes the target learners of the training module. The target learners of this training module are fourth graders (ages from 9 to 10) who are interested in robotic programming and have basic knowledge in arithmetic and science. In my project, there are 20 students who are interested in robotic programming from a local elementary school. They all have experience in LEGO® NXT programming as a beginner.

Cognitive Characteristics

In this group of students, they are all advanced in mathematics. They have stronger logical thinking capability than average. They understand some basic science concepts, such as weight, length, speed, and time. They also can measure length and time. They can do conversion between different units in different unit of measure system.

According to Piaget's stages of cognitive development, a child of age 9/10 is at the stage of "Concrete Operational". At this stage, learners have the abilities to do arithmetic, understand volume and weight. They also can think logically if given concrete objects but will have difficulty in abstract verbal description or symbols.

Based on their academic records, all of my target learners meet the standards of Indiana Common Core State Standards (Indiana Department of Education, 2009) in English/Language Arts and Mathematics for 4th Grade. They have reading and writing capabilities at 4th grade level.

Most of students in my group are visual learners. Because of their age, they prefer pictures and demonstrations more than text. In my skill level test, the items which described only in words made them a little confusing.

During the skill level test, we also observed students' concentration time. The pretest was designed to be a one hour test. However, 15 students had signs of distracting after 45 minutes.

Physiological Characteristics

In this group, students are 9 or 10 years old. They are all very healthy and don't have learning disabilities.

Affective characteristics. In this group, students are interested in robotic programming and they already have the experience as a beginner in robotic programming. They might have already finished the tutorials provided by LEGO® NXT software, and liked it very much. Therefore, they took part in my instruction voluntarily. When they saw their robot machines moving, talking, rotating, or flashing, they were excited and satisfied. My instruction need to keep their motivation to attract them to solve more difficult problems.

Social characteristics. In my group, most of the students would like to be a robot designer or an inventor in the future. Most of them like to share their ideas and discuss with other students. When we talked about what you want to do in this project, they all gave me great ideas. Even some ideas were more beyond this instruction, I still can feel their passionate.

Learners' Prior Knowledge. A skill level test was conducted for this group of students. The skill level test would find out whether they already have the prerequisite knowledge and skills (as shown in Table 1) for this instruction; how much they know about the learning tasks (as shown in Table 2) of this instruction.

Table 2

Prerequisite knowledge and skills.

No.	Description
P1	Identify different shapes of bricks
P2	Identify color sensor, touch sensor, ultrasound senor, and motors
P3	Build a machine with a building guide
P4	Connect sensors and motors using connectors
P5	Connect LEGO® MINDSTORM brick to NXT software via USB cable or Bluetooth
P6	Create and save a NXT program
P7	Use *Move* block(s) to drive motors
P8	Use *Sound* block(s) to let a brick play (a) sound file(s)
P9	Use *Display* block(s) to show (a) picture(s)
P10	Use *Wait* block to let brick stop action for a specific time
P11	Make a liner program flow to perform single action or a sequence of actions
P12	Download a NXT program to MINDSTORM brick and run the program on brick
P13	When a problem occurred, locate the problem and fix the problem.

Table 3

Target knowledge and skills.

No.	Description
T1	Recognize the *Loop* block in toolbox
T2	Recognize the *Switch* block in toolbox
T3	Identify a *Loop* block or a *Switch* block in a program
T4	When given conditions which are need decision making in a program, choose the correct control method and other parameters in *Switch* block
T5	When given a situation which is need to repeat action(s) in a program, choose the correct control method and parameters in *Loop* block
T6	When given a compound problem which need decision making and repeating actions in a program, solve the problem with *Switch* and *Loop* blocks
T7	When given a real problem, solve the problem with *Switch* and *Loop* blocks along with prior skills

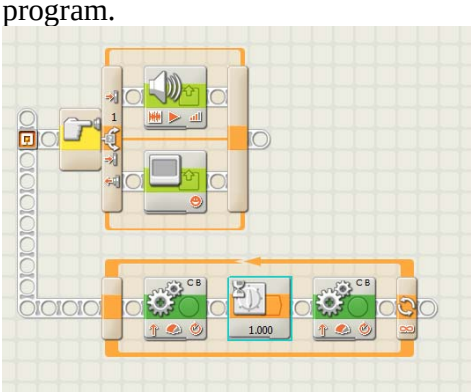
Table 4

Skill level test items and explanations

Item	Explanation

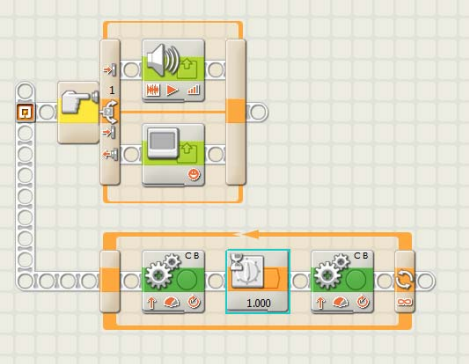
1. Please assemble your robo-car through following steps.	A walkthrough guide will be presented to test takers. This guide will help them build a robot car with LEGO® bricks. In this item, test takers will be assessed for P1~ P4.
2. Please build a program which can let the robo-car you made before to: • Go straight forward for two seconds with 75 percent power and then stop. • Say "Hello" and display a heart on brick's screen. • Wait for two seconds. • Make a u turn and go back to the start point. Download the program to your robo-car and run it.	This item will assess P4~12. In this test, test takers will create a new program (P5), use *Move* block (P7) to drive the motors, *Sound* block (P8) to play sound file, *Display* block (P9) to show picture, and *Wait* block (P10) to stop the robo-car for seconds. This program only needs one liner flow control. Actions will happen one-by-one (P11). Test takers are asked to download and run the program (P6 and P12) If there are problems happened at running phase, a supervisor also can observe and assess test taker's program locating skill and debugging skill (P13).
3. Please circle the *Switch* block. 	This item will assess T1.
4. Please circle the *Loop* block. 	This item will assess T2.

5. Please circle the switch part in following program.



This item will assess T3.

6. Please circle the loop part in following program.



This item will assess T3.

7. If the robo-car's color sensor found a colored ball, which block you would use to judge the color, *Switch* or *Loop*?

This item will assess T4 and T5.

8. Please build a program which can let the robo-car to:

- Go straight forward with 75 percent power.
- When the color sensor sees red color, wait for on second and say "Red", then continue going forward.
- When the color sensor sees green color, say "Green" then stop.

Download the program to your robo-car and run it.

This item will assess T1-T6. In this test, test takers will use *Switch* block to determine the color, *Loop* block to do the color detecting job till the color sensor sees green color.

Not all test takers can finish this item. So, if they think they cannot complete the task, they can stop the test here.

The skill level test has 7 items as shown in Appendix. After the entry test, two students demonstrated lacking the knowledge for P5; eight students had trouble in problem locating (P13); there were no problems in other prerequisite knowledge. For target learning tasks, all students could recognize the *Loop* block and *Switch* block in toolbox (T1&T2); three students could identify both blocks in a program (T3); one student chose right blocks to use (T4 and T5), but they couldn't use these blocks.

Processing Style of Instruction for the Learner

Because most of the students in this group are visual leaners, this instruction will use a full pictures guide tutorial plus recorded video playbacks. The learning goals are more complex than what in school curriculum, so cognitive load is a problem for elementary students.

Implications for Design

According to the result of learner analysis, we have following implications for design:

- Tips for prerequisite knowledge and how to locate relative information. Learners showed that they might have different knowledge level in programming. Therefore, some tips or information should be provided for them.
- Use simple language with funny and attractive illustrations. Most learners in ages from 9 to 10 are visual learners. Therefore illustrations will help them understand the content. Fourth graders cannot understand sophisticated words, so simple language will be better for them.
- Paper materials and multimedia files (mostly video files). Videos and motion screen captures might help learners repeat or follow the steps of examples.
- Use terms in NXT software instead of terms in computer science or engineering.

- Use square (8.5’’×8.5’’) paper to deliver our paper based material to provide less information at one time.

Project C: Learning Task Analysis

This instruction is a training module for 4th graders (target learner) who are interested in graphical robotic programing in LEGO® MINDSTORMS NXT 2. Project C is will analyze learning task through following steps (Smith & Ragan, 2005):

- Write a learning goal.
- Determine the types of learning of the goal
- Conduct an information-processing analysis of that goal.
- Conduct a prerequisite analysis and determine the type of learning of the prerequisites.
- Write learning objectives for the learning goal and each of the prerequisites.

Instructional Goal and Learning Domain

In this instructional module, learners will learn to use *Switch* block and *Loop* block in their program to perform complex tasks like getting information, making decision, and responding. The instructional goal is:

When given a task which will use *Switch* and *Loop* blocks, the learner will be able to design a program with correct blocks, and run the program on a provided robot car.

According to Gagné's types of learning outcomes this learning is the type of problem solving. To accomplish this learning goal, the learner should apply the principles for selecting blocks and sensors and follow the procedures to set parameters and to test a program.

Information Processing Analysis

To perform information processing analysis, the learning goal will be converted to a test question first(Smith & Ragan, 2005). Following is a simple representative question for the learning goal, “The robot car goes forward on 75% power. When it sees a red line, it will stop at the line”.

As show in Figure 1, an expert will start with analyze tasks for this problem. The tasks for the test question are,

- The car goes forward (action).
- When it sees a red line (condition), it will stop at the line (action).

Step 2 is to analyze the tasks. The target task here is “When it sees a red line (condition), it will stop at the line (action)”.

Step 3 is to select proper block. For this task, a *Switch* block is selected for condition and a *Stop* block is selected for action.

Step 4 is to set proper parameters. The expert places a *Switch* block on default thread of a program, selects the color sensor, and sets parameter for red color (stop the car).

Step 5 is to test the program on the provided robot car. If the task is accomplished, the expert will move to next task. If not, he will re-analyze the sub-task, and check whether he made a right decision and set correct parameters.

After all tasks are finished, the expert will test the whole program to check whether the problem is solved. If not, he will re-define the tasks.

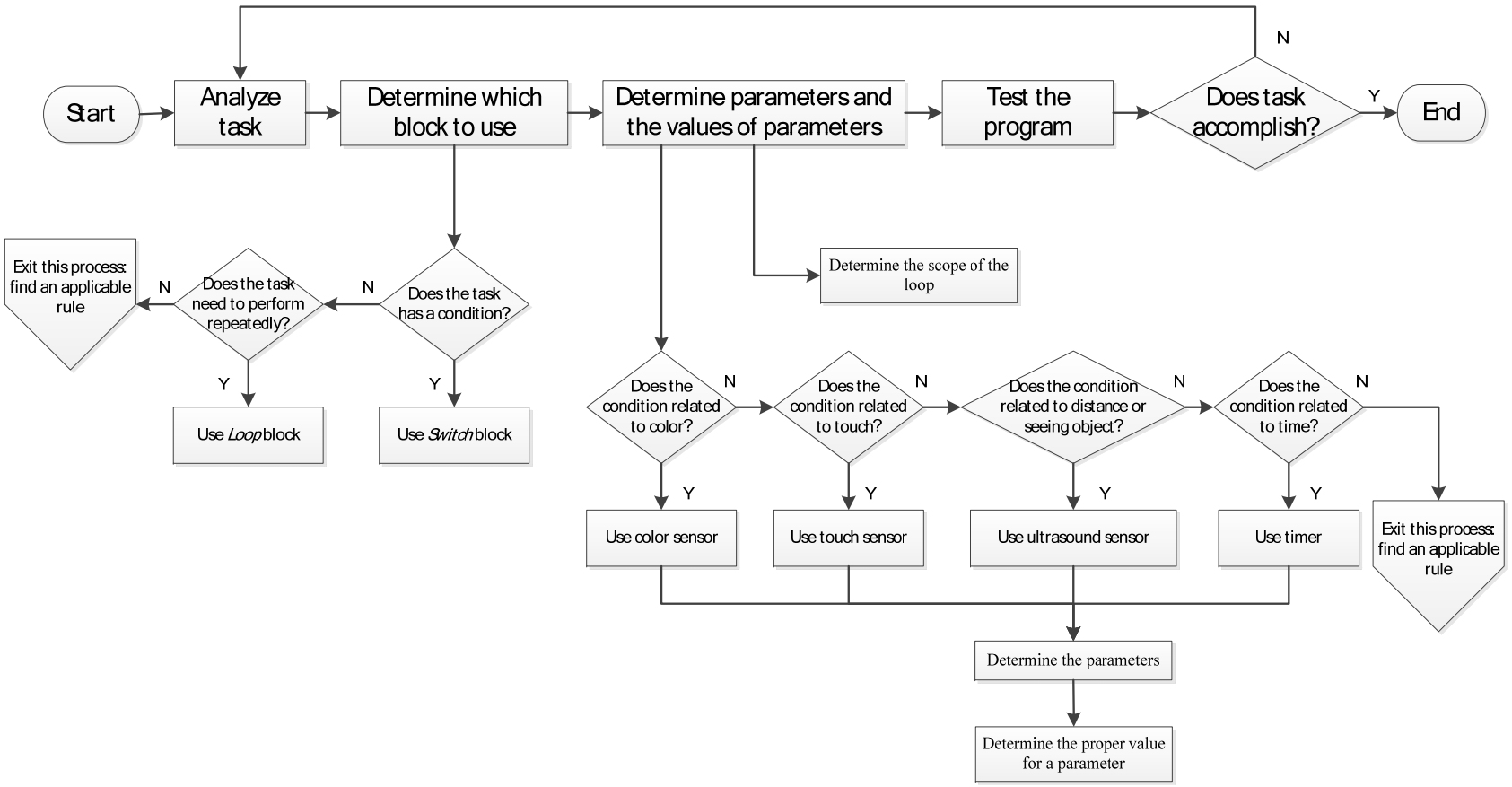


Figure 1 Information-processing analysis for the question

Prerequisite Analysis

For the information-processing of the learning goal, we determined the prerequisites (as shown in figure 2, gray square represent prerequisites; dashed lines refer to support):

1. Analyze task.
    a. Recall task types (action, condition, and loop). (Declarative)
    b. Determine the type of a task. (Principle)
2. Select proper blocks.
    a. Recognize Switch block and Loop block in common palette. (Discrimination)
    b. Identify the scope of loop section. (Discrimination)
    c. When given the type of a task, select the proper blocks. (Principle)
3. Set proper parameters.
    a. Identify sensor type. (Discrimination)
    b. Identify the types of condition of a switch block. (Principle)
    c. Identify connection port for each sensor and motor. (Discrimination)
    d. Determine the proper value for a parameter. (Procedure)
4. Test/debug the program.
    a. Download and run the program on a robotic car. (Procedure)
    b. Identify switch sections and loop sections in a program. (Concept)

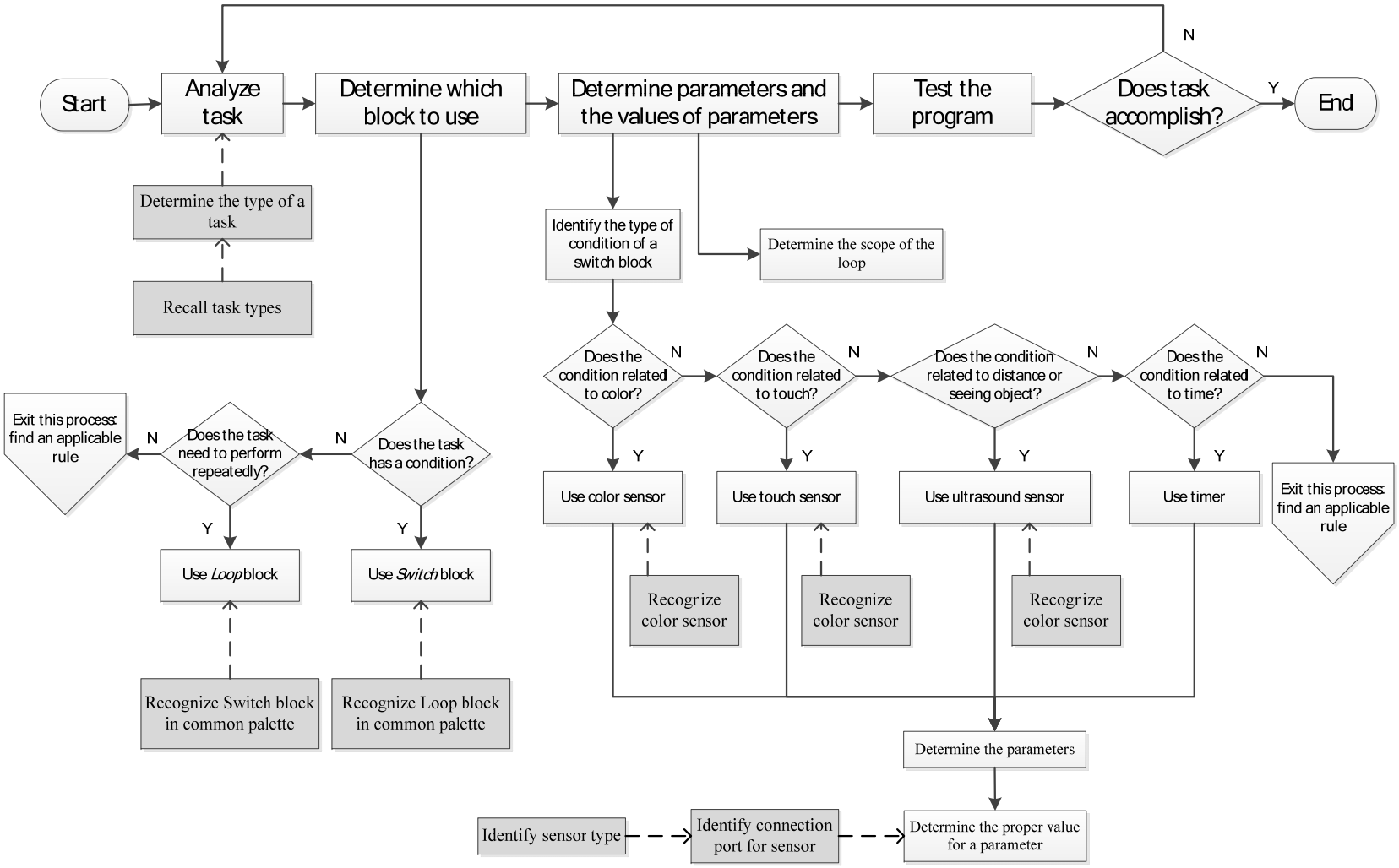


Figure 2 Prerequisite analysis for learning goal.

Learning Objectives

After information-processing and prerequisite analysis, following learning objectives of this instructional module are set:

1. Given a task statement, learners can identify action/condition tasks.
2. Given the common palette, learners can recognize the *Loop* block.
3. Given the common palette, Learners can recognize the *Switch* block in.
4. Given a program, learners can identify a loop section or a switch section.
5. Learners can identify the types of condition of a *Switch* block and explain each type with examples.
6. Given a condition task, learners can identify the type of the condition.
7. Given a pair of condition and action tasks, learners can use *Switch* block to complete the tasks.
8. Given a problem which needs repeat action(s) in a program, learners can define the repeat scope and use *Loop* block to complete the task.
9. Given a compound problem which need decision making and repeating actions in a program, learners can solve the problem with *Switch* and *Loop* blocks

Project D: Objectives and Assessments

Introduction

This instruction is a training module for 4th graders (target learner) who are interested in graphical robotic programing in LEGO® MINDSTORMS NXT 2. The instructional goal is: When given a task which will use *Switch* and *Loop* blocks, the learner will be able to design a program with correct blocks to complete the task, and run the program on a provided robot car. Project D will summarize terminal and subordinate objectives at first, then conduct assessment blue print for entry skill assessment, pre- and post-assessment. Test examples for each type of assessment are provided.

Performance Objectives

Terminal and subordinate objectives.

O.1 Given a conditional task, learners can identify the conditions and actions, and rewrite the task in ITE (If…Then…Else…) pattern. (Procedure)

O.1.1 Learners can identify the part of condition and the part of action in a conditional task description. (Concept)

O.1.2 Given conditions and actions, learners can compose a conditional task in ITE pattern. (Principle)

O.2 Given an ITE task, learners can use a Switch block to complete the task. (Problem solving)

O.2.1 Learners can recognize the Switch block in common palette. (Discrimination)

O.2.2 In three types of condition, learners can determine the type of condition of a task. (Principle)

O.2.3 Given a color condition, learners can use color sensor in Switch block and set correct parameters to identify black, blue, green, yellow, and red colors. (Procedure)

O.2.4 Given a touch condition, learners can use touch sensor in Switch block and set correct parameters to accept three touch actions. (Procedure)

O.2.4.1Learners can define three touch actions (pressed, released, and bumped). (Concept)

O.2.5 Given an ultrasonic condition, learners can use ultrasonic sensor in Switch block and set correct parameters to detect the distance of an object. (Procedure)

O.3 Given a task, learners can determine whether actions in the task need to perform repeatedly. (Principle)

O.4 Given a task with repeatedly actions, learners can use a Loop block to complete the task. (Problem solving)

O.4.1 Learners can recognize the Loop block in common palette. (Discrimination)

O.4.2 In four types of loop condition, learners can determine the type of condition to end the loop. (Principle)

O.4.2.1Learners can explain four types of loop condition (forever, sensor, count, and time) with example. (Concept)

O.4.3 Given a forever loop condition, learners can set a forever loop. (Procedure)

O.4.4 Given a color sensor loop condition, learners can set the color condition to end the loop. (Procedure)

O.4.5 Given a touch sensor loop condition, learners can set the touch action to end the loop. (Procedure)

O.4.6 Given a time loop condition, learners can set the timer to end the loop. (Procedure)

O.4.7 Given a count loop condition, learners can set the counter to end the loop. (Procedure)

O.4.8 Given an ultrasonic sensor loop condition, learners can set the distance value to end the loop.

Entry skills.

S1 Learners can identify color sensor, touch sensor, ultrasound senor, and motors.

S2 Learners can connect sensors and motors using connectors.

S3 Learners can connect LEGO® MINDSTORM brick to NXT software via USB cable or Bluetooth.

S4 Learners can create and save a NXT program.

S5 Learners can use *Move* block to drive motors.

S6 Learners can use *Sound* block to let a brick play sound files.

S7 Learners can use *Display* block to show pictures.

S8 Learners can use *Wait* block to let brick stop action for a specific time.

S9 Learners can make a liner program flow to perform single action or a sequence of actions.

S10 Learners can download a NXT program to MINDSTORM brick and run the program on brick.

Assessments

Some of the assessment items might need to assemble physical parts and design a program using a computer. Grade guidelines are provided for both pre- and post-assessment. Learners are encouraged to evaluate themselves using these guidelines. However, having an observer or proctor to conduct the evaluation will be better in this module.

Entry skills assessment.

Table 5

Instrument blueprint for entry skills assessment

	Form	Number of Items	Criterion Level	Proportion
S1	Matching	1	100%	.5
S2	Operation	1	100%	.5
S3	Operation	1	100%	.5
S4	Operation	1	100%	.5
S5	Multiple choices, operation	2	90%	.15
S6	Multiple choices, operation	2	90%	.15
S7	Multiple choices, operation	2	90%	.15
S8	Multiple choices, operation	2	90%	.15
S9,S10	Operation	2	90%	.20

Table 6

Entry skills assessment sample

Entry Skills Assessment

Time: 20 minutes

Name: _____ Date: _____

Section#	Section I	Section II	Total
Available score	13	77	100
Your score			

Section I: Matching question.

Directions: In this section, you will match the text item with the picture. Draw lines between matched items.

1. Match the block name with picture.

Move block

Sound block

Display block

Wait block

2. Match the sensor name with picture.

Color sensor

Touch sensor

Ultrasonic sensor

Section II: Operation questions.

Directions: In this section, you will be asked to assemble some parts or design a program to complete a given task. You can use NXT software and run the program on a robo-car.

3. There are three different sensors, a cable, and a brick on the desk. Please connect color sensor to port #3 of the brick.
4. There are a computer and an USB cable on the desk. Please connect the brick you used in previous question to the computer using an USB cable.
5. Create a new NXT program, and then save the program as “pp.rbt” on desktop.
6. Open the program you saved in question 4, and then complete following task:
    - Go straight forward for two seconds with 75 percent power and then stop.
    - Say “Hello” and display a heart on brick’s screen.
    - Wait for two seconds.
    - Make a u turn and go back to the start point.
    - Download and run the program.

If you successfully run the program on the robo-car, and completed all the task, then congratulations, you passed the entry skills assessment for this module.

Table 7

Pre-assessment grade sheet sample

Q#	**Objective**	**Available Points**	**Actual Points**
1	S5,6,7,8	8	
2	S1	5	

3	S2	5	
4	S3	5	
5	S4	5	
6	S5	13	
	S6	13	
	S7	13	
	S8	13	
	S9	10	
	S10	10	
		Total	

Pre-assessment.

Table 8

Instrument blueprint for pre-assessment

	Form	Number of Items	Criterion Level	Proportion
O1	Short answer	1	90%	.10
O1.1	Multiple choices	1	100%	.5
O1.2	Multiple choices	1	100%	.5
O2				
O2.1	Multiple choices	1	100%	.5
O2.2	Multiple choices	1	100%	.5
O2.2.1	Multiple choices			
O2.3	Operation	1~2	90%	.25
O2.4	Operation			
O2.5	Operation			
O3	Multiple choices	1	100%	.5
O4				
O4.1	Multiple choices	1	100%	.5
O4.2	Multiple choices	1	100%	.5
O4.2.1	Matching			

O4.3	Operation	1~2	90%	.30
O4.4	Operation			
O4.5	Operation			
O4.6	Operation			
O4.7	Operation			
O4.8	Operation			

Table 9

Pre-assessment sample

Pre-assessment

Time: 35 minutes

Name: _____ Date: _____

Section#	Section I	Section II	Section III	Total
Available score	30	15	55	100
Your score				

Section I: Multiple Choices

Direction: In this section, you will choose the best answer for each question.

1. ( ) Which one of the following tasks is a conditional task?

   a. Display a picture on screen.

   b. Play a sound.

   c. When a button pressed, stop the car.

   d. Go forward, don't stop.

2. ( ) Which one of the following blocks is Switch block?

   a. 

   b. 

c.  d. 

3. ( ) Which one of the following conditions is a color condition?

   a. When the robo-car sees a red line, it stops.

   b. When the robo-car sees an object, it stops.

   c. When the robo-car touches an object, it stops.

   d. When the robo-car runs 2 minutes, it stops.

4. ( ) Which one of the following actions needs a Loop block to repeat?

   a. Display a picture on screen.

   b. Play a sound for 50 times.

   c. When a button pressed, stop the car.

   d. Go forward, never stop.

5. ( ) Which one of the following blocks is a Loop block?

   a.  b. 

   c.  d. 

6. ( ) Which one of the following loops will use a color sensor to stop?

   a. A robo-car will play a sound for 50 times.

   b. A robo-car will display a picture for 5 minutes.

   c. A robo-car will always say “Green” when it sees a green ball until it sees a red ball.

d. A robo-car will shoot balls until a touch sensor pressed.

Section II: Short answer

Direction: In this section, you will answer the question using one sentence.

7. Use the following condition-action pair to compose an ITE sentence:

   Condition: sees red color; Action: says “Red”.

   _____

8. Find the condition-action pair in the following task and rewrite it using an ITE sentence:

   A robo-car will move forward. When the robo-car hit an object, it will backward and make a U-turn.

   _____

   _____

Section III: Operation

Direction: In this section, you will be asked to design a program to complete a task.

You can use NXT software and run the program on a robo-car.

9. Design a program to perform following task. You will use a robo-car to finish the task.

   A robo-car will move forward. When the robo-car sees object in 20 cm, it will say “Object” and stop.

10. Design a program to perform following task. You will use a robo-car to finish the task.

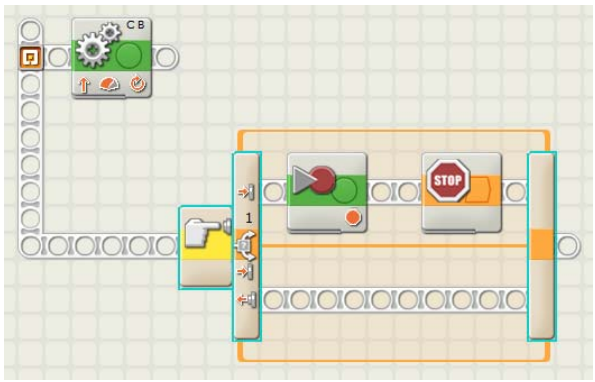
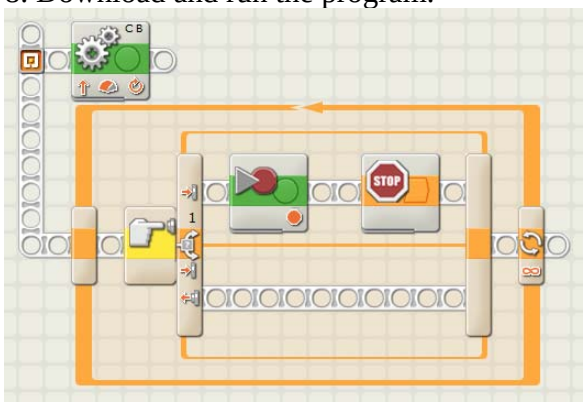
    A robo-car has been programed to move forward; When the robo-car sees object in 20 cm, it will say “Object” and stop.

    Redesign the program to let the robo-car continue to move forward after the object is not in sight.

Table 10

Pre-assessment grade sheet sample

Q#	**Objectives**	**Grade Guidelines**	**Available Points**	**Your Points**
1	O1.1	c	5	
2	O2.1	b	5	
3	O2.2	a	5	
4	O3	b	5	
5	O4.1	c	5	
6	O4.2	c	5	
7	O1.2	**If** car sees red color (2.5), **then** it will say "red". (2.5)	5	
8	O1	**If** car hits an object (3.5), **then** it will say backward and make a U-turn (3.5), **else** it will continue to move forward. (3)	10	
9	O2.3, 2.4, 2.5	1. Place a Move block and set to move forward on first thread line. (0) 2. Place a Switch block on second thread line. (4) 3. Select "sensor" for Control.(4) 4. Select "Touch Sensor" for Sensor.(4) 5. Check Port#1.(4) 6. Select "Pressed" for Action.(4) 7. Play sound and stop (1) on the first line of Switch branches (4) 8. Download and run the program.	25	

				
10	O4.3, 4.4, 4.5, 4.6, 4.7, 4.8	1. Place a Loop block on second thread line.(5) 2. Select all items in second thread except the loop block. (5) 3. Move the selected items into the Loop block. (5) 4. Delete the Stop block. (2) 5. Select the Loop block. (5) 6. Select “Forever” for Control. (5) 7. Place a Move block on the second line of Switch branches and set to move forward.(3) 8. Download and run the program. 	30	

Post-assessment.

Table 11

Instrument blueprint for post-assessment

	Form	Number of Items	Criterion Level	Proportion

O1	Short answer	1	90%	.10
O1.1	Multiple choices	1	100%	.5
O1.2	Multiple choices	1	100%	.5
O2				
O2.1	Multiple choices	1	100%	.5
O2.2	Multiple choices	1	100%	.5
O2.2.1	Multiple choices			
O2.3	Operation	1~2	90%	.25
O2.4	Operation			
O2.5	Operation			
O3	Multiple choices	1	100%	.5
O4				
O4.1	Multiple choices	1	100%	.5
O4.2	Multiple choices	1	100%	.5
O4.2.1	Matching			
O4.3	Operation	1~2	90%	.30
O4.4	Operation			
O4.5	Operation			
O4.6	Operation			
O4.7	Operation			
O4.8	Operation			

Table 12

Post-assessment sample

Post-assessment* **Time: 35 minutes** Name: _____ Date: _____				
Section#	Section I	Section II	Section III	Total
Available score	30	15	55	100

* Blue parts are different than pre-assessment

Your score				

Section I: Multiple Choices

Direction: In this section, you will choose the best answer for each question.

1. ( ) Which one of the following tasks is a conditional task?

   a. Display a picture on screen.

   b. When sees a red ball, says red.

   c. Play a sound.

   d. Go forward, don't stop.

2. ( ) Which one of the following blocks is Switch block?

   a. 

   b. 

   c. 

   d. 

3. ( ) Which one of the following conditions is a touch condition?

   a. When the robo-car sees a red line, it stops.

   b. When the robo-car sees an object, it stops.

   c. When the robo-car conflicts with an object, it stops.

   d. When the robo-car runs 2 minutes, it stops.

4. ( ) Which one of the following actions needs a Loop block to repeat?

   a. Play a sound forever

   b. Play a sound for 50 times.

   c. When a button pressed, stop the car.

   d. Go forward, never stop.

5. ( ) Which one of the following blocks is a Loop block?

a. 

b. 

c. 

d. 

6. ( ) Which one of the following loops will use a touch sensor to stop?

   a. A robo-car will play a sound for 50 times.

   b. A robo-car will display a picture for 5 minutes.

   c. A robo-car will always say “Green” when it sees a green ball until it sees a red ball.

   d. A robo-car will shoot balls until a touch sensor button pressed.

Section II: Short answer

Direction: In this section, you will answer the question using one sentence.

7. Use the following condition-action pair to compose an ITE sentence:

   Condition: touches an object; Action: says “Object”.

   _____

8. Rewrite the following task in to an ITE sentence:

   A robo-car will move forward. When the robo-car sees any object in 20cm, it will say “Object” and stop.

   _____

   _____

Section III: Operation

Direction: In this section, you will be asked to design a program to complete a task.

You can use NXT software and run the program on a robo-car.

9. Design a program to perform following task. You will use a robo-car to finish the task.

   A robo-car will move forward. When the robo-car sees a green line, it will say "Green" and stop. When the robo-car sees a red line, it will say "Red" and stop.

10. Design a program to perform following task. You will use a robo-car to finish the task.

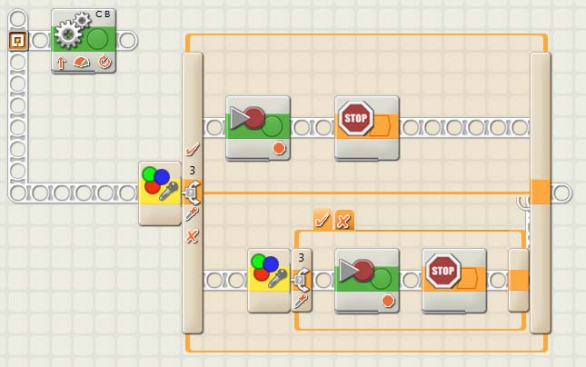
    A robo-car has been programed to move forward; When the robo-car sees a green line, it will say "Green" and stop.

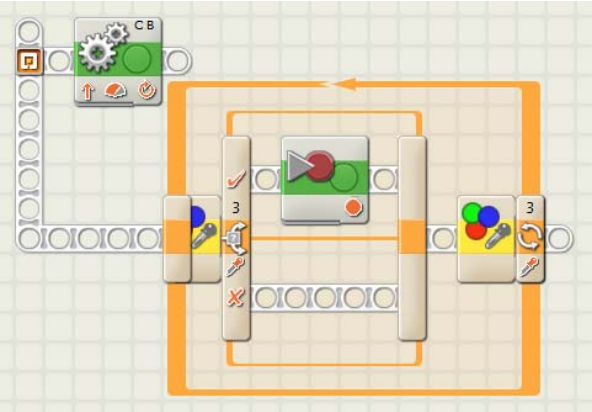
    Redesign the program to let the robo-car continue to move forward after it says "Green" until it sees a red line.

Table 13

Post-assessment grade sheet sample

Q#	**Objectives**	**Grade Guidelines**	**Available Points**	**Your Points**
1	O1.1	b	5	
2	O2.1	c	5	
3	O2.2	c	5	
4	O3	b	5	
5	O4.1	b	5	
6	O4.2	d	5	
7	O1.2	**If** car touches an object (2.5), **then** it will say "object". (2.5)	5	
8	O1	**If** car sees an object in 20cm (3.5), **then** it will say "object" and stop (3.5), **else** it will continue to move forward. (3)	10	

9	O2.3, 2.4, 2.5	1. Place a Move block and set parameter to move forward on first thread line. (0) 2. Place a Switch block on second thread line. (3) 3. Select “sensor” for Control.(3) 4. Select “Color Sensor” for Sensor.(3) 5. Check Port#3.(3) 6. Select “Color Sensor” for Action.(3) 7. Select “Inside Range” for Compare, and set color to red.(3) 8. Play sound and stop (0) on the first line of Switch branches (2) 9. Place a Switch block on the second line of Switch branches (1) 10. Repeat step from 2 to 7, but set color to green.(4) 11. Download and run the program. 	25	
10	O4.3, 4.4, 4.5, 4.6, 4.7, 4.8	1. Place a Loop block on second thread line.(3) 2. Select all items in second thread except the loop block. (3) 3. Move the selected items into the Loop block. (3) 4. Delete the Stop block. (3) 5. Select the Loop block. (3) 6. Select “Sensor” for Control. (3) 7. Select “Color Sensor” for Sensor. (3) 8. Select Port#3. (3) 9. Select “Color Sensor” in Action. (3) 10. Select “Inside Range” for Until, and set the color to red. (3)	30	

Project E: Instructional Strategy Development

Project Overview

This instruction is a training module for 4th graders (target learner) who are interested in graphical robotic programing in LEGO® MINDSTORMS NXT 2. The instructional goal is: When given a task which will use *Switch* blocks, the learner will be able to design a program with correct blocks to complete the task, and run the program on a provided robot car. Project E will develop instructional strategies for this lesson.

Instructional objectives. In project C and D, I discussed four main objectives. Because of the time limitation (1.5 hours' self-instruction), only objectives 1 and 2 are covered in this project and later projects. The two objectives are:

O.1 Given a conditional task, learners can identify the conditions and actions, and rewrite the task in ITE (If…Then…Else…) pattern. (Procedure)

O.1.1 Learners can identify the part of condition and the part of action in a conditional task description. (Concept)

O.1.2 Given conditions and actions, learners can compose a conditional task in ITE pattern. (Procedure)

O.2 Given an ITE task, learners can use a Switch block to complete the task. (Problem solving)

O.2.1 Learners can recognize the Switch block in common palette. (Discrimination)

O.2.2 In three types of condition, learners can determine the type of condition of a task. (Principle)

O.2.2.1 Learners can explain three types of condition (color, touch, ultrasonic) with example. (Concept)

O.2.3 Given a color condition, learners can use color sensor in Switch block and set correct parameters to identify black, blue, green, yellow, and red colors. (Procedure)

O.2.4 Given a touch condition, learners can use touch sensor in Switch block and set correct parameters to accept three touch actions. (Procedure)

O.2.4.1 Learners can define three touch actions (pressed, released, and bumped). (Concept)

O.2.5 Given an ultrasonic condition, learners can use ultrasonic sensor in Switch block and set correct parameters to detect the distance of an object. (Procedure)

Sequencing and grouping of objectives.

Table 14

Sequencing and grouping of objectives

Sequence No.	Objectives group
1	O.1, O.2.1
2	O.2.2, O.2.3, O.2.4, O.2.5, O.2

Introduction (20-25 minutes)

Activate attention (5 minutes). The instruction begins with a completed final lesson project. The project will be shown by a movie clip. A robo-car moves forward. When a connected color sensor detects a red color first, the robo-car says “Red” and stops;

when the color sensor detects a green color first, the robo-car says “Green” and continues to move forward. Learners are asked to describe what they have seen.

Establish purpose (3 minutes). Explain what were happened in the movie clip and gives the learning goal of this lesson - When given a task which will use *Switch* blocks, the learner will be able to design a program with correct blocks to complete the task, and run the program on a provided robot car.

Arouse interest and motivation (5 minutes). Show and explain a complex robotic system. An abstract scratch picture or diagram of a complex robotic system will be presented here. Learners are indicated that complex system is composed by many small pieces of single operation that uses sensors to detect condition and then performs actions.

Preview learning activity (5-10 minutes). Perform pre-test with a title, such as “Test yourself: How much do you know about this lesson”. The pre-test activity will give learner a small task, but need to use skills that will be learned in this module. For example,

> *“Design a program to perform following task. You will use a robo-car to finish the task.*
>
> *A robo-car will move forward. When the robo-car sees an object in 20 cm, it will say “Object” and stop.”*

This sample task has one single action (move forward), one condition (sees an object in 20cm), and two conditional actions (say “Object” and stop). Learners might be able to complete the single action, and stuck at the conditional tasks.

Body (35-45 minutes)

Recall relevant prior knowledge 1 (5 minutes). Show some action tasks to leaners. Each task action is a statement and looks like a command to be able to execute by the robo-car.

Action task examples:

- Move forward
- Stop
- Wait for 5 seconds
- Say “Green” and stop

Conclude the concept of an action task – One action or a group of direct actions to perform with action blocks.

Process information and examples 1 (5 minutes). Show some action tasks with one conditional clause in daily life and demonstrates to separate the conditions and actions. For example,

If it rains, close the windows, otherwise keep the windows open.

The condition is “it rains”; the actions are “close the windows” and “keep the windows open”.

Conclude the concept of a condition – the situation which something happens. A condition has two results. One is YES which means something happens, the other is NO which means something not happens.

Show some conditional tasks for a robo-car. For example,

- When a touch sensor’s button is pressed, the robo-car will say “Good job”.
- When a color sensor detects a red ball, the robo-car will say “Red”.

- When a touch sensor's button is pressed, the robo-car will stop moving.

Let learners identify the condition and action. Then learners are asked to fill following table.

Table 15

Condition and actions

Condition	Condition's status	Action
	Yes	
	No	

Provide a flow chart for the conditional task based on table 2. Learners are not required to draw the flow chart. Presenting the flow chart here is to analog the image on *Switch* block, so the learners can recognize the *Switch* block.

Figure 1 is a sample flow chart for the conditional task, "When a touch sensor's button is pressed, the robo-car will stop moving."

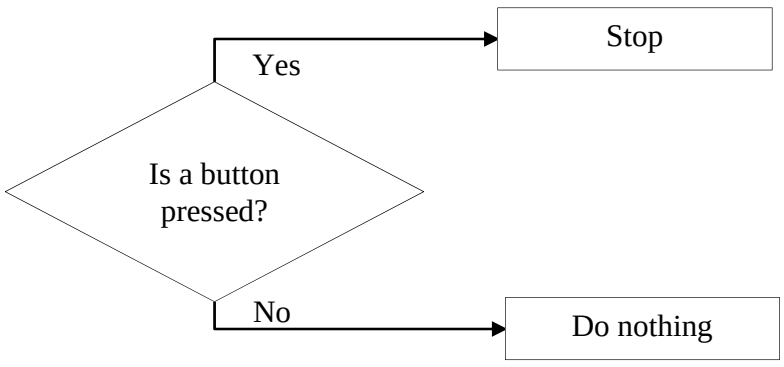


Figure 3 Sample flow chart

Focus attention 1 (5 minutes). Give a set of conditions and a set of actions which are suitable to be implemented in NXT, and ask learners to pair them based on the learners' knowledge. Following is an example for this activity.

Please draw lines between conditions, actions, and sensor types.

Condition	Action	Sensor Type
Hits an object	Says "Object"	Ultrasonic
Sees green line	Stops	Touch
Detect an object 30cm ahead	Says "Green"	Color

Employ Learning Strategies 1. To identify conditions and actions in a task description, learners need to understand the concepts of condition and action. 4th graders have already accumulated a lot of life experiences. Generate analogies here will help learners to understand the condition and action in this module. Therefore, the first conditional task is about daily life. In such a task, learners can identify the condition and the action more easily. Then, a conditional task with the same pattern as the daily life example presented to the learner as second example. With the help of the example of daily life, the learner can create the concepts of condition and action.

Generate an IF… THEN… ELSE… (ITE) pattern. This pattern is scaffold to decompose a conditional task. The learners are asked to put the true condition into the gap after "IF", put the action which performs when the condition is true into the gap after "THEN", and put the action which performs when the condition is false into the gap after "ELSE". Sometimes, there might be only one action in the pattern. By using the ITE

pattern, learners won't mess up conditions, actions, and the status of conditions to perform the actions.

Practice/ Evaluate Feedback 1 (5 minutes). There are four conditional tasks. Learners are asked to rewrite the tasks in ITE pattern. For examples,

1. When you hear a fire alarm, get out of the building.
2. When you feel cold, put on your coat.
3. When a robo-car touches an object, the robo-car says "object" and stops.
4. When a robo-car sees red light, it stops.

The answers for odd-numbered questions are provided.

1. **IF** you hear a fire alarm **THEN** you get out of the building **ELSE** do nothing

3. **IF** a robo-car touches an object **THEN** the robo-car says "object" and stops **ELSE** do nothing.

Recall relevant prior knowledge 2 (5 minutes). Ask learners to identify the sensors connected to a robo-car. The default connections are two touch sensors connected to port 1 and port 2, a color sensor connected to port 3, and an ultrasonic sensor connected to port 4.

There is a reference to different types of sensors. If learners cannot identify the sensor, they can go to the reference and review the prior knowledge about sensors.

Process information and examples / focus attention 2 (15-20 minutes). Use the 3rd question in practice 1 as an example. The task in ITE pattern is:

> **IF** a robo-car touches an object **THEN** the robo-car says "object" and stops **ELSE** do nothing.

Provide a movie clip to demonstrate the final result.

Provide a step-by-step tutorial to complete the conditional task.

Employ learning strategies 2. This is a procedure learning. A checklist (shown as table 3) will be provided for learners to perform the procedure as job aids.

Table 16

Steps to complete a conditional task

Steps	Tasks
1 ☐	Determine the type of condition. ☐ Color ☐ Touch ☐ Distance
2 ☐	Drag and drop a Switch block onto a new thread.
3 ☐	Select “sensor” for Control.
4 ☐	Select sensor which checked in step 1 for Sensor.
5 ☐	Select the port which is connected to the sensor in step 4. ☐ 1 ☐ 2 ☐ 3 ☐ 4
6 ☐	Select an action. The action is _____
7 ☐	Set a parameter if needed. The parameter is _____
8 ☐	Drag and drop action blocks onto upper branch to perform the action when the condition is true
9 ☐	Drag and drop action blocks onto lower branch to perform the action when the condition is false. (optional)
10 ☐	Download and run the program on a robo-car.

Learners will be given a step-by-step tutorial as a case study to familiar with the steps involved in the task. In practice, learners will be asked to go through the similar steps to complete a similar task.

Practice / evaluate feedback 2 (5-10 minutes). Give learner another conditional task to do.

“When a robo-car detects that there is an object 40cm away ahead, the robo-car says “object” and stops.”

Provide a movie clip to show what the robo-car acts.

Conclusion (5 minutes)

Summarize and review. Review objects. Remind leaners use IET pattern and checklist table when design a program.

Transfer learning. Ask learners what types of sensor they want to have other than the sensors used in this lesson. Do some research on Internet and find more stories about sensors.

Remotivation and close. Encourage learners design more conditional tasks. Give hints on further studies, e.g. use *Switch* block with a *Loop* block.

Assessment (20-25 minutes)

Assess learning. Conduct an online post-test. The post-test will have following structure.

Table 17

Post-test structure

Section	Question Type	Score
I	Multiple choices	30
II	Short answer	15
III	Programming	55

For section III, learners can use a computer and a robo-car to test their program.

Evaluate feedback and seek remediation. Provide self-evaluation guidelines. All multiple choices will be evaluated automatically. Answers for short answers will be provided. A movie clip of finished programing will be provided.

Lesson Delivery Strategy

This lesson is a self-instructional unit. A computer and LEGO MINDSTORM NXT 2.0 are required. The instructional materials include a paper based study manual, an online learning module, and several interactive activities. The pre-assessment and post-assessment will be delivered by both paper and computer. Movie clips will be posted on Internet and delivered by DVD as well. If learners don't have Internet access or have limited Internet access, they still can use paper based material and DVD to finish this unit.

Lesson Management Strategies

Allocations of time for each events of this unit are indicated after each event's title. A summary of time allocation is shown in table 5.

Table 18

Time allocation

No.	Event	Time (minutes)
1	Introduction	20-25
2	Body 1	20
3	Body 2	25-35
4	Conclusion	5
5	Assessment	20-25
	Total	90-110

Learning space is not specified for this lesson. Learners can complete the lesson at any places where has a computer with NXT software. No instructor is involved in this lesson. All activities will be complete by learners. Self-evaluation guidelines are provided for self-check.

Project Summary

In this project, a full map of instructional strategies is presented. The project uses organizational strategies to chunk the objectives into two groups and presented in a series of instructional events, uses printed materials plus online video tutorial as delivery media, and uses detailed time schedule to help manage the instruction. The final instructional product will be based on this project, but small updates could be made during the producing phase.

Project F: Instructional Materials Development

Project F is a standalone package that was developed based on the content from project A to E. Please see Appendix A for the project F.

References

Indiana Department of Education (2009). Indiana Standards Retrieved 9/11, 2012, from https://learningconnection.doe.in.gov/Standards/Standards.aspx?st=&sub=5&gl=5&c=0&stid=0

Indiana Department of Education. (2009). Standards for English/Language Arts Retrieved 9/18, 2012, from https://learningconnection.doe.in.gov/Standards/Standards.aspx?st=&sub=2&gl=6&c=0&stid=0

Inhelder, B., & Piaget, J. (1958). *The growth of logical thinking*. New York: Basic Books.

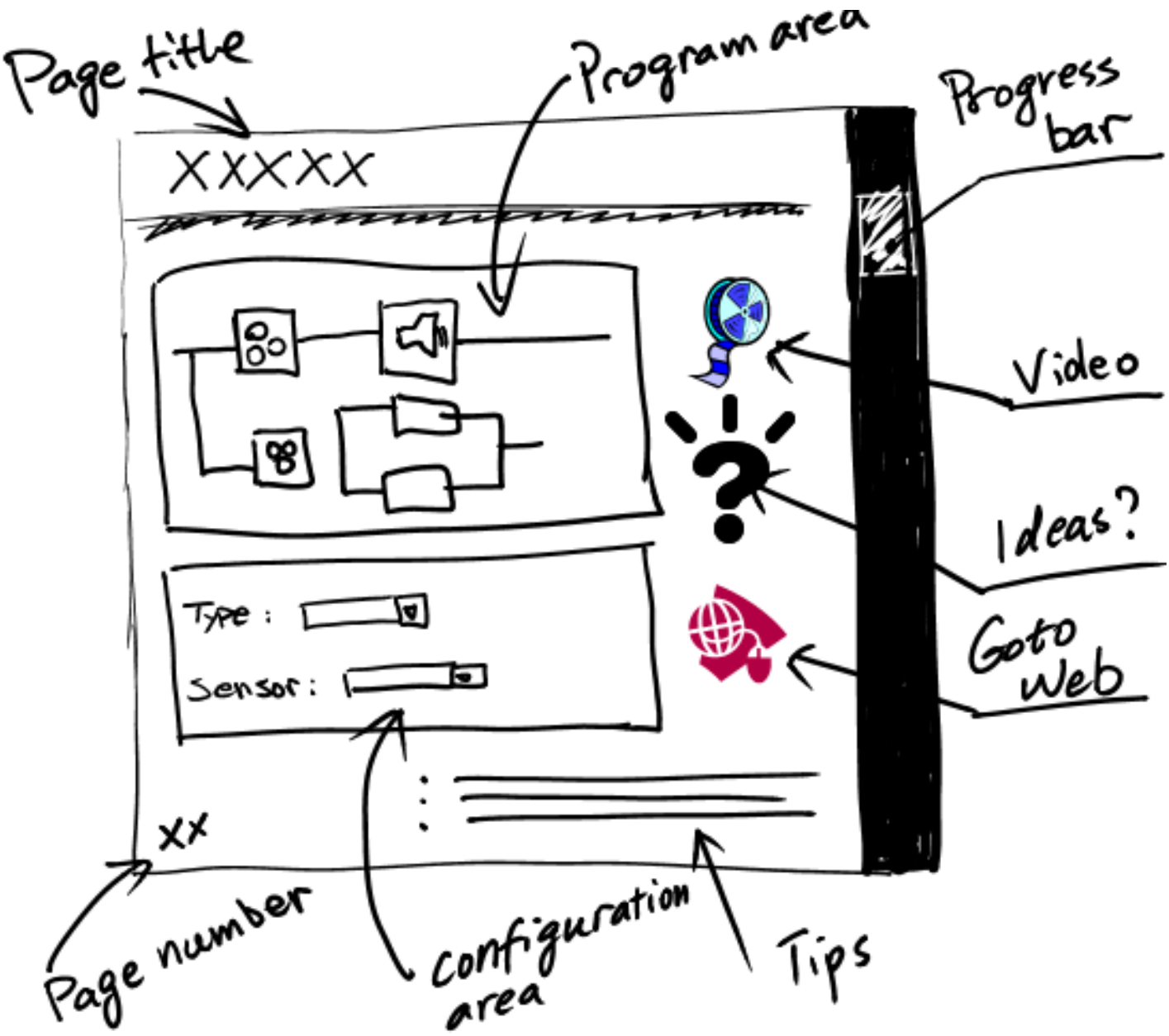
Smith, P. L., & Ragan, T. J. (2005). *Instructional design* (3rd ed.). Hoboken, NJ: John Wiley & Sons.

Appendix A. Project F: Instructional Materials Development

How to Make Your Robot

A Sample Self-instructional Project for CIMT 610

COURSE & COPYRIGHT INFORMATION

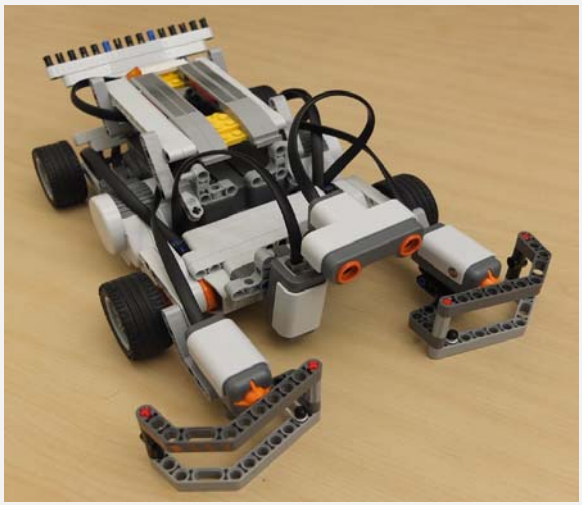

Page title
XXXXX
Program area
Progress bar
Video
Ideas?
Goto Web
Type :
Sensor :
xx
Page number
configuration area
Tips

WHAT'S IN THIS BOOKLET

WHAT DO YOU NEED FOR THIS TUTORIAL?

LEGO MINDSTORM NXT 2.0 (8547)

A COMPUTER WITH NXT 2.0 SOFTWARE



A ROBO-CAR (CUSTOMIZED OR USE SHOOTERBOT*)

- ☐ I CAN CONNECT NXT TO COMPUTER
- ☐ I CAN CREATE A NEW PROGRAM
- ☐ I CAN IDENTIFY SENSORS
- ☐ I CAN IDENTIFY PORTS
- ☐ I CAN USE ACTION BLOCKS (MOVE, WAIT, SOUND, DISPLAY)
- ☐ I CAN DOWNLOAD AND RUN PROGRAMS ON NXT

ALL SKILLS YOU NEED TO CONTINUE

* NXT has a tutorial for assembling your first bot. You can find it in HELP menu.

BLANK PAGE

1. WHAT CAN A ROBO-CAR DO?

http://www.youtube.com/watch?v=123

In this example, we use a color sensor to detect the color of lines. If it detects green line, the car will say "Green" and continue moving. If it detects a red line, the car will say "Red" and stop at the redline. (You can view the video clip at http://www.youtube.com/watch?v=123.)

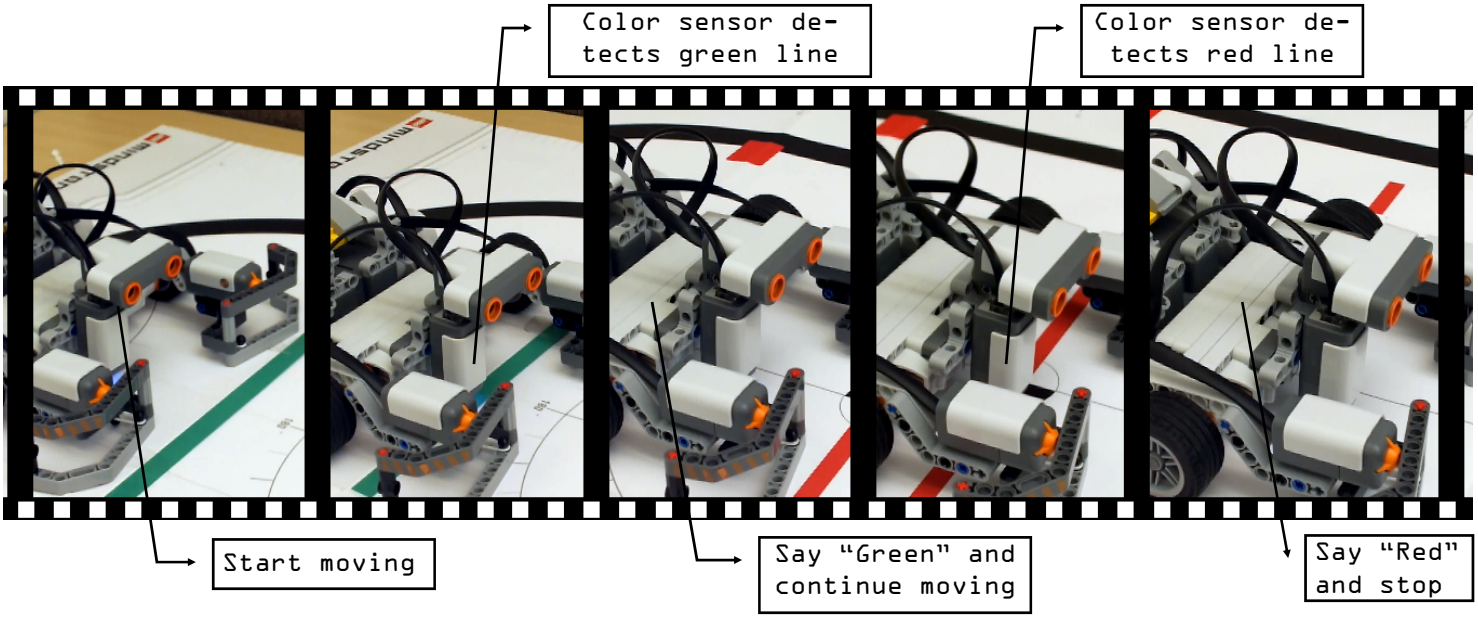


Figure 1 Color detector

Think about what does the robo-car do in the example?

Yes, it makes **decisions**.

Thank about what tool does the robo-car use to detect a condition in this example?

The answer is **sensor**. In this example, it uses a **color sensor** to detect different colors.

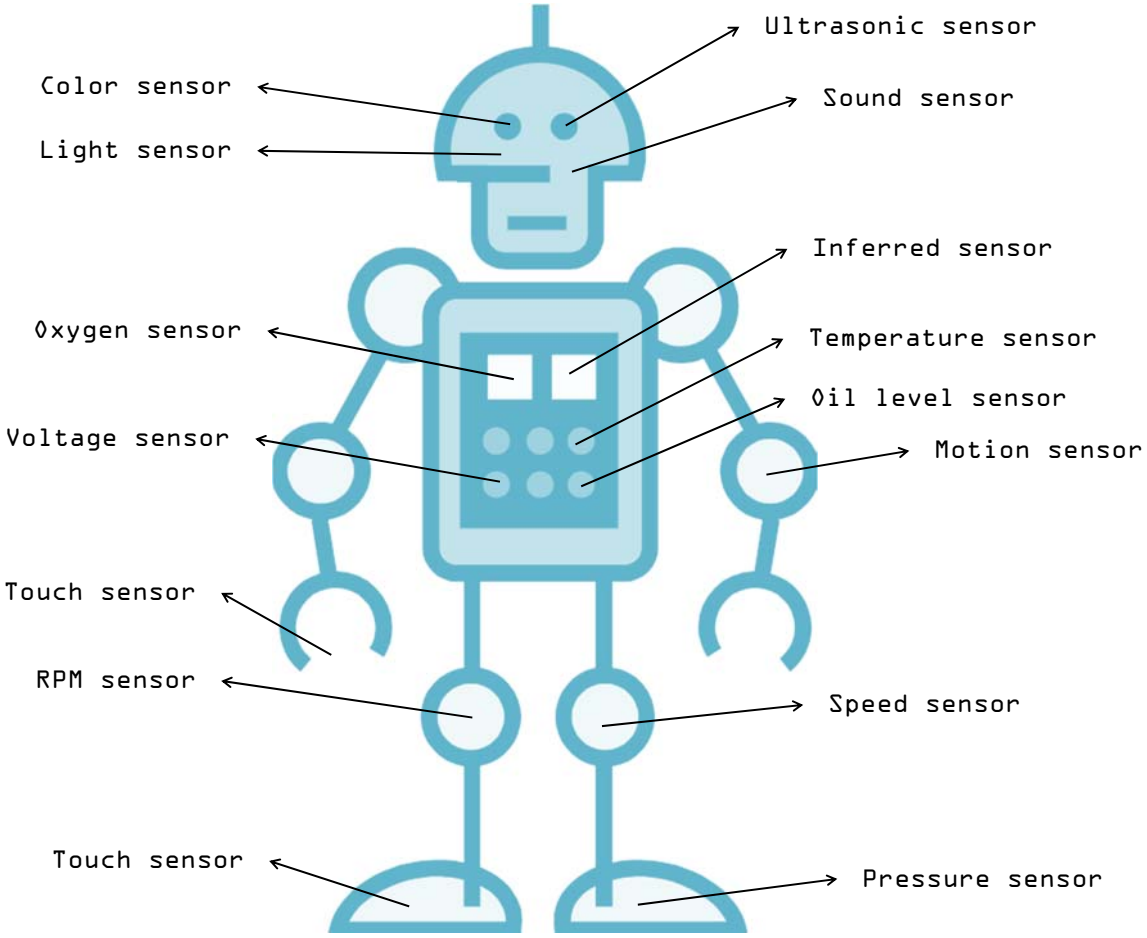


Figure 2 A Robot

This robot uses thousands of sensors to detect different conditions and then make his decision. Sensors let robots "know" the outer world, and let them be smarter. In this tutorial, you will learn how to use **SWITCH** blocks and sensors in LEGO MINDSTORM NXT 2.0.

Activity 1:

In this activity, you will identify the three types of sensors that will be used in this tutorial.

Please draw lines between pictures and names.

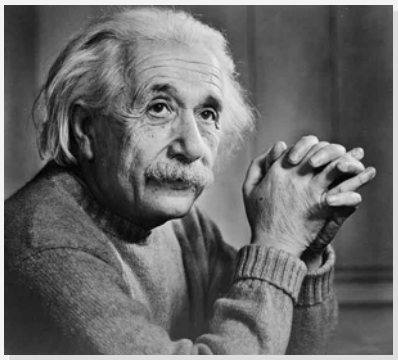
Color Sensor

Ultrasonic Sensor

Touch Sensor

If you are not sure about the type of sensors, please check the NXT manual.

Answer: 1. Ultrasonic sensor; 2. Touch sensor; 3. Color sensor



“If I had 60 minutes to solve a problem, I’d spend 55 minutes defining it, and 5 minutes solving it.”

—— Albert Einstein

This quote illustrates a very important point that before you solve a problem invest your time to understand the problem. In this tutorial, you will solve problems of conditional task. Sometimes, these problems are not well formatted, or with reluctant information. Therefore, the first step to solve these problems is to define problems in effective language structure, or present problems using a visual map.

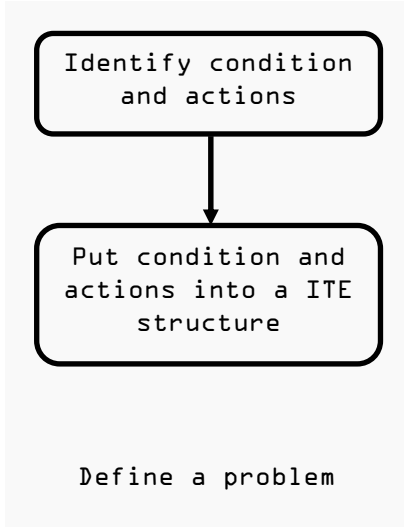


Figure 3 Define problem

Activity 2:

Action tasks can be performed using action blocks.

Following are 7 tasks which a robo-car can do. Please check out the action tasks.

- ☐ 1. Move forward
- ☐ 2. Stop
- ☐ 3. Wait for 5 seconds
- ☐ 4. Say "Green" and keep moving
- ☐ 5. Turn left
- ☐ 6. See a red ball
- ☐ 7. See an object 20 cm ahead

Answer: 1, 2, 3, 4, 5

Condition and Action

The last two tasks in activity 2 are conditions, not action tasks. Conditions are the situations when something happens.

Activity 3

Look at this statement:

> *If it rains, close the windows, otherwise keep the windows open.*

There are condition and actions in this task. Can you find them?

Condition	Action

Answer: Condition - if it rains
Action - close the windows & keep the windows open

YES or NO?

A condition has two results, Yes or No. The following image is a visual map for the raining problem.

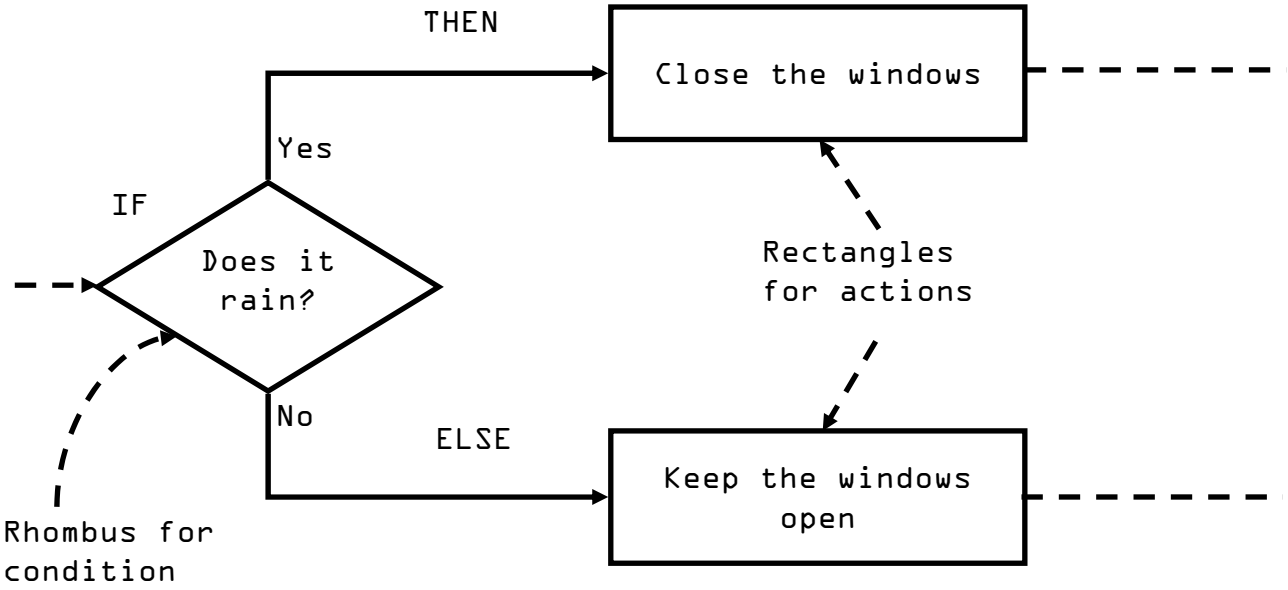


Figure 4 ITE pattern

You can use IF...THEN...ELSE… (ITE) structure to define the problem. Put the condition after IF, put the action when the condition is true after THEN, put the action when the condition is false after ELSE. Make the problem looks like:

IF it rains,

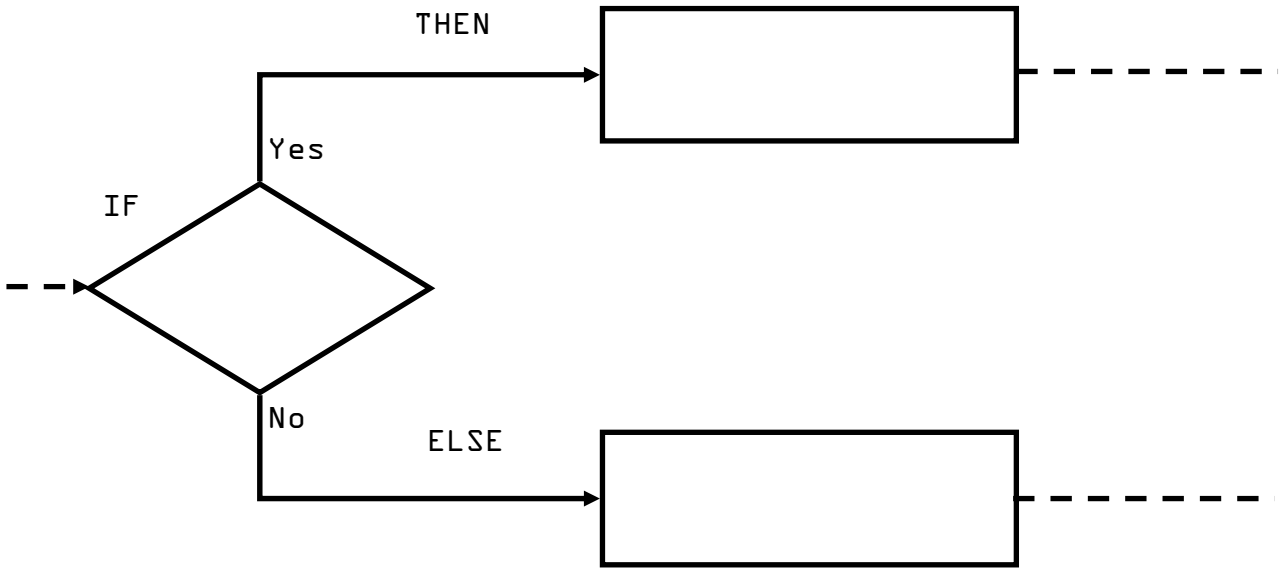
THEN close the windows,

ELSE keep the windows open.

Activity 4

Define the following problem:

> When a robo-car detects red color, it will say "Red" and stop.



IF _____, THEN _____,

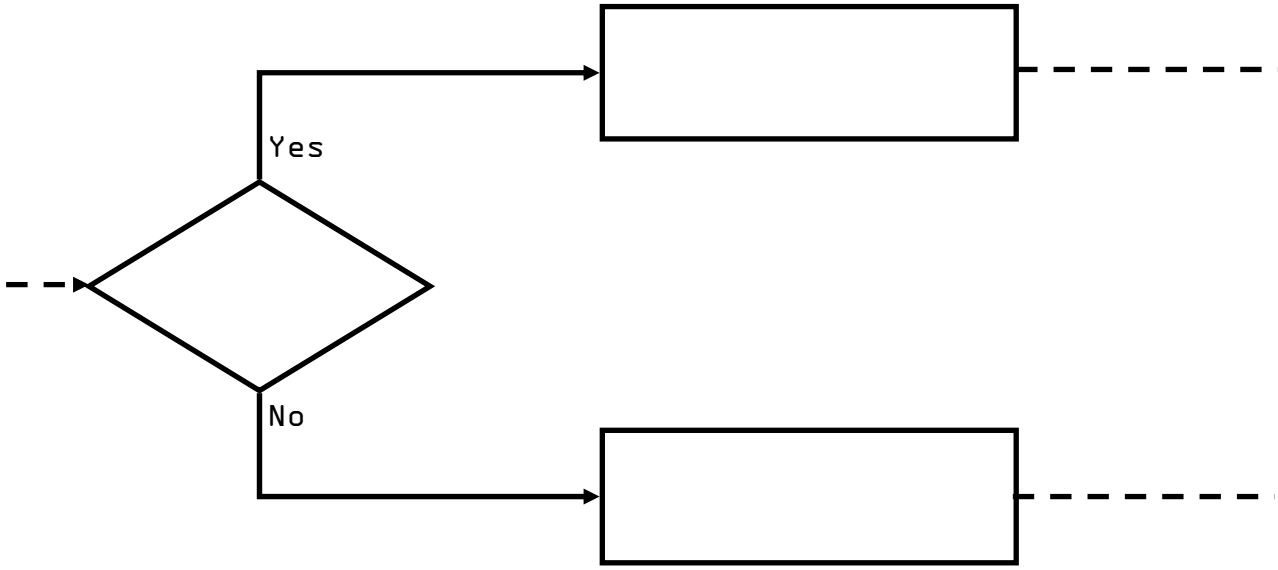
ELSE _____.

Answer: IF detect red color , THEN say "Red" and stop ,
ELSE do nothing .

Activity 5

Define the following problem:

> When a robo-car touches an object, it will say "Object" and stop.



IF _____, THEN _____,

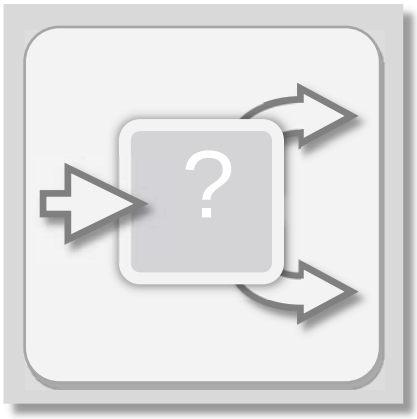
ELSE _____.

Answer: IF touch an object , THEN say "Object" and stop ,

ELSE do nothing .

A switch block is used to make decisions.

A switch block can use time, variable, and sensor as conditions. It has two branch, one is for actions when the condition is YES, and the other is for actions when the conditions is NO.



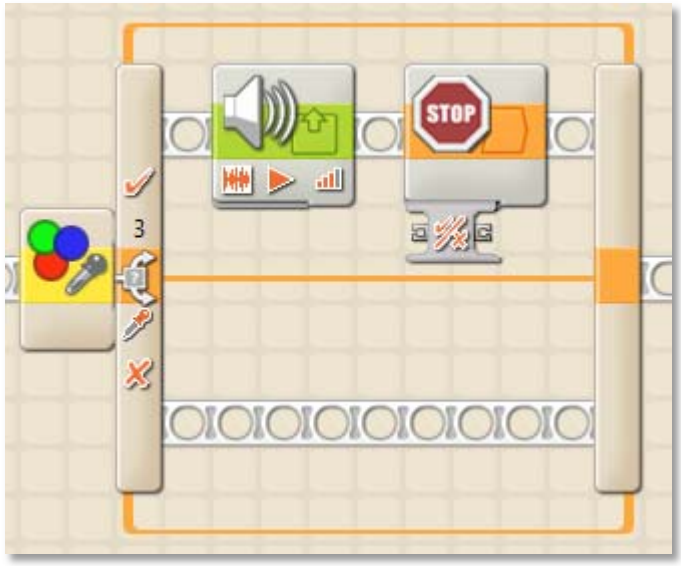


Figure 4 Switch block

The left icon is the switch block icon on toolbox; and the right image is a switch block in a program. Like the pattern you used in previous section, a switch block has a condition block and two branches. The upper branch is used when condition is YES, and the lower branch is used when condition is NO.

In the whole tutorial, you will use three types of sensors as conditioner, color sensor, touch sensor, and ultrasonic sensor. The next sample will use a touch sensor.

3. THE SWITCH BLOCK

Now, here is your problem:

> **IF** the button on left touch sensor is pressed, **THEN** say "object" and stop, **ELSE** do nothing.

Before you jump into the program, you have to do another thing, identify the port number of sensors.

A robo-car has one ultrasonic sensor, one color sensor, two touch sensors, and two motors. Sensors are connected to the number ports (figure 5), and motors are connected to the alphabet ports (figure 6).

Figure 5 Port 1, 2, 3, and 4

Figure 6 Port A, B, and C

Record your connection in following table:

Table 1 Sensors and ports

Sensor/Motor	Left Touch	Right Touch	Color	Ultrasonic	Left Motor	Right Motor
Default Port No.	1	2	3	4	A	B
Your Port No.						

You will use these port numbers later.

Suppose you have your left touch sensor connected on port 1. Go through following steps to make the robo-car say "Object" when the button on your left touch sensor is pressed.

Step 1. Drag and drop a Loop block to the default sequence.

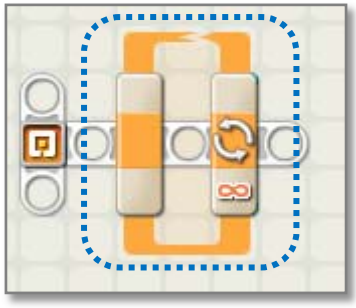


Figure 7 Loop block

Step 2. Drag and drop a Switch block into the Loop block.

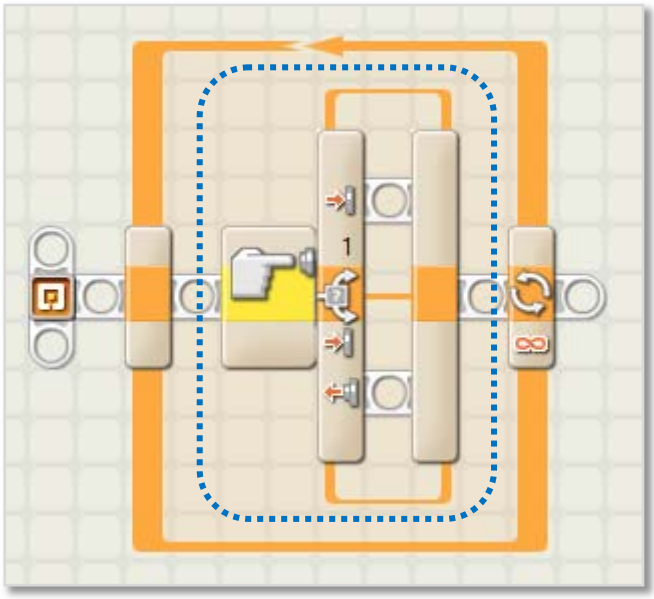


Figure 8 Switch block

Step 3. Click the Switch block. Choose "Sensor" in the Control filed (options will change depends on what you choose here). Then, choose "Touch Sensor" in the Sensor field.

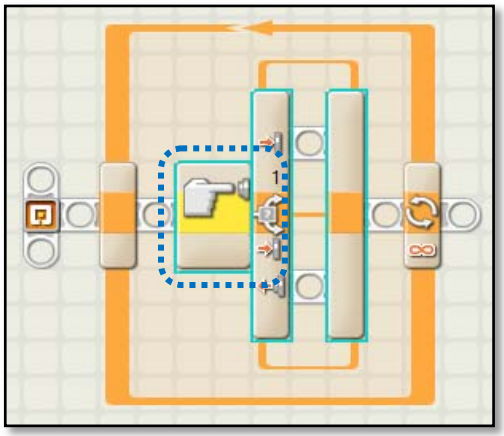


Figure 9 Touch sensor

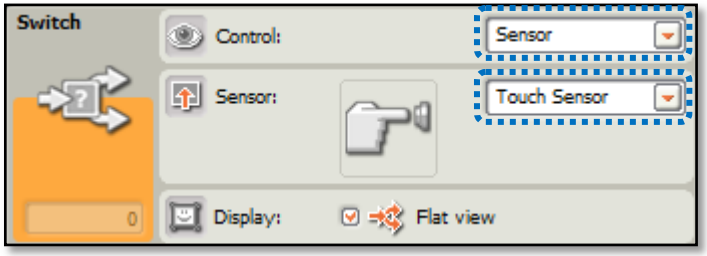


Figure 10 Touch sensor parameters 1

Step 4. In Port field, check the port number which is connected to the left touch sensor. By default, it will use port 1. Then, check "Pressed" in Action field.

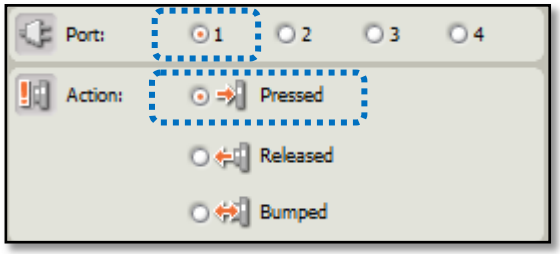


Figure 11 Touch sensor parameters 2

Step 5. Put a Sound block on the upper branch of the switch block and choose the "Object" sound file (When you click the sound file, the robo-car will play the sound file for testing). Then, put a Stop block after it. These actions will perform when the condition is YES, which means the button of the left touch sensor was pressed.

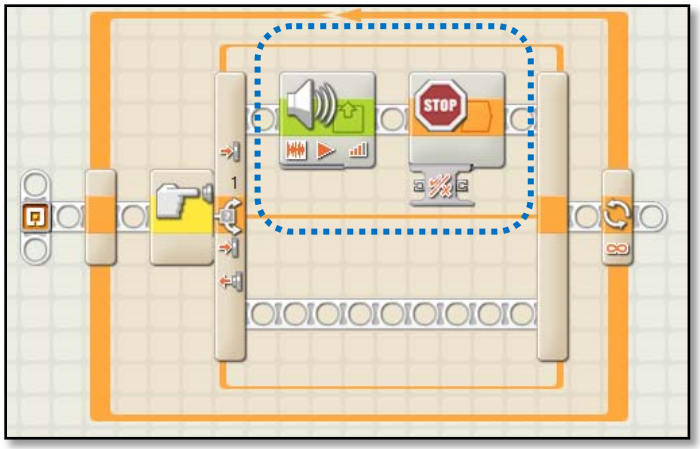


Figure 12 Sound block

Figure 13 Sound block parameters

The lower branch of the switch block is for actions when the condition is not happened. In this sample, keep it empty, because there are no actions for it.

Step 6. Save the program as "touchleft.nxt". Download and run the program on the robo-car.

Step 7. Testing. Press the button on left touch sensor. Do you hear the robo-car say "Object"? If not, go back and check every steps.

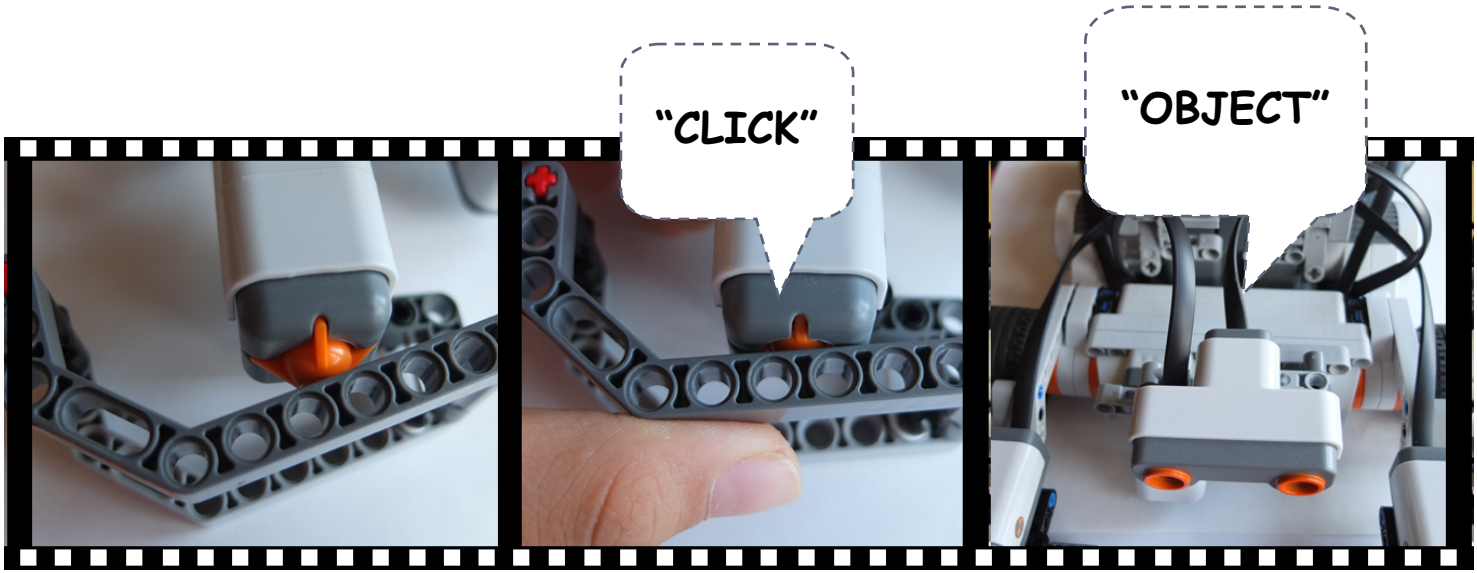


Figure 14 Touch demo

http://www.youtube.com/watch?v=123

Conclusion

In this section, you have been presented a procedure for a conditional task.

1. Define the problem in ITE pattern. (You did it in activity 5)
2. Add a Loop block to keep the Switch block working repeatedly. (step 1)
3. Add a Switch block. (step 2)
4. Set Condition type to "Sensor"; select "Touch Sensor" in sensor type field. (step 3)
5. Set parameters for touch sensor. (step 4 and 5)
6. Save and test the program. (step 6 and 7)

Activity 6

Create a new program named "touchright.nxt" and finished the task: **IF** the button on **right** touch sensor is pressed, **THEN** say "object" and stop, **ELSE** do nothing. (Use the testing method in step 7 to test your program.

http://www.youtube.com/watch?v=123

4. MAKE IT HAPPEN

In a real situation, you might use multiple sequences to finish your task. For example, a robo-car might need to detect colors or objects when it is moving, like the sample you saw at the beginning of this tutorial.

Activity 7

Redefine the problem of the sample at the beginning of this tutorial:

A robo-car is moving forward.

IF the robo-car _____, **THEN** _____

_____, **ELSE** _____.

IF the robo-car _____, **THEN** _____

_____, **ELSE** _____.

Answer: A robo-car is moving forward.

IF the robo-car detects a green line, THEN play sound "Green", ELSE do nothing.

IF the robo-car detects a red line, THEN play sound "Red" and stop, ELSE do nothing.

In this problem, which sensor you will use? which port does it connected?

Go through following steps to make the robo-car detect color lines.

Step 1. Drag and drop a Move block on the default sequence.

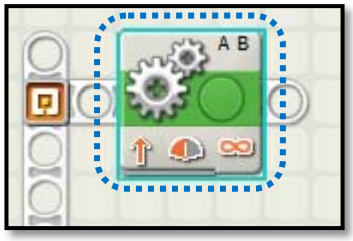


Figure 15 Move block

Step 2. Select the Move block. Check **port A** and **B** for left and right motors (according to table 1 on page 11). The direction is **UP**; keep steering in middle (means no steering); set power to **50%**, so the car won't move too fast; set duration to **Unlimited**, so the car will keep moving until a Stop action performed.

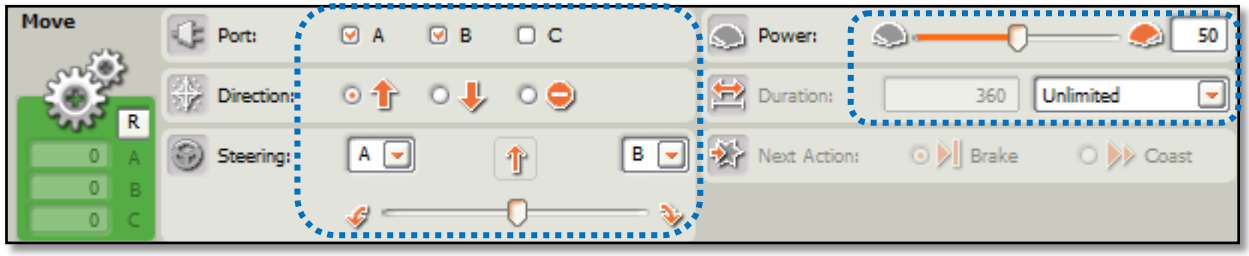


Figure 16 Move block parameters

Step 3. Drag and drop a Loop block below the Move block (figure 17).

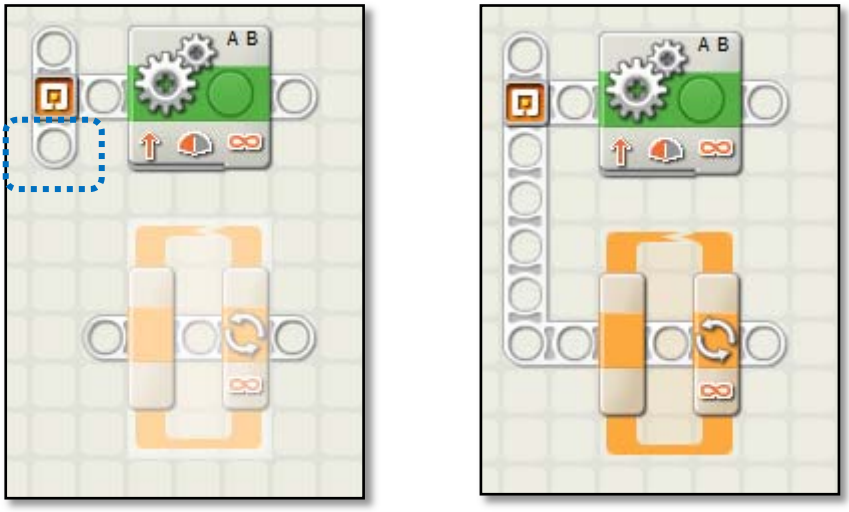


Figure 17 Parallel sequences

Step 4. Move your mouse over the starting point, the mouse point will change to a wire connector. Press left button and hold it, then move the sequence connector over the sequence of the Loop block. The two sequences will connect automatically (figure 17).

Parallel sequences will execute at the same time. You can create a parallel sequence from start point or in the middle of a sequence. If you would like know more, you can use "Content and Index…" in "Help" menu.

Step 5. Drag and drop a Switch into the Loop block (figure 18).

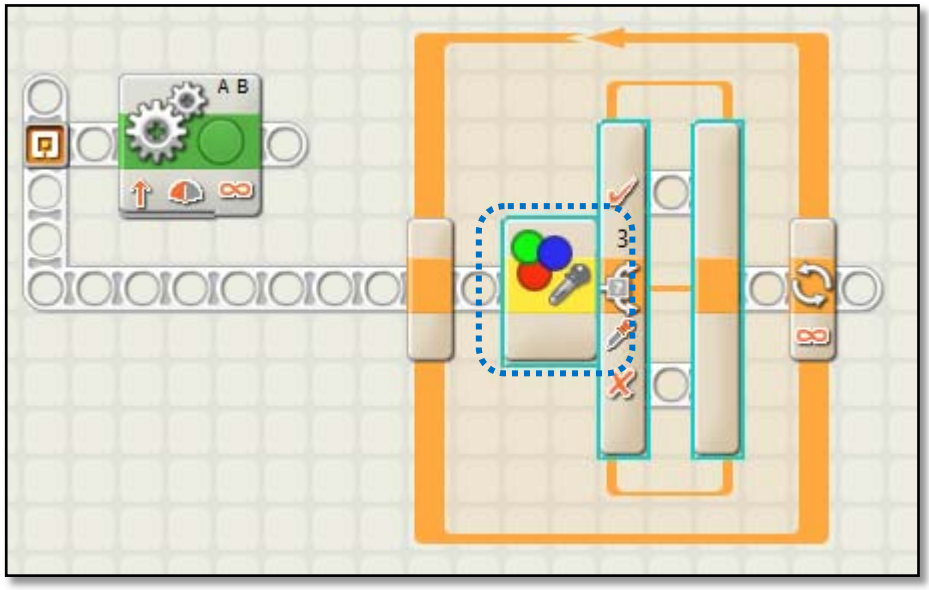


Figure 18 Switch block

Step 6. Select the Switch block. Then, choose "**Sensor**" in Control field, choose "**Color Sensor**" in Sensor field, check port 3 (or the port that connect your color sensor in table 1 on page 11), choose "**Color Sensor**" in Action field, choose "**Inside Range**" in Compare field, move the slide bar to select the **green** color (figure 19).

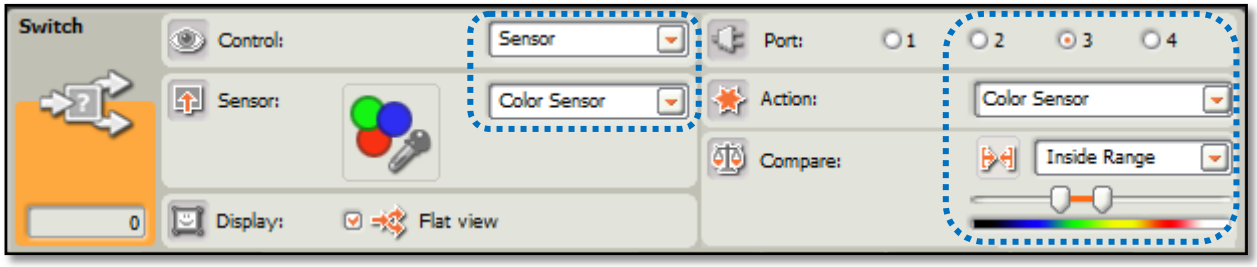


Figure 19 Color sensor parameters

Step 7. Put a Sound block (figure 20) on the upper branch of the switch block and choose the "Green" sound file (figure 21).

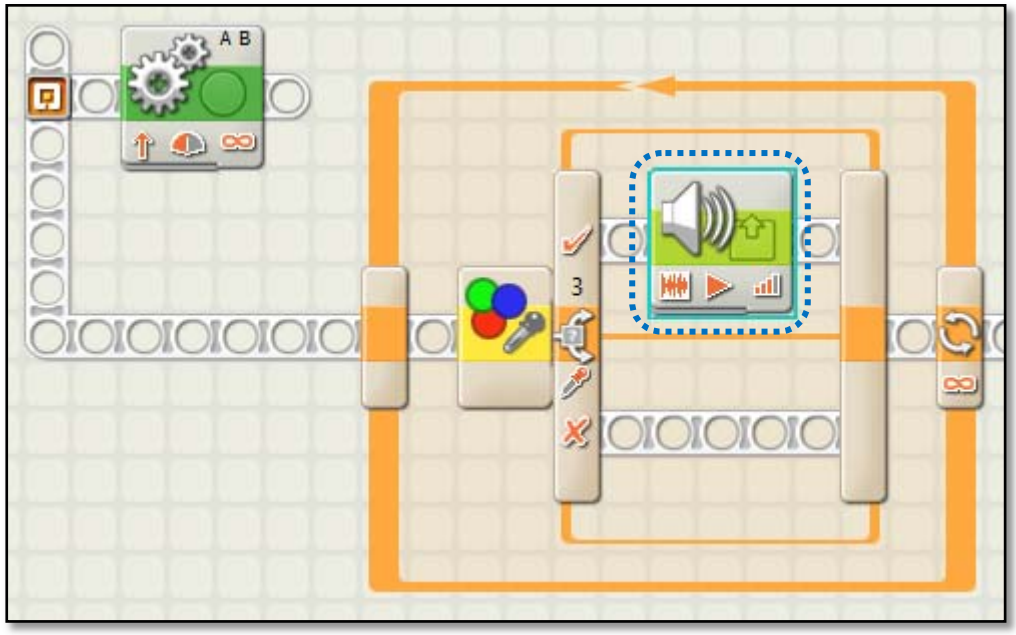


Figure 20 Sound block

Figure 21 Sound block parameters

Step 8. Drag and drop another Switch into the Loop block (figure 22).

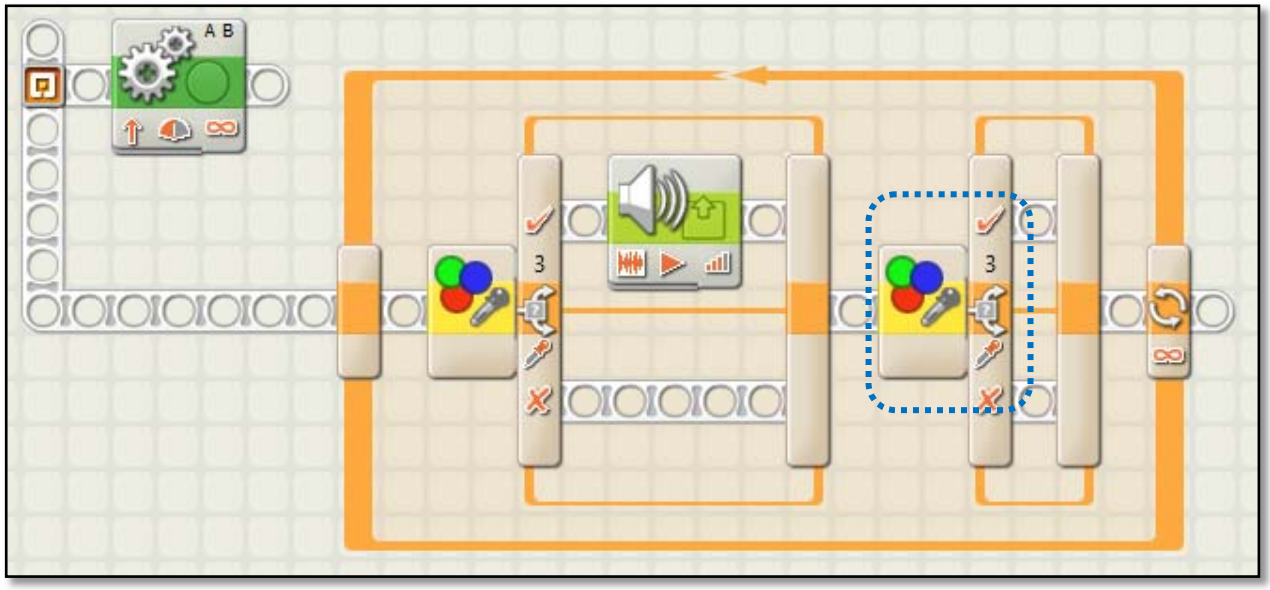


Figure 22 Second switch block

Step 9. Select the Switch block. Then, choose "**Sensor**" in Control field, choose "**Color Sensor**" in Sensor field, check port 3 (or the port that connect your color sensor in table 1 on page 11), choose "**Color Sensor**" in Action field, choose "**Inside Range**" in Compare field, move the slide bar to select the **red** color (figure 23).

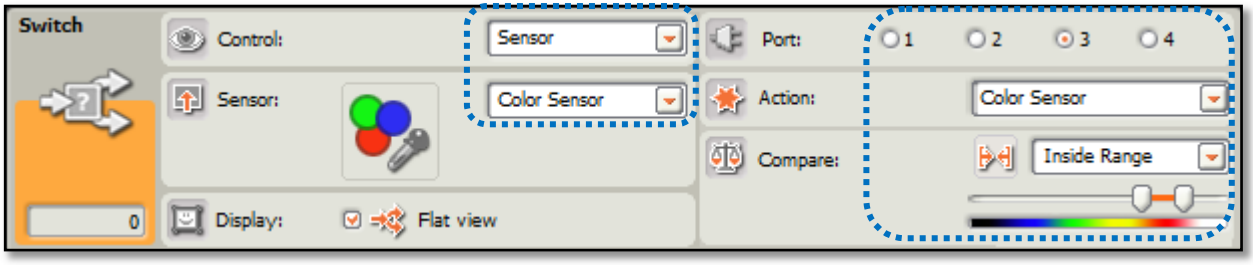


Figure 23 Second color sensor parameters

Step 10. Put a Sound block (figure 24) on the upper branch of the switch block and choose the "Object" sound file (figure 25) . Then, put a Stop block after it (figure 24) .

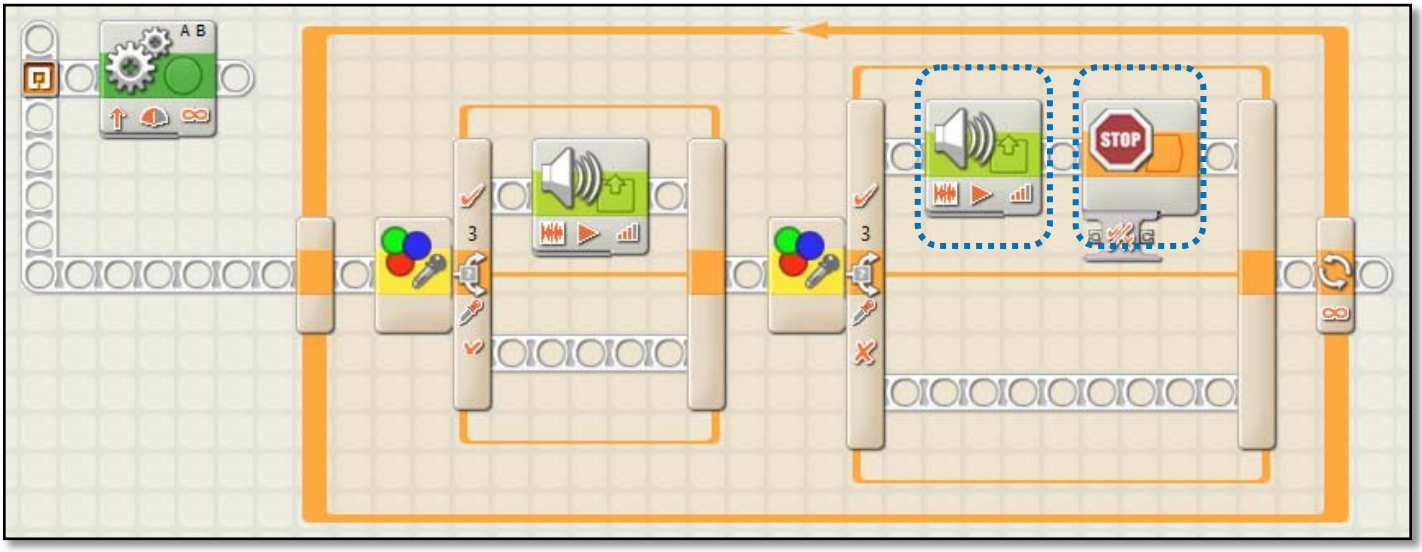


Figure 24 Stop block

Figure 25 Second sound block paramters

Step 11. Save the program as "touchleft.nxt". Download and run the program on the robo-car.

Step 12. Testing. Put the robo-car on a test pad which has a green line and a red line. You can make a green line on the standard test pad paralleled with the red line. First round, let the robo-car detect green line first; second round, let the robo-car detect the red line first. Please review the video file online.

http://www.youtube.com/watch?v=123

Remember what we just did?

Table 2 Common steps for a conditional task

Steps	Tasks
1	Determine the type of condition.
2	Drag and drop a Switch block onto a new thread.
3	Select "sensor" for Control.
4	Select the sensor type determined in step 1 for Sensor.
5	Select the port which is connected to the sensor in step 4.
6	Drag and drop action blocks onto upper branch to perform the action when the condition is true
7	Drag and drop action blocks onto lower branch to perform the action when the condition is false.
8	Download and run the program on a robo-car.

The steps in table 2 are the common steps to solve a conditional task problem. Some steps might need to be adjusted.

Activity 8

Now, try a task by yourself. When you finished, check the online video.

A robo-car goes forward. When the robo-car detects that there is an object 20cm ahead, it says "Object detected" and stops.

http://www.youtube.com/watch?v=123

CONCLUSION

In this tutorial, you have learned to:

- Identify conditions and actions in a task
- Define a conditional task with ITE structure
- Determine a condition's type and the sensor's type
- Recognize the Switch block in common palette
- Use parallel sequences
- Solve conditional task problem using Switch block

Time: 35 minutes

Name: _____ Date: _____

Section #	Section I	Section I	Section III	Total
Score				

Section I: Multiple Choices

Direction: In this section, you will choose the best answer for each question.

1.( ) Which one of the following tasks is a conditional task?

A. Display a picture on screen.

B. When sees a red ball, says red.

C. Play a sound.

D. Go forward, don't stop.

2.( ) Which one of the following blocks is Switch block?

A. 

B. 

C. 

D. 

CHECK YOUR UNDERSTANDING

3.( ) Which one of the following conditions is a touch condition?

A. When the robo-car sees a red line, it stops.

B. When the robo-car sees an object, it stops.

C. When the robo-car conflicts with an object, it stops.

D. When the robo-car runs 2 minutes, it stops.

4.( ) Which one of the following actions needs a Loop block to repeat?

A. Play a sound forever

B. Play a sound for 50 times.

C. When a button pressed, stop the car.

D. Go forward, never stop.

5.( ) Which one of the following blocks is a Loop block?

A. 

B. 

C. 

D. 

6. ( ) Which one of the following loops will use a touch sensor to stop?

A. A robo-car will play a sound for 50 times.

B. A robo-car will display a picture for 5 minutes.

C. A robo-car will always say "Green" when it sees a green ball until it sees a red ball.

D. A robo-car will shoot balls until a touch sensor button pressed.

Section II: Short answer

Direction: In this section, you will answer the question using one sentence.

7. Use the following condition-action pair to compose an ITE sentence:
Condition: touches an object; Action: says "Object".

_____

8. Rewrite the following task in to an ITE sentence:
A robo-car will move forward. When the robo-car sees any object in 20cm, it will say "Object" and stop.

_____

_____

Section III: Operation

Direction: In this section, you will be asked to design a program to complete a task. You can use NXT software and run the program on a robo-car.

9. Design a program to perform following task. You will use a robo-car to finish the task.

A robo-car goes forward. When the left touch sensor touched an object, it will backward for 5cm, and turn 45 degree to right, then move forward. When the right touch sensor touched an object, it will backward for 5 cm, and turn 45 degree to left, then move forward.

Table 3 Answer sheet

Q#	Grade Guidelines	Available Points	Your Points
1	B	5	
2	C	5	
3	C	5	
4	B	5	
5	B	5	
6	D	5	
7	**If** car touches an object (2.5), **then** it will say "object". (2.5)	5	
8	**If** car sees an object in 20cm (3.5), **then** it will say "object" and stop (3.5), **else** it will continue to move forward. (3)	10	
9	Give a self-evaluation based on the video at http://www.youtube.com/watch?v=123	55	

REFERENCES

Valk, L. (2010). The LEGO MINDSTORM NXT 2.0 discovery book - a beginner's guide to building and programming robots. San Francisco, CA: No Starch Press.