ROBOLAB Reference Guide

Version 1.3

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Preface: Getting Help with ROBOLAB

ROBOLAB is a growing application for which users can receive support in many different ways. Embedded in ROBOLAB are context help and extended help information and examples. Three Teacher's Guides have also been written that focus on teaching readers how to use different components of the software. The web also serves as a means of finding information, new capabilities, suggested ways to use the software, classroom curriculum, and additional tutorials on using ROBOLAB. The web component is a continually growing environment where we update and add new material and sample curriculum all the time.

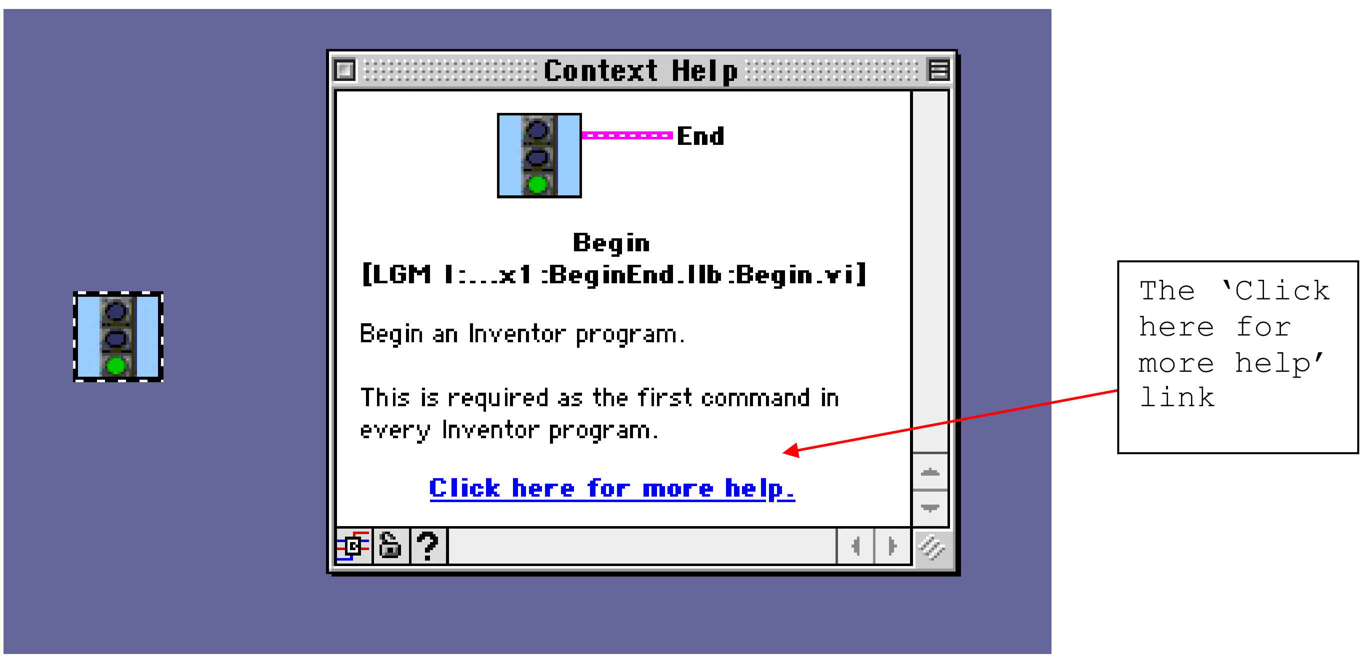
This guide is designed to serve as an extended reference book for the ROBOLAB software. It aims to supplement the other support and help components that already exist. With nearly 300 pages of information, the ROBOLAB Reference Guide is best used as an interactive electronic document from which pages or sections can be printed if needed. The Guide is a dynamic project, continually growing and being updated. The most current version is made available at www.ceeo.tufts.edu/robolabatceeo for download.

Help in ROBOLAB

The first place to look for help is within ROBOLAB. All versions of ROBOLAB have context help for the software environment. Help can be turned on and off using the Help Button or via keyboard shortcuts (CTRL-H)

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

After turning on the help, placing your mouse over any object in a window or icon the context help will present a description of the item or directions on how to use it. This type of help can be useful for learning how to use the Administrator, Pilot, or Investigator sections of ROBOLAB.



When programming in Inventor or higher levels of Investigator, context help is also available for each of the icons. This help that comes up describes the

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

functionality of the command, default settings, and the elements of it that can be modified. In ROBOLAB 2.5 and higher, the 'Click here for more help' link appears in the Context Help window. This brings up extended help information and a sample program that uses the command (that can be opened and downloaded). The extended help allows the user to see examples and is fully described in the Getting Started Guide #3.

The ***Using ROBOLAB*** guide

For a more detailed introduction to using the software, step by step directions, and more examples and challenges, the ***Using ROBOLAB*** guide is very useful. This book is available through your LEGO Education distributor or in electronic (pdf) format at the website. If you would like to find a distributor, visit www.lego.com/education and then click on Where to Buy.

Topics in the User's Manual Include

The Pilot and Inventor levels of programming in the PROGRAMMER component of ROBOLAB. This is all one needs to program a LEGO robot.

The INVESTIGATOR component that is used for data logging. This section adds scientific investigation to the robot's capabilities, allowing the RCX to become a smart and mobile data logging tool. The Investigator component also includes a Journal Area for documenting a project in text and photographs or illustrations and a Publish feature for sharing results.

The enhanced Media features: the Piano Player and Camera available in ROBOLAB 2.5.1. It also shows how to use some of the higher level capabilities including Vision Center and subroutines.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

ROBOLAB Web Sites

ROBOLAB has a growing web based community that has information, ideas, and examples of users in action. Two web sites that have learning resources, developer information and SDKs, new downloads, more documentation and support are:

http://www.lego.com/eng/education/mindstorms
(under support)

and

http://www.ceeo.tufts.edu/robolabatceeo

These sites are a great place to visit for up to the minute patches, new features, additions, and tutorials. Both sites also maintain a large number of links to curriculum resources, activity databases, and information on classroom materials. The Lego Users Group Network (LUGNET) at www.lugnet.com (not officially supported by LEGO) also provides a great resource for discussing ROBOLAB and other LEGO issues and locating additional information.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

TABLE OF CONTENTS

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

1. About ROBOLAB

ROBOLAB is the software that provides the tools and environment for programming, creating, learning and exploring with the LEGO RCX. With an intuitive, graphical interface, students of all ages are able to create autonomous robotic creations, collect data, compose music, and snap pictures.

The RCX – The Programmable LEGO Brick

Low Entry --High Ceiling

LEGO elements can be a powerful teaching tool for students of all ages. Using LEGO blocks a kindergarten student can explore the concept of numbers and sorting. Add a few motors and sensors to those LEGO blocks and a college student can learn about engineering and physics by building a robot or a set-up to measure spring constants. Having students from 5 - 25 use the same toolset to learn allows more time to be focused on learning and exploring and less on learning how to use software.

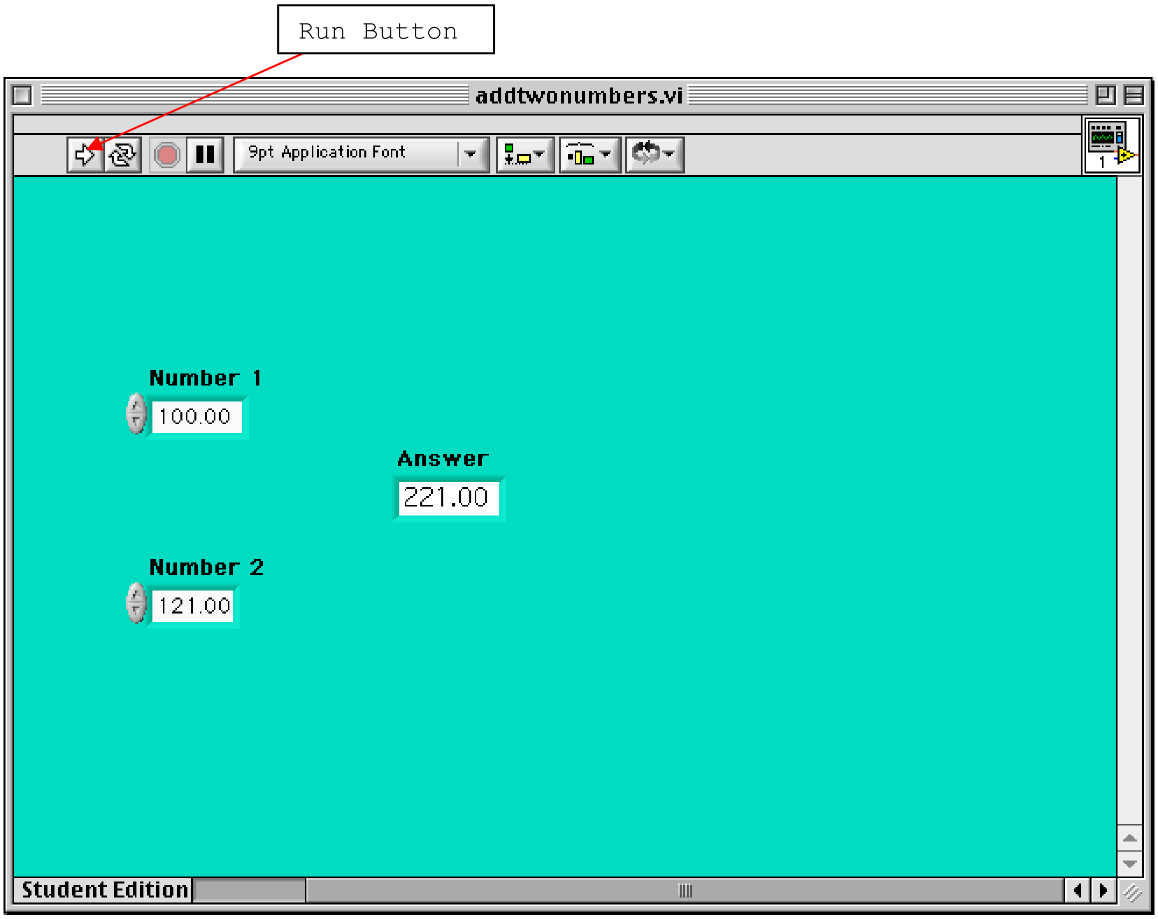
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

ROBOLAB and LabVIEW

ROBOLAB is built on top of a robust piece of software called LabVIEW. LabVIEW is a powerful programming environment used by engineers and scientists in colleges and industry. It is the leading software development tool for measurement and control. Created by National Instruments (Texas, US) in 1997, LabVIEW is used to analyze and compute real results for biomedical, aerospace, energy research applications and numerous other applications. NASA, for example, monitored the Mars Sojourner Rover's location and position in relation to the landing craft, its orientation to the ground, its overall physical health and more using LabVIEW.

LabVIEW is a graphical programming development environment that allows scientists and engineers to write programs and create user interfaces. A basic LabVIEW program, called a virtual instrument (vi) has 2 components to it - a front panel and a diagram. The front panel is where standard user interface options like buttons, dials, and graphs are displayed.

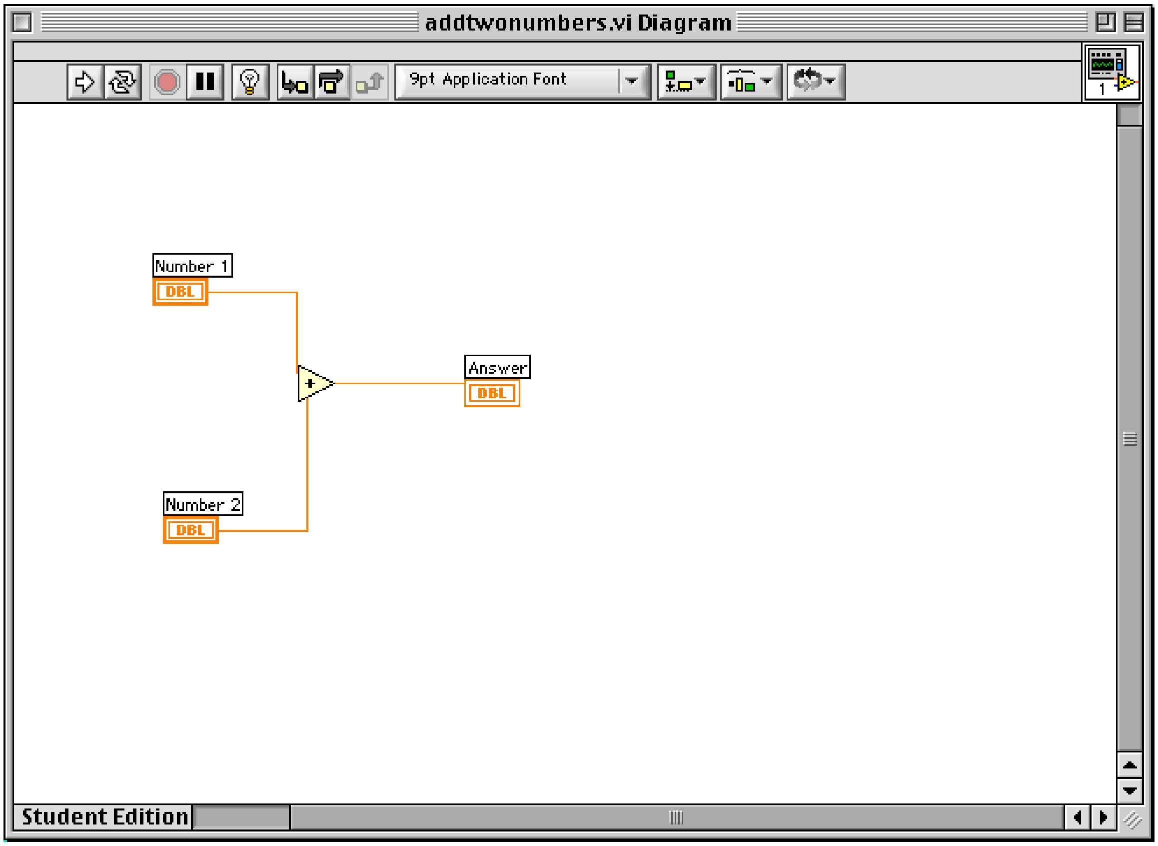
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



A Simple LabVIEW Front Panel

The simple LabVIEW front panel for the vi, addtwonumbers.vi, above allows the user to enter two numbers (Number 1 and Number 2) when the program is executed by pressing the run button. What happens when the run button is pressed is defined in the diagram. The diagram defines the logic and operations of the vi (LabVIEW program).

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



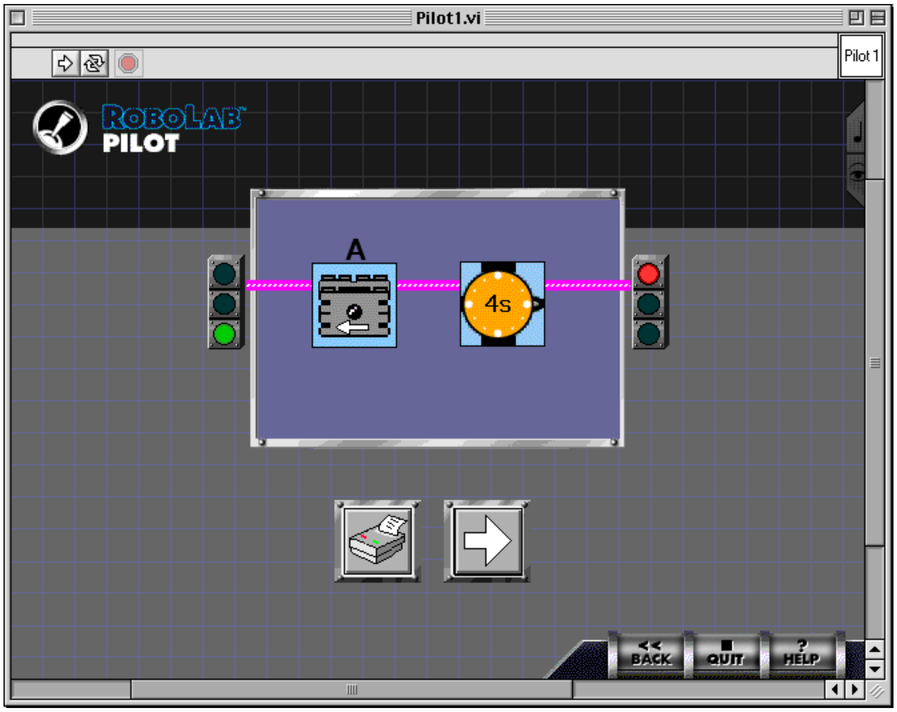
A Simple LabVIEW Diagram

The diagram above takes Number 1 and Number 2 and adds them together and displays the result in Answer. Addtwonumbers.vi is an overly simple example of LabVIEW's capabilities but helps to demonstrate the graphical interface and basic concepts.

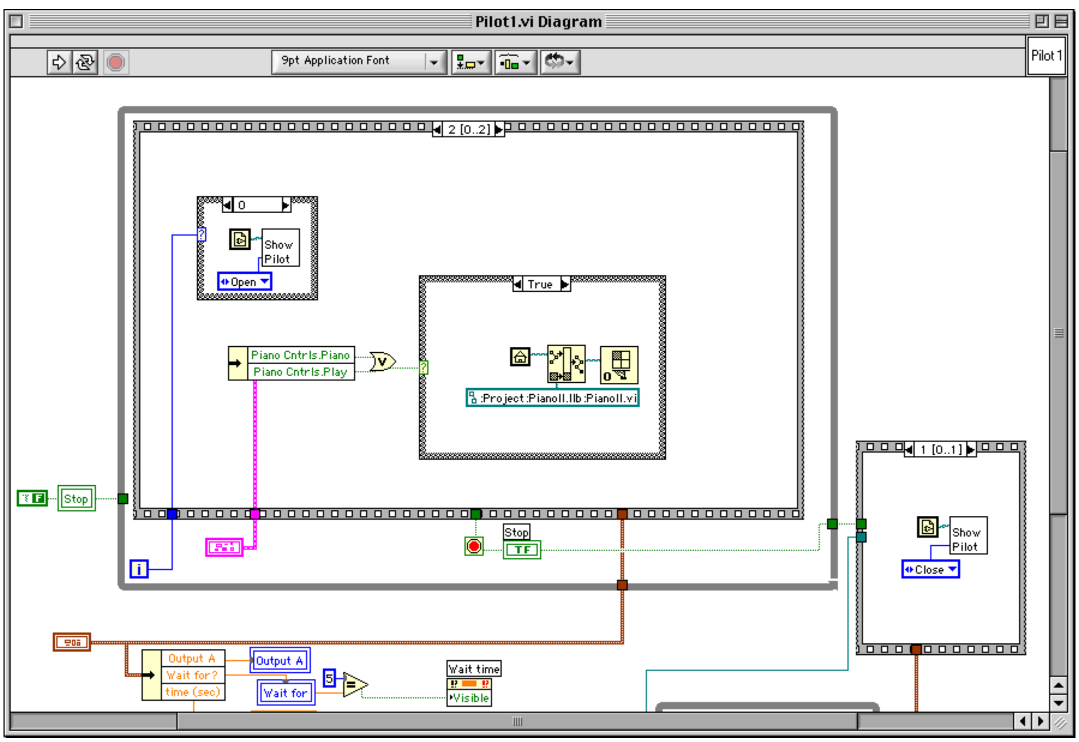
ROBOLAB makes use of LabVIEW's capabilities and interface in several ways. LabVIEW's intuitive graphical interface makes it easy for the developers of ROBOLAB to program quickly and easily. All the customized screens and templates, including the Administrator, the Pilot Levels, Investigator, Vision Center, and Piano Player are LabVIEW front panels. Behind those front panels are diagrams that contain the complicated programming that provides their functionality.

LEGO educational division

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



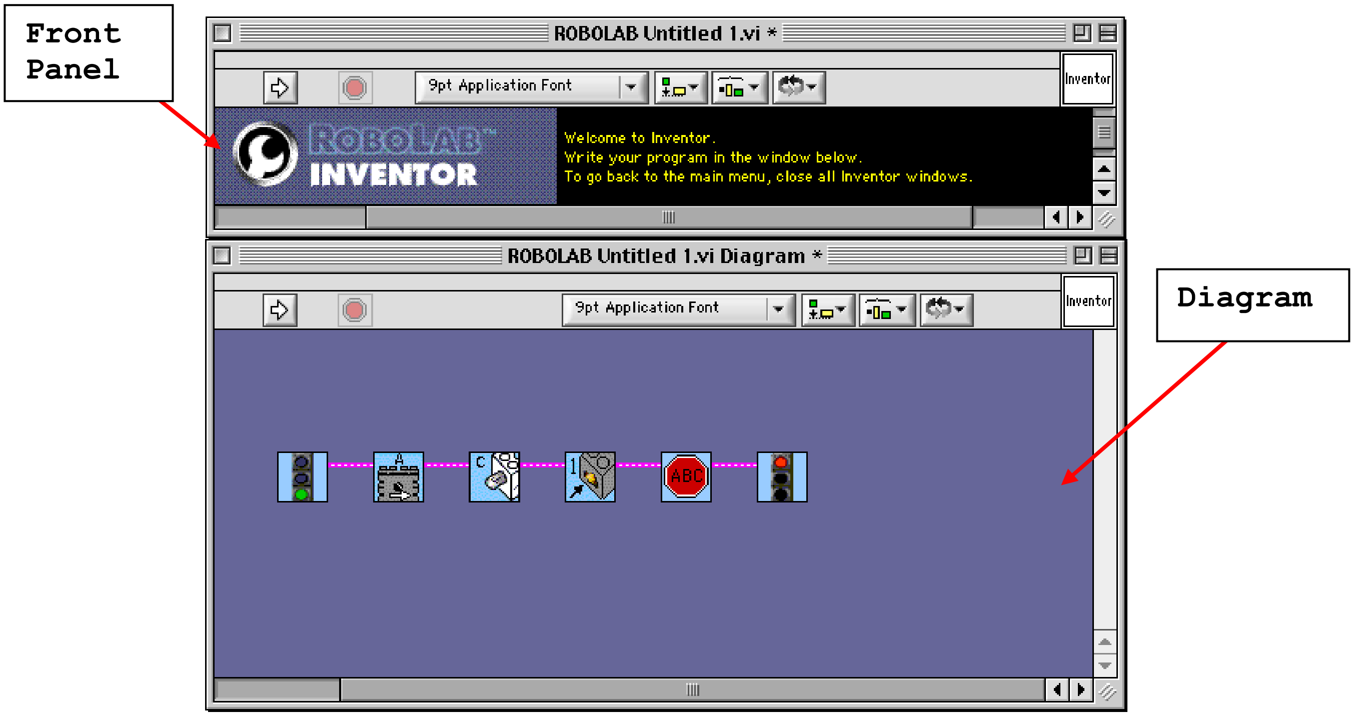
The Pilot 1 Front Panel



The Pilot 1 Diagram

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

At the Inventor Level and the higher programming levels of Investigator, users are actually doing modified LabVIEW programming. The bottom purple window that appears is actually a LabVIEW diagram.



When the run button is pressed, the code is compiled and sent to the RCX. The majority of ROBOLAB programming doesn't require the front panel so it is presented in a smaller format and used to display additional directions and information (like the 'Welcome to Inventor' message).

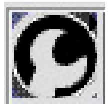
The similarities between the upper levels of ROBOLAB and LabVIEW mean that ROBOLAB users have much of the power of LabVIEW at their disposal allowing them to perform nearly any task or challenge they can conceive. It also means that users who wish to go further with programming have a solid foundation in this type of graphical programming environment and can easily transition to using LabVIEW.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

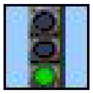
2. Programming Icons and Examples

This chapter provides additional hints and example programs for all of the ROBOLAB icons (excluding a few higher level palettes). The examples can also be accessed through the extended help in ROBOLAB 2.5 and higher (using the 'Click here for more help' link in the Help Window). The sections were created using the highest, most comprehensive level of ROBOLAB (Investigator – Program Level 5). Hence, if you are using lower levels of Inventor or Investigator some of the icons listed may not appear on your Functions Palette.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

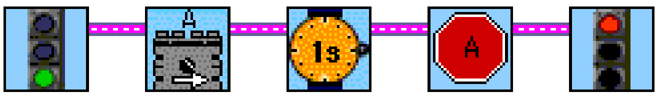


Basic Outputs

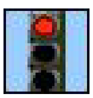

Begin

Always start an Inventor program with this command.

Example

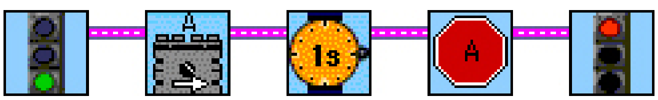


This piece of code would turn on motor A for one second and then turn it off.


End

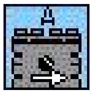
Always end an Inventor program with this command. Each task will need its own end command.

Example

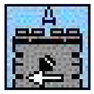


This piece of code would turn on motor A for one second and then turn it off.

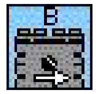
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



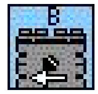
Motor A Forward



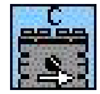
Motor A Reverse



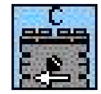
Motor B Forward



Motor B Reverse



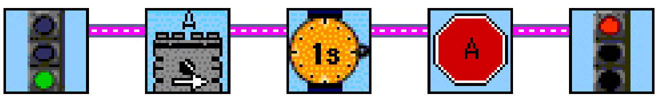
Motor C Forward



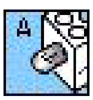
Motor C Reverse

Use this if you have a motor connected to port A, B, or C that will be spinning in the forward or reverse direction. Use the power level modifier to change how fast the motor spins, from 1 (slow) to 5 (fast). Without a modifier, the power level will be 5 (fast).

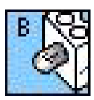
Example



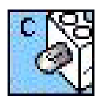
This piece of code would turn on motor A in the forward direction for one second and then turn it off.



Lamp A



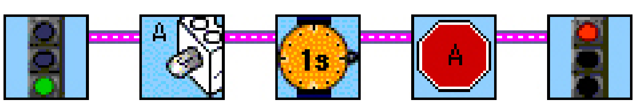
Lamp B



Lamp C

Use this if you want the lamps connected to port A, B, or C to turn on. Use the power level modifier to change it from bright (5) to dim (1). Without a modifier, the power level will be 5 (bright).

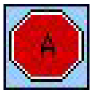
Example



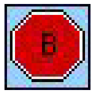
This piece of code would turn on the lamp connected to port A for one second and then turn it off.

LEGO educational division

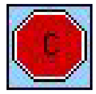
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



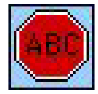
Stop A



Stop B



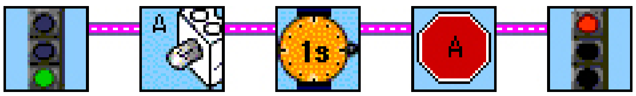
Stop C



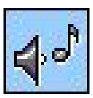
Stop all Outputs

Use this to stop the motor or lamp connected to ports A, B, and C by braking (abrupt stop). To get a more gradual stop, use the float command.

Example

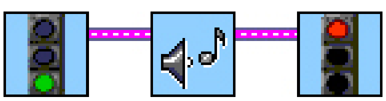


This piece of code would turn on the lamp connected to port A for one second and then turn it off.


Play Sound

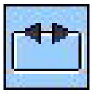
Use this command to get an audio response from the RCX. Change sounds using the sound type modifier. This modifier must be an integer from 1 to 6. Each represents a pre-programmed sound. 1: Key-click, 2: Beep Beep, 3: Descending Sweep, 4: Rising Sweep, 5: Buzz, 6: Fast Rising Sweep.

Example



This piece of code would play a fast rising sweep sound (the default).

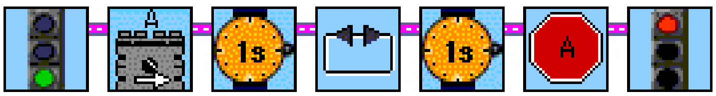
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



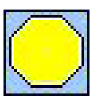
Flip Direction

Use this command to flip the directions of the motors once they are on. Use the ports modifier to flip specific ports. Without a modifier, this command will flip the direction of all ports. This does NOT turn the motor on. It needs to be used in combination with a command that starts the motor or at a point in your program where the motor is already on.

Example



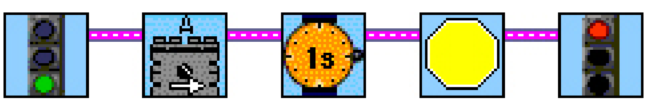
This piece of code would turn on motor A in the forward direction, wait one second, flip the direction of motor A (to reverse), wait one second, and then turn motor A off.



Float Outputs

Use this command if you want motors to come to a more gradual stop. This command will simply stop powering the motors. Use the ports modifier to float specific ports. Without a modifier, this command will float all ports.

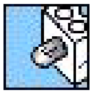
Example



This piece of code would turn on motor A in the forward direction for one second then it would float the motor.

LEGO educational division

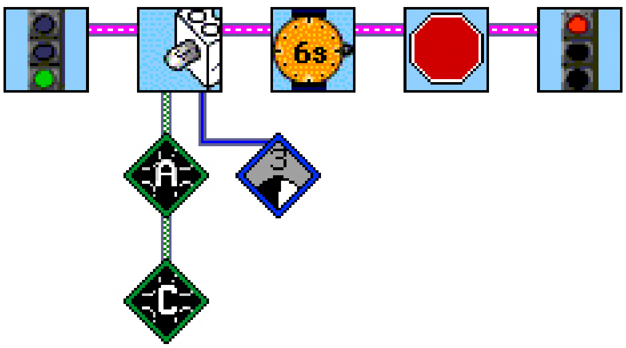
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



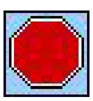
Lamp

Use this command to turn on lamps. The ports modifier allows you to choose which lamps you want to turn on. If no modifier is used, this command will turn on all ports. Use the power level command to change from 5 (bright) to 1 (dim). Without a modifier, the power level will be 5.

Example



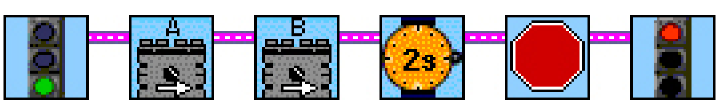
This piece of code would turn on lamps connected to ports A and C at power level 3. After 6 seconds, it would turn them off.



Stop Outputs

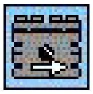
Use this to stop motors or turn off lamps. This command abruptly stops motors by braking. To get a more gradual stop, use the float command. Use modifiers to choose which ports to stop. Without a modifier, this command stops all ports.

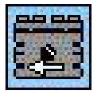
Example



This piece of code would turn on motors A and B in the forward direction for two seconds and then stop both motors.

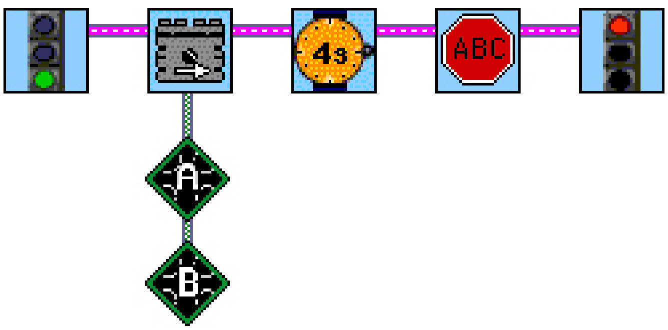
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.


Motor Forward


Motor Reverse

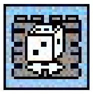
Use this to turn on the motor(s) in the forward direction. Use the ports modifier to choose which motors to turn on. Without a modifier, this command will turn on all ports. Use the power level modifier to change how fast the motor spins, from 1 (slow) to 5 (fast). Without a modifier, the power level will be 5 (fast).

Example



This piece of code would turn on motors connected to ports A and B in the forward direction. After 4 seconds, it would turn them off.

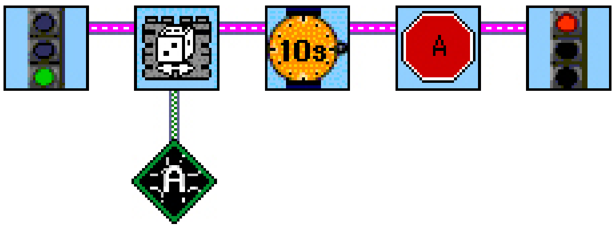
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



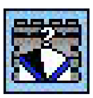
Motor Random

Use this icon when you want the motor to turn on in a random direction. This icon DOES turn the motor on. The default is to turn on all ports to power level 5.

Example



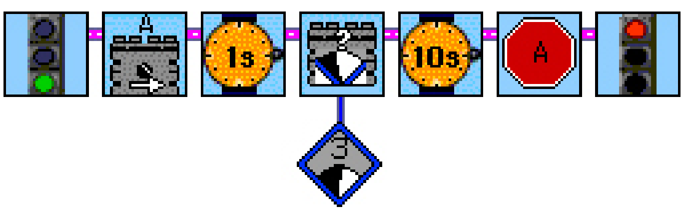
This piece of code turns motor A on in a random direction. The motor runs for 10 seconds, and then stops.



Change Motor Speed

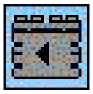
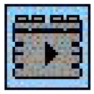
Use this icon when you want to change or assign motor speeds. This does NOT turn the motor on. It needs to be used in combination with a command that starts the motor or at a point in your program where the motor is already on.

Example



This piece of code turns on motor A in the forward direction at full power for one second, then it changes to power level 3 for ten seconds and turns off.

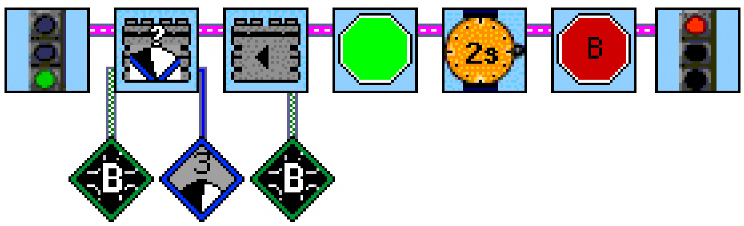
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Reverse **Forward**

This command just sets the direction of the motor to forward or reverse. It does NOT turn the motor on. It needs to be used in combination with a command that starts the motor or at a point in your program where the motor is already on. If the motor is already going in the reverse or forward direction, there will be no change.

Example



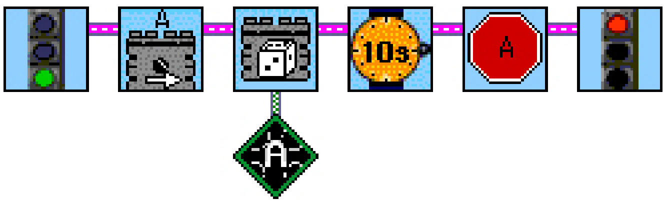
This piece of code assigns the speed of motor B to 3, and the direction of motor B to reverse, and then turns it on. After 2 seconds it turns the motor off.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

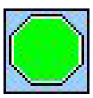
Random Direction

Use this icon when you want the direction of the motor to be randomly selected. This does NOT turn the motor on. It needs to be used in combination with a command that starts the motor or at a point in your program where the motor is already on.

Example



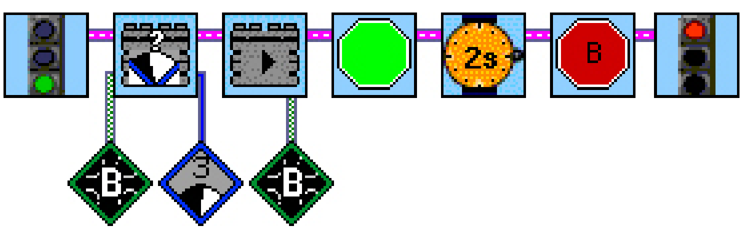
This piece of code turns motor A on in the forward direction and then randomly selects a direction for the motor (forward or backward), the motor runs for 10 seconds and is then stopped.



Turn Outputs On

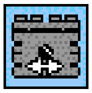
Use this command to turn on motors and lamps with their last power and direction settings.

Example



This piece of code assigns the speed of motor B to 3, and the direction of motor B to forward, and then turns it on. After 2 seconds it turns motor B off.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



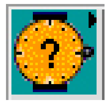
Motor Forward or Back

Use this command when you want the speed and direction of the motor related to positive and negative numbers.

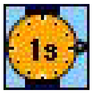
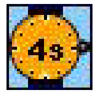
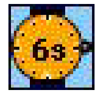
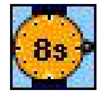
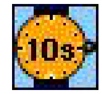
Example

This piece of code puts the value -5 in the red container. It then loops 11 times running the motor at the speed in the red container for 1 second , beeping and then augmenting the value of the container by 1. Hence, the motor runs backwards from speeds 5 - 1, stops while the power is equal to 0, and the runs forward from speeds 1-5.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

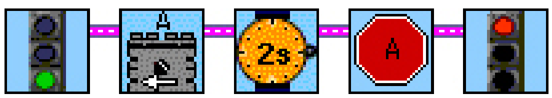


Wait Fors

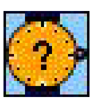
					
Wait for 1 second	**Wait for 2 seconds**	**Wait for 4 seconds**	**Wait for 6 seconds**	**Wait for 8 seconds**	**Wait for 10 seconds**

Use these icons if you want to wait for the specified amount of time before moving on to the next command. Useful to choose duration of motor and lamp activity.

Example



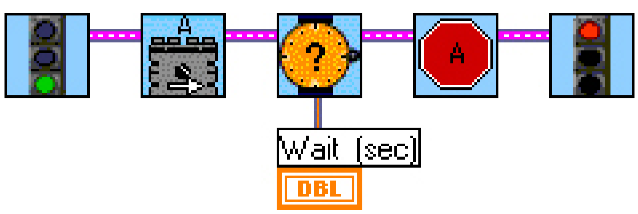
This piece of code would turn on motor A in the reverse direction for 2 seconds, and then turn it off.



Wait for Time

Use this icon if you want to specify the number of seconds to wait before moving on to the next command. Useful to choose duration of motor and lamp activity.

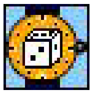
Example



This piece of code would turn on motor A for the number of seconds specified on the control panel and then turn it

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

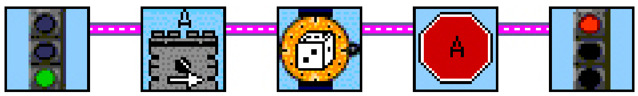
off.



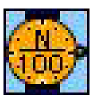
Wait for Random Time

Use this if you want to wait a random amount of time before moving on to the next command. Use modifier to change the max random time from 5 seconds. Useful to choose duration of motor and lamp activity.

Example



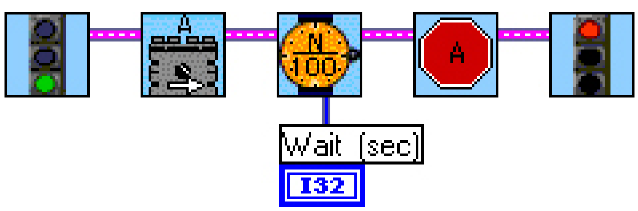
This piece of code would turn on motor A for a random amount of time between 0 and 5 seconds and then turn it off.



Wait for N hundredths of a second

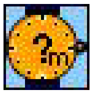
Use this if you want to wait a certain number of hundredths of seconds before moving on to the next command. Useful to choose duration of motor and lamp activity.

Example



This piece of code would turn on motor A for the number of hundredths of seconds specified on the control panel and then turn it off.

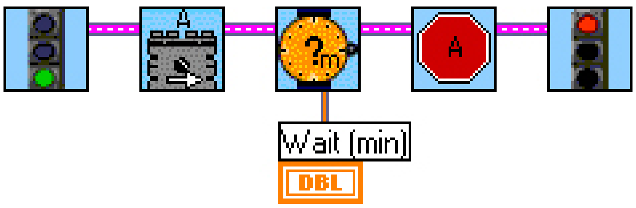
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



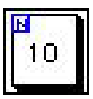
Wait for Time (min)

Use this if you want to wait a certain number of minutes before moving on to the next command. Useful to choose duration of motor and lamp activity.

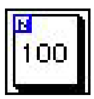
Example



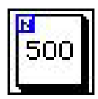
This piece of code would turn on motor A for the number of minutes specified on the control panel and then turn it off.



Wait for 10 Points



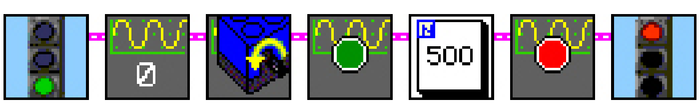
Wait for 100 Points



Wait for 500 Points

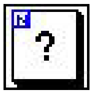
Most useful in Investigator. Use these icons when you want to collect the specified number of data points before moving on to the next command.

Example



This piece of code clears the data memory, then logs 500 points of rotation sensor data.

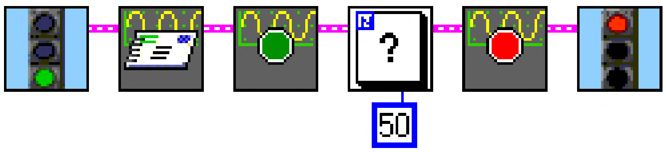
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



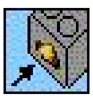
Wait for N Points

Most useful in Investigator. Use this icon when you want to collect a certain number of data points before moving on to the next command.

Example



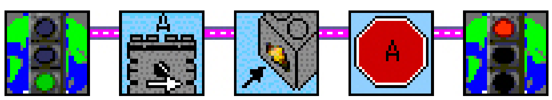
This piece of code would take 50 data points from the mailbox.



Wait for Push

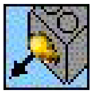
Use this icon to wait for something to push in the touch sensor before moving on to the next command.

Example



If an IP number was strung onto the Begin icon, this piece of code would be sent over the Internet. It would turn on motor A until a touch sensor was pressed and then shut it off.

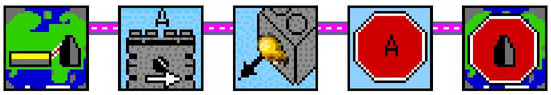
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



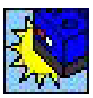
Wait for Let Go

Use this icon to wait for something to release the touch sensor before moving on to the next command. If the touch sensor is not pressed to begin with, the touch sensor must be pressed then released.

Example



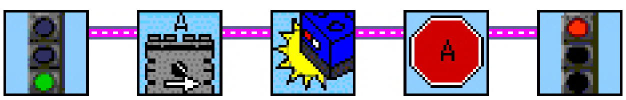
If an IP number was strung onto the Begin direct icon, this piece of code would control an RCX in direct mode over the Internet. It would turn on motor A on the remote RCX until a touch sensor was released and then turn it off.



Wait for Light

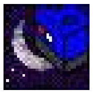
Use this icon to wait for the light reading to become greater than a certain value before moving on to the next command.

Example



This piece of code turns on motor A, then waits for the light sensor to read a value greater than 55 (the default). Then, it turns off the motor.

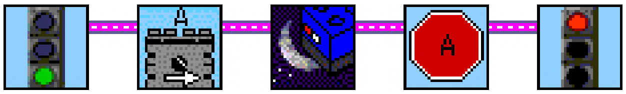
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



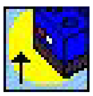
Wait for Dark

Use this icon to wait for the light reading to become less than a certain value before moving on to the next command.

Example



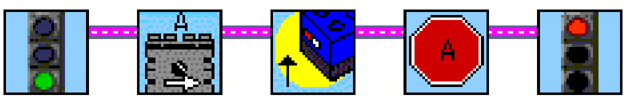
This piece of code turns on motor A, then waits for the light sensor to read a value less than 55 (the default). Then, it turns off the motor.



Wait for Brighter

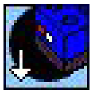
Use this icon to wait for the light reading to increase by a certain value before moving to the next command.

Example



This piece of code turns on motor A, then waits for the light reading to increase by 5 (the default). Then, it turns off the motor.

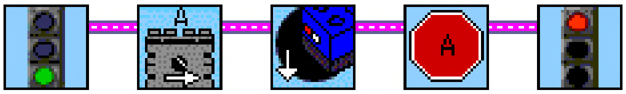
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Wait for Darker

Use this icon to wait for the light reading to decrease by a certain value before moving to the next command.

Example

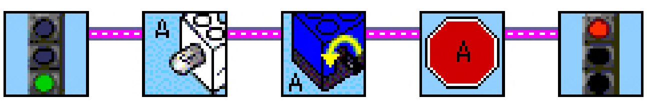


This piece of code turns on motor A, then waits for the light reading to decrease by 5 (the default). Then, it turns off the motor.

Wait for Rotation w/o Reset

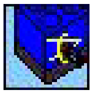
Use this icon to wait for a number of rotations (or a fraction of a rotation) to be completed but do NOT want to reset the rotation sensor to 0. (The general wait for rotation sensor resets to 0 and then waits for the cutoff rotation.) The angle sensor value is measured in sixteenths, so a reading of 16 is one rotation.

Example



This piece of code turns on lamp A, waits for 1 rotation (a reading of 16, the default) from the sensor, then turns lamp A off.

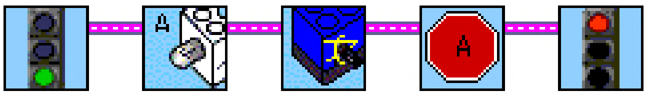
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



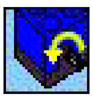
Wait for Angle

Use this icon to wait for the angle sensor value to be greater than the cutoff value before moving on to the next command. The rotation can be either forward or backward. The default is 180 degrees, or half a rotation.

Example



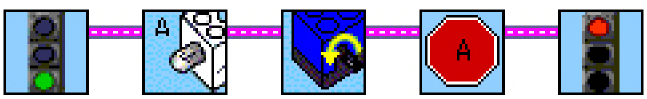
This piece of code turns on lamp A. Once the angle sensor reads a value of greater than 180 degrees (the default), lamp A turns off.



Wait for Rotation

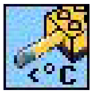
Use this icon to wait for the angle sensor value to be greater than the value specified before moving on to the next command. The angle sensor value is measured in sixteenths, so a reading of 16 is one rotation.

Example



This piece of code turns on lamp A. Once the angle sensor value reaches 16 (or one rotation, the default), lamp A turns off.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Wait for Decreasing Temp (C)

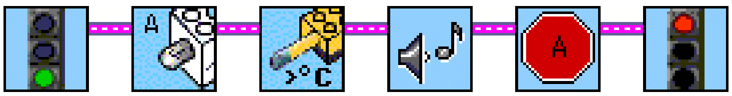
Wait for Increasing Temp (C)

Wait for Decreasing Temp (F)

Wait for Increasing Temp (F)

Use these icons to wait for the temperature sensor to read a value less (Wait for Increase) or greater than (Wait for Decrease) than the cutoff temperature specified before moving on to the next command.

Example

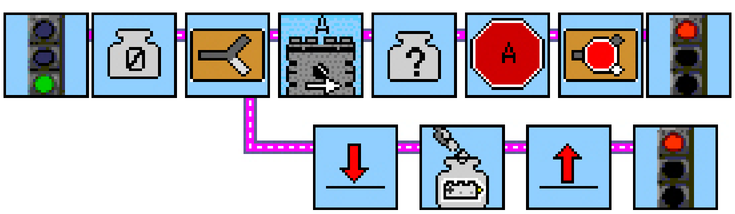


This piece of code turns on lamp A,then waits for the temperature sensor to read above 30 degrees Celsius (the default). After that it plays a sound and turns off the lamp.

Wait for Container

Use this icon to wait for the value of a container to be equal to some number before moving on to the next command.

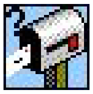
Example



This piece of code empties the red (default) container. Then it turns on motor A. It waits for the container value to equal 1 (the default), and when it does, it turns off motor A and stops all tasks. At the same time, a second task is filling the container with the value of the touch sensor. In essence, this program waits for the touch sensor

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

to be pressed, and then turns off the motor.



Wait for Mail

Use this icon to wait for mail before moving on to the next command.

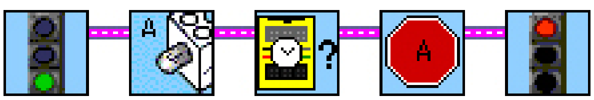
Example

This piece of code turns on motor A and waits to receive mail. Once the mail is received, it turns off motor A.

Wait for Clock

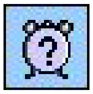
Use this icon to wait for the clock on the RCX to reach a certain time value in minutes before moving on to the next command. Useful if you do not want the RCX to stay on too long to conserve batteries.

Example



This piece of code turns on lamp A and waits for the RCX clock to reach 1 minute (the default), then turns off the lamp.

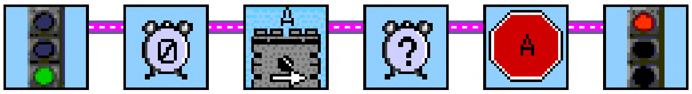
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



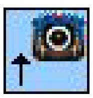
Wait for Timer

Use this icon to wait until the clock reaches a certain time (measured in tenths of seconds) before moving on to the next command. Always use a zero timer command prior to this command.

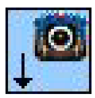
Example



This program zeros the red timer and turns on motor A. Then it waits for the red timer to reach 1 second (the default), after which it stops motor A.



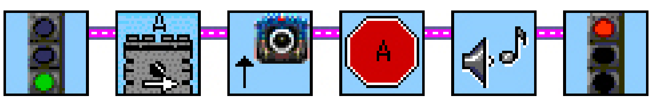
Wait for Increase in Camera Sensor



Wait for Decrease in Camera Sensor

Use these icons to wait until the camera sensor reads a value higher (Increase icon) or lower (Decrease icon) than the compare to number before moving on to the next command.

Example



This piece of code would turn on motor A. When the camera sensor reads a value of greater than 55 (the default), The motor would turn off and the RCX would play a sound.

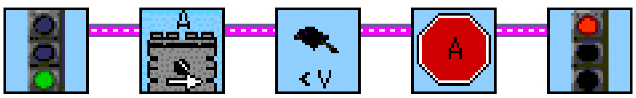
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Wait for Decrease in Voltage (Generic)

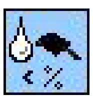
Wait for Increase in Voltage (Generic)

Use these icons to wait until the sensor reads a voltage of less than (Wait for Decrease icon) or greater than (Wait for Increase icon) the cutoff voltage specified before moving on to the next command.

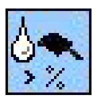
Example



This piece of code would turn on motor A. When the sensor reads a value of less than 2 Volts (the default), motor A would turn off.



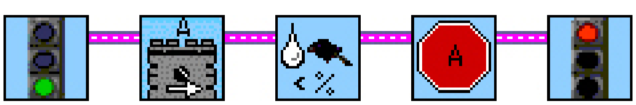
Wait for Decreasing Humidity



Wait for Increasing Humidity

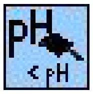
Use these icons to wait until the sensor reads a value of less than (Wait for Decrease icon) or a value of greater than (Wait for Increase icon) the cutoff humidity percentage before moving on to the next command.

Example



This piece of code would turn on motor A. When the sensor reads a value of less than 50% humidity (the default), motor A would be shut off.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

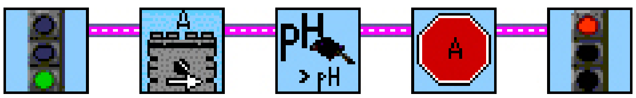


Wait for Decreasing pH

Wait for Decreasing pH

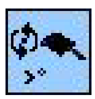
Use these icons to wait until the sensor reads a value of less than (Wait for Decrease) or greater than (Wait for Increase) the cutoff pH before moving on to the next command.

Example



This piece of code would turn on motor A. When the sensor reads a value of greater than 7 (the default), motor A would be shut off.

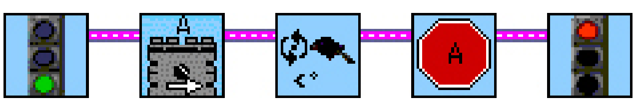
Wait for Position Increase



Wait for Position Decrease

Use these icons to wait until the sensor reads a value of less than (Wait for Position Increase icon) or greater than (Wait for Position Decrease icon) the cutoff position before moving on to the next command.

Example



This piece of code would turn on motor A. When the sensor reads a value of less than 180 degrees (the default), the motor would be shut off.

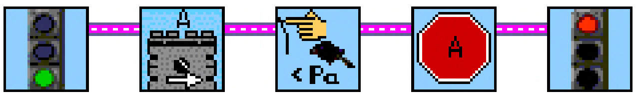
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Wait for Increasing Pressure

Wait for Decreasing Pressure

Use these icons to wait until the sensor reads a value of less than the cutoff pressure before moving on to the next command.

Example



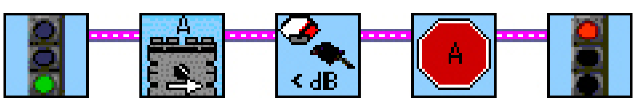
This piece of code would turn on motor A. When the sensor reads a value of less than 100 kPa (the default), the motor would be shut off.

Wait for Increasing Sound Level

Wait for Decreasing Sound Level

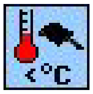
Use these icons to wait until the sensor reads a value of less than (Wait for Increase icon) or greater than (Wait for Decrease icon) the cutoff sound level before moving on to the next command.

Example

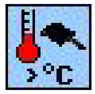


This piece of code would turn on motor A. When the sensor reads a value of less than 60 decibels (the default), the motor would be shut off.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



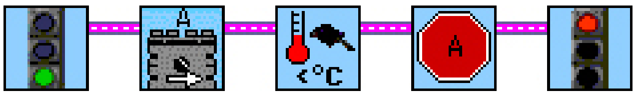
Wait for Increasing Temperature (C)



Wait for Decrease in Temperature (C)

Use these icons to wait until the sensor reads a value of less than (Wait for Increase icon) or greater than (Wait for Decrease icon) the cutoff temperature before moving on to the next command.

Example



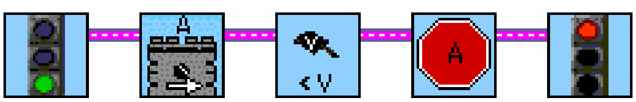
This piece of code would turn on motor A. When the sensor reads a value of less than 30 degrees Celsius (the default), the motor would be shut off.

Wait for Increase in Voltage

Wait for Decrease in Voltage

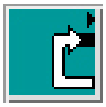
Use these icons to wait until the sensor reads a value of less than (Wait for Increase icon) or greater than (Wait for Decrease icon) the cutoff voltage before moving on to the next command.

Example

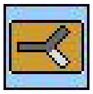


This piece of code would turn on motor A. When the sensor reads a value of less than 2V (the default), the motor would be shut off.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



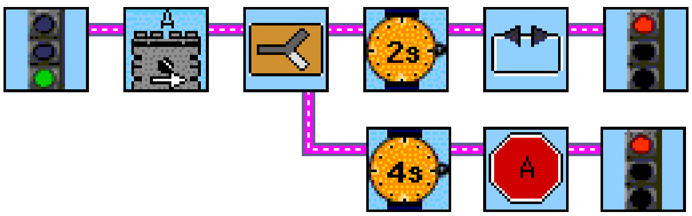
Tasks & Subroutines (Structures)



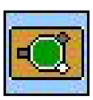
Task Split

Use this icon to start a new task. The tasks run simultaneously. Each task needs an End icon (stoplight).

Example



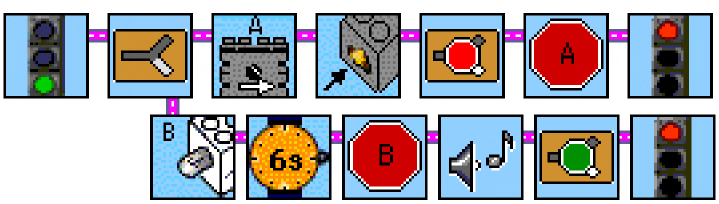
This piece of code turns on motor A. After 2 seconds it flips directions, and after 4 seconds from the beginning (2 seconds from flip) it turns off motor A.



Start Task

Use this to restart tasks after they have been completed or stopped. The Task Number modifier is only necessary if you only wish to restart a certain task and not all tasks.

Example

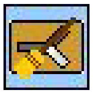


This piece of code turns on motor A and lamp B. If 6 seconds pass and the touch sensor has not been pressed, the RCX will play a noise and turn off lamp B. Then both tasks will be restarted from the task split. If the touch sensor

LEGO educational division

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

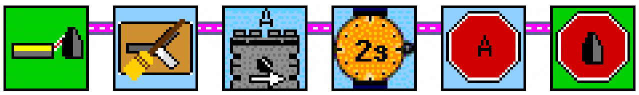
is pressed within 6 seconds, both tasks will be stopped and the program ends.



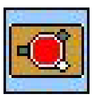
Delete Tasks

Use this in direct mode ONLY. Use this icon to clear the RCX of any tasks it previously had. The Task Number modifier is only necessary if you want to delete only a certain task and not all tasks.

Example



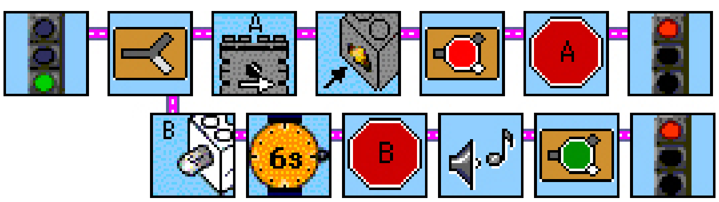
This piece of code deletes all tasks then turns on motor A for 2 seconds and turns it off.



Stop Tasks

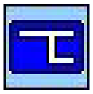
Use this to stop tasks that may still be in progress. The Task Number modifier is only necessary if you want to stop a certain task and not all tasks.

Example



This piece of code turns on motor A and lamp B. If 6 seconds pass and the touch sensor has not been pressed, the RCX will play a noise and turn off lamp B. Then both tasks will be restarted from the task split. If the touch sensor is pressed within 6 seconds, both tasks will be stopped and the program ends.

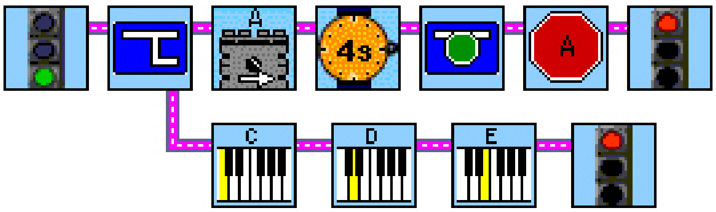
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



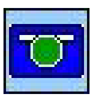
Create Subroutine

Use this to define a subroutine. This does NOT start the subroutine. Each subroutine needs its own end (stoplight) icon. The main task will continue running by itself until a Run Subcommand is reached.

Example



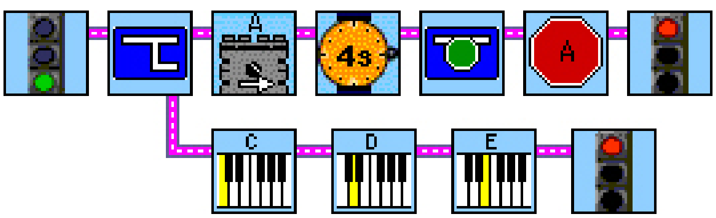
This piece of code turns on motor A. After 4 seconds it plays 3 notes and then turns off the motor.



Start Subroutine

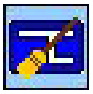
Use this to start a subroutine. You must define the subroutine before using this command.

Example



This piece of code turns on motor A. After 4 seconds it plays 3 notes and then turns off the motor.

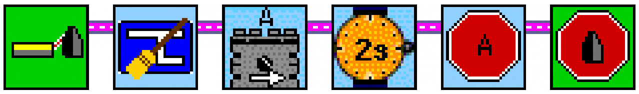
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Delete Subroutine

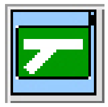
Use this in direct mode ONLY. Use this icon to clear the RCX of any subroutines it previously had.

Example

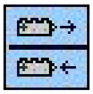


This piece of code clears the RCX of any subroutines and then turns on motor A for 2 seconds and turns it off.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



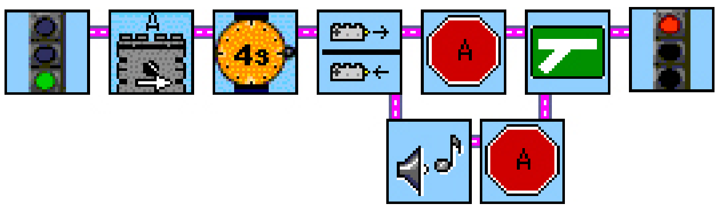
Forks



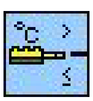
Touch Sensor Fork

Use this fork if you want to take different courses of action based on a touch sensor. A fork merge is required at some point after this icon.

Example



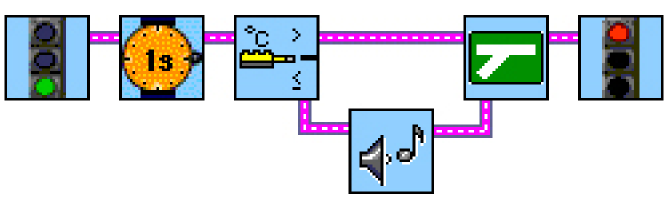
This piece of code would turn on motor A. If, after 4 seconds, the touch sensor is pushed, the RCX will play a sound and shut off the motor. Otherwise, the motor will just be turned off (no sound).



Celsius Fork

Use this fork if you want to take different courses of action based on a temperature sensor. A fork merge is required at some point after this icon.

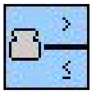
Example



This piece of code waits one second and reads the temperature. If the temperature is 30 degrees Celsius or

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

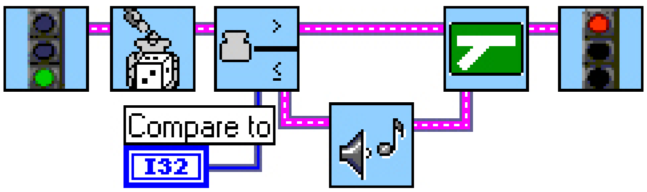
less, it will play a sound.



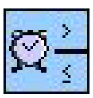
Container Fork

Use this fork if you want to take different courses of action based on the value of a container. A fork merge is required at some point after this icon.

Example



This piece of code fills a container with a random number. If that number is less than the number specified by the compare to control, the RCX plays a sound.



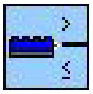
Timer Fork

Use this fork if you want to take different courses of action based on the value of a timer. A fork merge is required at some point after this icon.

Example

This piece of code turns on lamp A for a random amount of time, and then turns it off. If the time is less than 3 seconds, the RCX plays a sound.

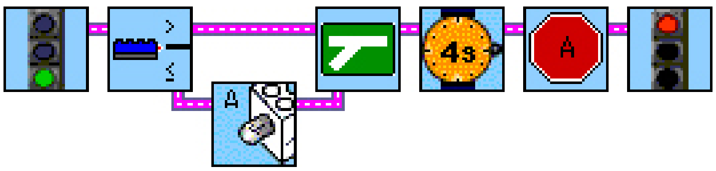
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



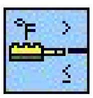
Light Sensor Fork

Use this fork if you want to take different courses of action based on a light sensor. A fork merge is required at some point after this icon.

Example



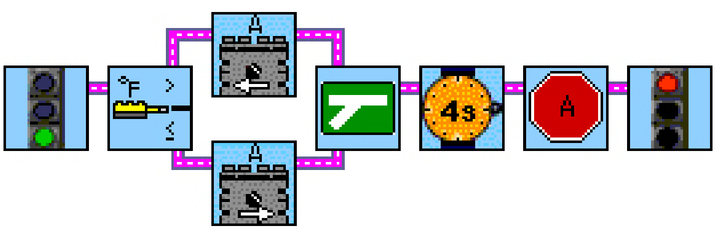
This piece of code turns on lamp A for 4 seconds if the light sensor reads a value of below 55 (the default).



Fahrenheit Fork

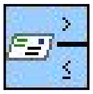
Use this fork if you want to take different courses of action based on a temperature sensor. A fork merge is required at some point after this icon.

Example



This piece of code turns on motor A for 4 seconds. The direction is determined by the temperature. If it reads above 80 degrees, the motor will go forward. Otherwise, it will go in reverse.

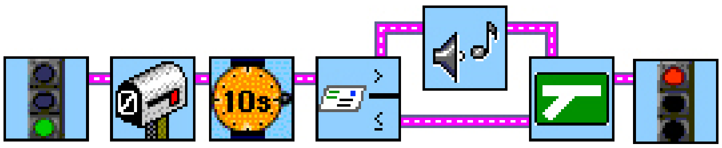
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



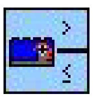
Mailbox Fork

Use this fork if you want to take different courses of action based on the value of the mailbox. A fork merge is required at some point after this icon. Also, an empty mailbox command should be used sometime before this icon.

Example



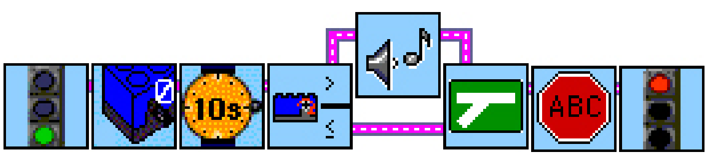
This piece of code would empty the mailbox. After ten seconds, it checks to see if it has received mail whose value is greater than 1 (the default). If it has, the RCX plays a sound.



Rotation Sensor Fork

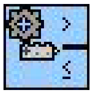
Use this fork if you want to take different courses of action based on a rotation sensor. A fork merge is required at some point after this icon. Also, a zero angle sensor command should be used sometime before this icon. Reminder: rotation sensor reads in sixteenths, so a reading of 16 equals one rotation. Rotation can be either forward or backward.

Example



This program zeros the angle sensor then waits 10 seconds. If at that point the rotation sensor reads greater than 1 rotation (a reading of 16), then the RCX plays a sound.

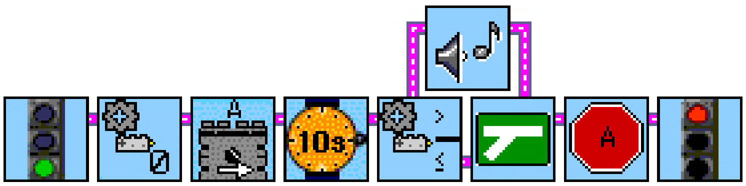
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



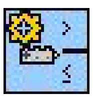
Click Fork

Use this fork if you want to take different courses of action based on the number of clicks on a touch sensor. A fork merge is required at some point after this icon. Also, a zero clicks sensor command should be used sometime before this icon.

Example



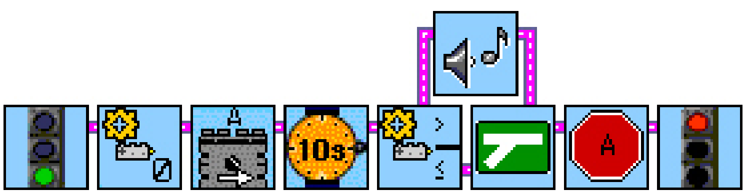
This piece of code turns on motor A for 10 seconds, then shuts it off. If after 10 seconds the touch sensor has been clicked more than 10 times, it plays a sound.



Touch and Release Fork

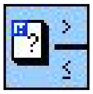
Use this fork if you want to take different courses of action based on the number of touches and releases on a touch sensor. A fork merge is required at some point after this icon. Also, a zero touch and release sensor command should be used sometime before this icon.

Example



This piece of code turns on motor A for 10 seconds, and then shuts it off. If after 10 seconds the touch sensor has been touched and releases more than 10 times, it plays a sound.

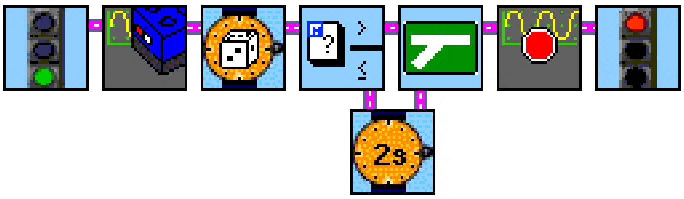
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



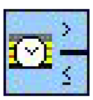
Data Points Fork

Use with Investigator commands. Use this fork if you want to take different courses of action based on the number of points gathered. A fork merge is required at some point after this icon.

Example



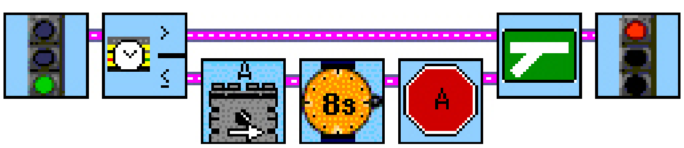
This piece of code logs light sensor data points for a random time. If after that time there are 5 (the default) or fewer data points, it logs data for 2 more seconds before stopping. If after the random time there are greater than 5 data points, it stops logging points.



Clock Fork

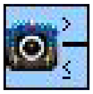
Use this fork if you want to take different courses of action based on the value of the RCX clock. A fork merge is required at some point after this icon.

Example



This piece of code would check the value of the RCX clock. If the clock read 1 minute (the default) or less, then motor A would turn on for 8 seconds.

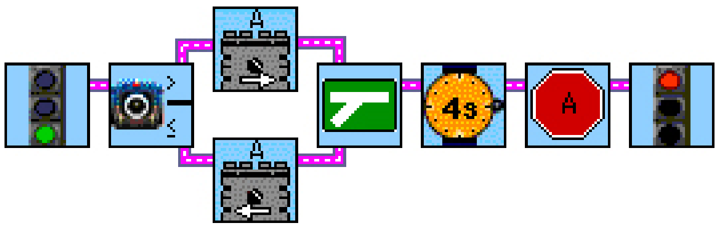
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



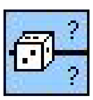
Camera Sensor Fork

Use this fork if you want to take different courses of action based on the value of the camera sensor. A fork merge is required at some point after this icon.

Example



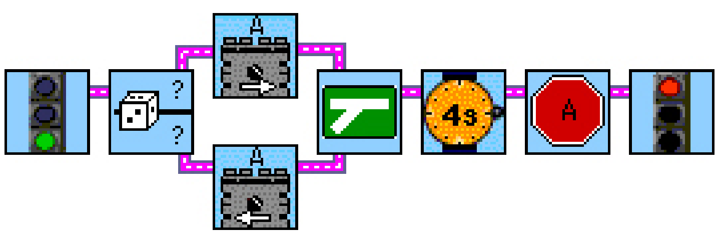
This piece of code would turn on motor A for 4 seconds, then turn it off. The direction of the motor is determined by the value of the camera sensor. If the camera value is over 55 (the default), the motor goes forward. Otherwise, the motor goes in the reverse direction.



Random Fork

Use this if you want different courses of action to be decided on randomly. A fork merge is required at some point following this icon.

Example



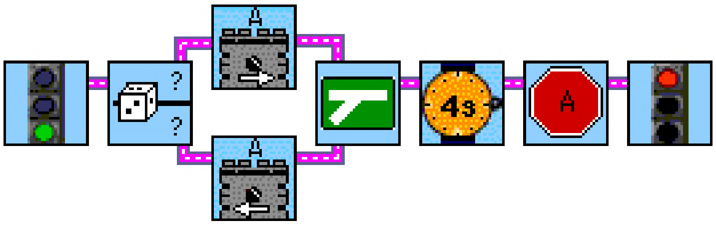
This piece of code would turn on motor A for 4 seconds. The direction would be decided on randomly.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

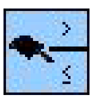
Fork Merge

ALWAYS needed at the end of a fork to merge two branches back together.

Example



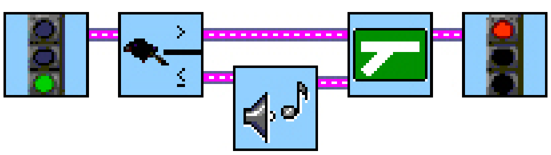
This piece of code would turn on motor A for 4 seconds. The direction would be decided on randomly.



Generic Sensor Fork

Use this fork if you want to take different courses of action based on the value of a generic sensor adapter. A fork merge is required at some point after this icon.

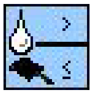
Example

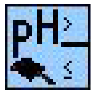


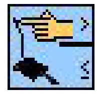
This piece of code would play a sound if the value of the generic sensor adapter was less than or equal to 2 volts (the default).

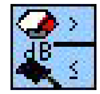
LEGO educational division

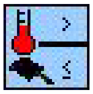
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

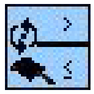

Humidity Sensor Fork

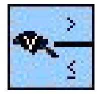

pH Sensor Fork


Pressure Sensor Fork


Sound Level Sensor Fork

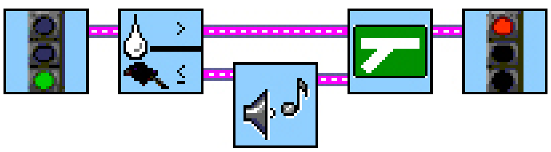

Temperature Sensor Fork


Position Sensor Fork


Voltage Fork

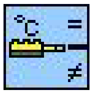
Use this fork if you want to take different courses of action based on the value of a LogIT sensor. A fork merge is required at some point after this icon.

Example



This piece of code would play a sound if the value of the humidity sensor was less than or equal to 50% (the default).

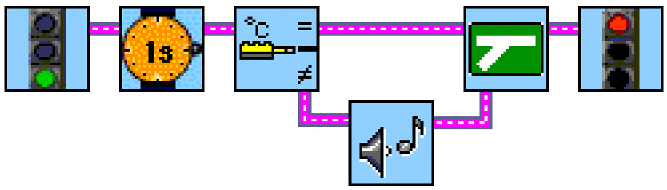
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Celsius Equal Fork

Use this fork if you want to take different courses of action based on a temperature sensor. A fork merge is required at some point after this icon.

Example

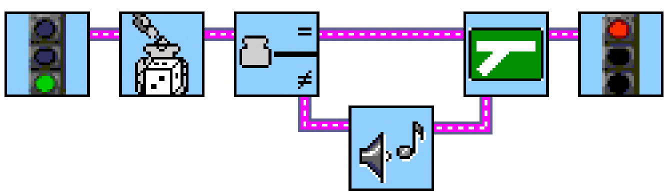


This piece of code waits one second and reads the temperature. If the temperature is not 30 degrees Celsius (the default), it will play a sound.

Container Equal Fork

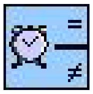
Use this fork if you want to take different courses of action based on the value of a container. A fork merge is required at some point after this icon.

Example



This piece of code fills a container with a random number. If that number is not equal to 1 (the default), the RCX plays a sound.

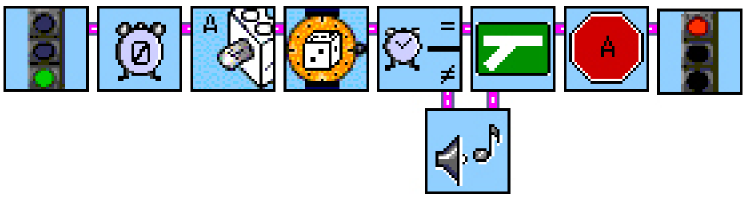
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



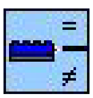
Timer Equal Fork

Use this fork if you want to take different courses of action based on the value of a timer. A fork merge is required at some point after this icon. Reminder: Comparison value is measured in tenths of seconds.

Example



This piece of code turns on lamp A for a random amount of time, and then turns it off. If the time is not equal to 5 seconds (the default), the RCX plays a sound.



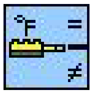
Light Sensor Equal Fork

Use this fork if you want to take different courses of action based on a light sensor. A fork merge is required at some point after this icon.

Example

This piece of code turns on motor A. If the light sensor does not read a value of 55, then the RCX plays a sound. After 2 seconds, the motor is shut off.

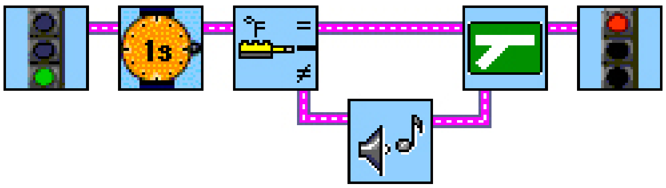
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



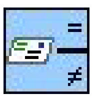
Fahrenheit Equal Fork

Use this fork if you want to take different courses of action based on a temperature sensor. A fork merge is required at some point after this icon.

Example



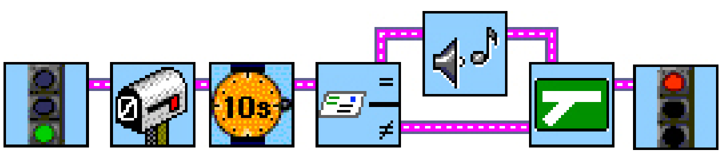
This piece of code waits one second and reads the temperature. If the temperature is not 80 degrees Fahrenheit (the default), it will play a sound.



Mailbox Equal Fork

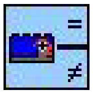
Use this fork if you want to take different courses of action based on the value of the mailbox. A fork merge is required at some point after this icon. Also, an empty mailbox command should be used sometime before this icon.

Example



This piece of code empties the mailbox and then waits 10 seconds. If after those 10 seconds the mailbox has a value of 1 (the default), the RCX plays a sound.

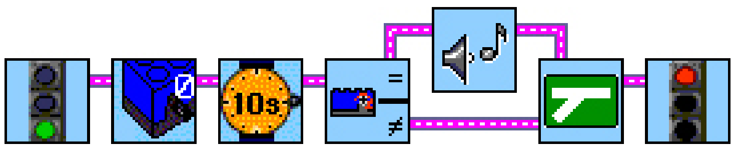
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



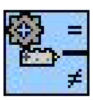
Rotation Sensor Equal Fork

Use this fork if you want to take different courses of action based on a rotation sensor. A fork merge is required at some point after this icon. Also, a zero angle sensor command should be used sometime before this icon. Reminder: rotation sensor reads in sixteenths, so a reading of 16 equals one rotation. Rotation can be either forward or backward.

Example



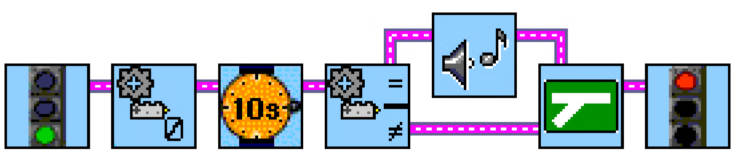
This piece of code zeros the angle sensor. If after 10 seconds the angle sensor reads 16 (one rotation, the default), then the RCX plays a sound.



Click Equal Fork

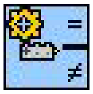
Use this fork if you want to take different courses of action based on the number of clicks on a touch sensor. A fork merge is required at some point after this icon. Also, a zero clicks sensor command should be used sometime before this icon.

Example



This piece of code zeros the click sensor. If after 10 seconds the click sensor reads 10 (the default), the RCX plays a sound.

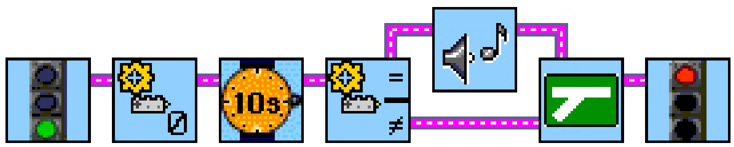
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



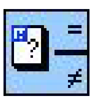
Touch and Release Equal Fork

Use this fork if you want to take different courses of action based on the number of touches and releases on a touch sensor. A fork merge is required at some point after this icon. Also, a zero touch and release sensor command should be used sometime before this icon.

Example



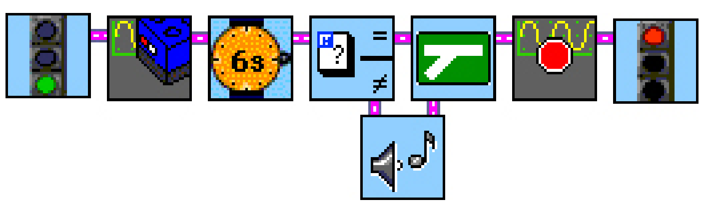
This piece of code zeros the touch and release sensor. If after 10 seconds the touch and release sensor reads 10 (the default), the RCX plays a sound.



Data Points Equal Fork

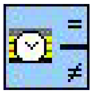
Use with Investigator commands. Use this fork if you want to take different courses of action based on the number of points gathered. A fork merge is required at some point after this icon.

Example



This piece of code logs light sensor data points for 6 seconds. If after that time there are not 5 (the default) data points, the RCX plays a sound before stopping. If after the 6 seconds there are 5 data points, it stops logging points (no sound).

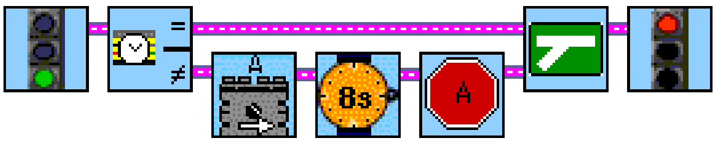
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Clock Equal Fork

Use this fork if you want to take different courses of action based on the value of the RCX clock. A fork merge is required at some point after this icon.

Example

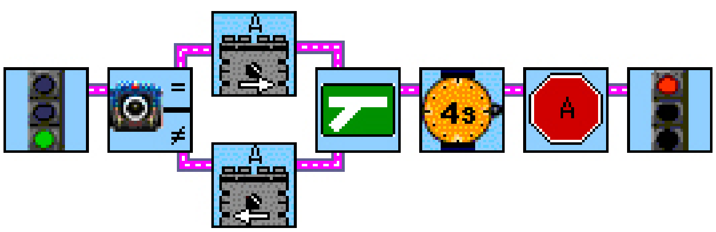


This piece of code would check the value of the RCX clock. If the clock did not read 1 minute (the default), then motor A would turn on for 8 seconds.

Camera Sensor Equal Fork

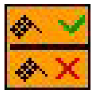
Use this fork if you want to take different courses of action based on the value of the camera sensor. A fork merge is required at some point after this icon.

Example



This piece of code would turn on motor A for 4 seconds, then turn it off. The direction of the motor is determined by the value of the camera sensor. If the camera value is 55 (the default), the motor goes forward. Otherwise, the motor goes in the reverse direction.

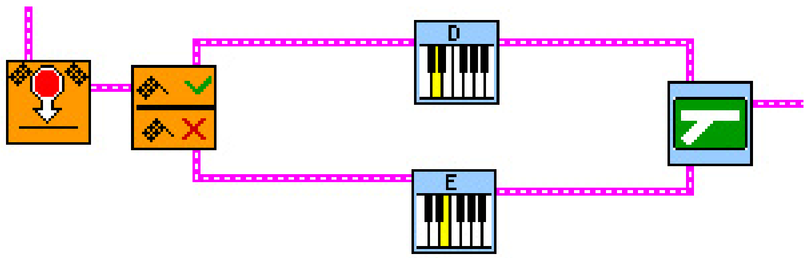
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Event Fork

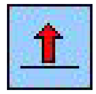
Use this fork to take different actions based on whether an event has occurred or not. Useful if you are monitoring for multiple events and wish to take different actions for different events.

Example

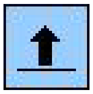
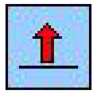
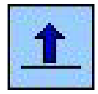
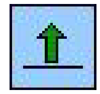
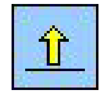


This piece of code (part of a larger program) would play a D note if the red event had occurred. If the red event had not occurred it would play an E note.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Jumps

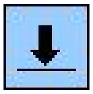
				
Black Jump	**Red Jump**	**Blue Jump**	**Green Jump**	**Yellow Jump**

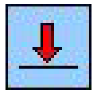
Jumping is useful when certain commands are to be repeated. Each Land command needs to be used in conjunction with its same-colored Jump command.

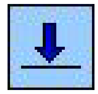
Example

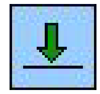
This piece of code will play the fast rising sweep sound until the touch sensor is pressed. The pressing of the touch sensor is set up as a red event.

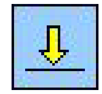
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.


Black Land


Red Land


Blue Land

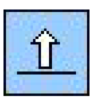

Green Land


Yellow Land

Jumping is useful when certain commands are to be repeated. Place the Land command where you want to start after a jump. Each Land command needs to be used in conjunction with its same-colored Jump command.

Example

This piece of code will play the fast rising sweep sound until the touch sensor is pressed. The pressing of the touch sensor is set up as a red event.

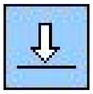

Jumping

Jumping is useful when certain commands are to be repeated. A modifier from 1-20 can be strung into this jump icon to denote which jump the program is executing. A Landing command with the same number needs to be used in conjunction with its corresponding Jumping command.

Example

This piece of code will play the fast rising sweep sound until the touch sensor is pressed. The pressing of the touch sensor is set up as a red event.

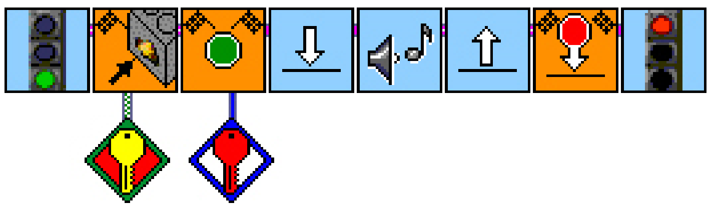
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Landing

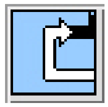
Jumping is useful when certain commands are to be repeated. Place the Landing command where you want to start after a jump. A modifier from 1-20 can be strung to this landing icon to denote which number jump the program is executing. A Landing command with the same number needs to be used in conjunction with its corresponding Jumping command.

Example

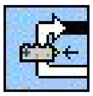


This piece of code will play the fast rising sweep sound until the touch sensor is pressed. The pressing of the touch sensor is set up as a red event.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



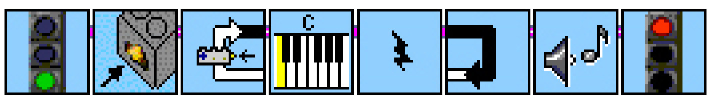
Loops



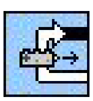
Loop While Touch Sensor is Pushed

Use this to start a loop while the touch sensor is pressed. The loop will end when the touch sensor is released. If the touch sensor is not pressed when the program reaches this command, it skips the loop. To avoid this, add a Wait for Push icon before this icon. An End of Loop icon is needed somewhere after this icon.

Example



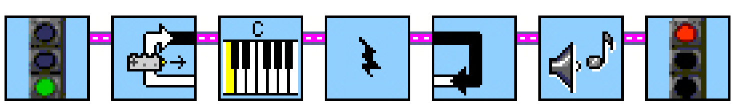
This piece of code would wait until the touch sensor was pushed, then start a loop that would continuously play the note C and a rest while the touch sensor remained pushed. When the touch sensor was released, it would play a sound.



Loop While Touch Sensor is Pushed

Use this to start a loop while the touch sensor is NOT being pressed. The loop will end when the touch sensor is pressed. If the touch sensor is pressed when the program reaches this command, it skips the loop. An End of Loop icon is needed somewhere after this icon.

Example



This piece of code would start a loop that would continuously play the note C and a rest while the touch sensor remained released. When the touch sensor was

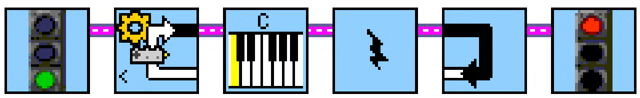
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

pressed, it would play a sound.

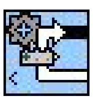
Loop While Number of Touches and Releases is Less Than

Use this to start a loop while the number of touches and releases is less than a certain value. It may be helpful to zero the touch and release sensor before this command. An End of Loop icon is needed somewhere after this icon.

Example



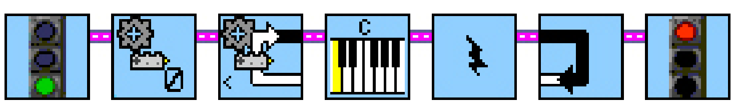
This piece of code would start a loop that would continuously play the note C and a rest until the number of touches and releases was 10 (the default).



Loop While Number of Clicks is Less Than

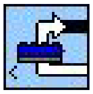
Use these to start a loop while the number of clicks is less than a certain value. It may be helpful to zero the clicks sensor before the command. An End of Loop icon is needed somewhere after this icon.

Example

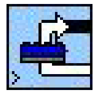


This piece of code zeros the clicks sensor and then starts a loop that would continuously play the note C and a rest until the number of clicks was 10 (the default).

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



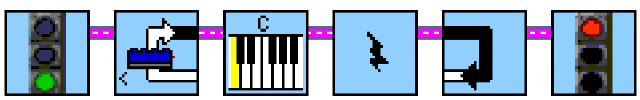
Loop While Light Sensor is Less Than



Loop While Light Sensor is Greater Than

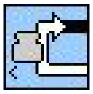
Use these to start a loop while the light sensor value is less than or greater than a certain number. The loop will end when the light sensor value goes above (Less Than icon) or below (Greater Than icon) that number. With respect to the Less Than icon, if the light sensor value is above that number when the program reaches this command, it will skip the loop. With respect to the Greater Than icon, if the light sensor value is below that number when the program reaches this command, it will skip the loop. An End of Loop icon is needed somewhere after each icon.

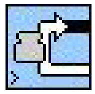
Example



This piece of code starts a loop that would continuously play the note C and a rest until the light sensor value became 55 (the default) or higher.

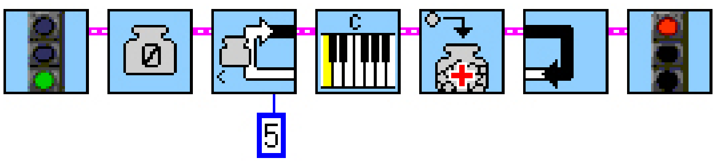
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.


Loop While Container is Less Than


Loop While Container is Greater Than

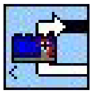
Use these to start a loop while the container value is less than or greater than a certain number. The loop will end when the container value gets to that number or above (Less Than icon) or to that number and below (Greater Than icon). If the container value is above that number (Less Than icon) or below (Greater Than icon) when the program reaches this command, it will skip the loop. An End of Loop icon is needed somewhere after each icon.

Example

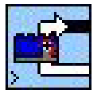


This piece of code would zero the container. Then it would start a loop that plays the note C and adds one to the container. The program plays the note C five times.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



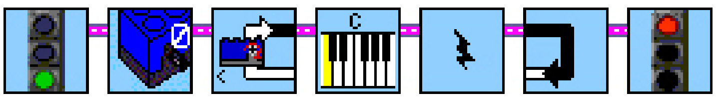
Loop While Rotation Sensor is Less Than



Loop While Rotation Sensor is Greater Than

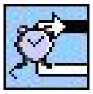
Use these to start a loop while the rotation sensor reads below or above a certain value. The loop will end when the angle sensor gets to or goes beyond (Less Than icon) or to or below (Greater Than icon) that value. It is helpful to use a zero angle sensor icon before this command. An End of Loop icon is needed somewhere after each icon.

Example

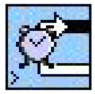


This piece of code zeros the angle sensor then starts a loop that continuously plays the note C and a rest until the rotation sensor reads a value of 16 (one rotation, the default).

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



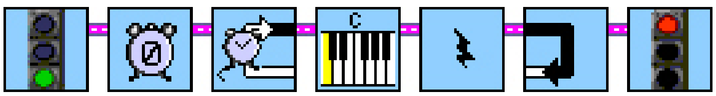
Loop While Timer is Less Than



Loop While Timer is Greater Than

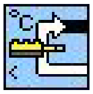
Use these to start a loop while the timer value is less than or greater than a certain number. The loop will end when the timer gets to that number (Less Than icon) or gets to or below the number (Greater Than icon). It is helpful to use a zero timer value before this command. An End of Loop icon is needed somewhere after each icon.

Example

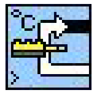


This piece of code zeros the timer then starts a loop that plays the note C and a rest until the time reaches 5 seconds (the default).

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



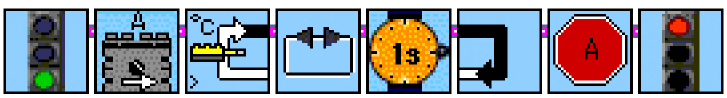
Loop While Celsius is Less Than



Loop While Celsius is Greater Than

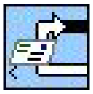
Use these to start a loop while the temperature is below or above a certain value. The loop will end when the temperature rises to or above (Less Than icon) or drops to or below (Greater Than icon) that value. If the temperature is above that value (Less Than icon) or below the value (Greater Than icon) when the program gets to this command, it will skip the loop. An End of Loop icon is needed somewhere after each icon.

Example

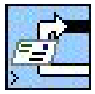


This piece of code turns on motor A in the forward direction then starts a loop which continuously flips the direction of the motor and waits 1 second until the temperature gets to or goes below 30 degrees Celsius. When this happens, motor A is shut off.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



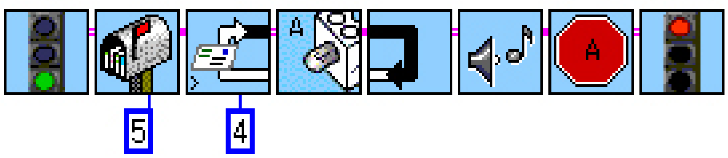
Loop While Mail is Less Than



Loop While Mail is Greater Than

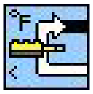
Use these to start a loop while the value of the mail is below or above a certain number. The loop will end when the value of the mail becomes that number or higher (Less Than icon) or when the mail becomes that number or lower (Greater Than icon). With respect to the Less Than icon, if the value of the mail is higher than that number when the program reaches this command, it will skip the loop. When using the Greater Than icon, if the value of the mail is less than that number when the program reaches this command, it will skip the loop. An End of Loop icon is needed somewhere after each icon.

Example

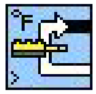


This piece of code fills the mailbox with the value of 5 then starts a loop which turns on lamp A. The loop ends when the RCX receives mail whose value is 4 or less. Then the RCX plays a sound and turns off the lamp.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



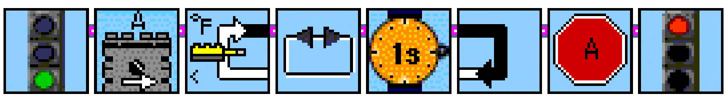
Loop While Fahrenheit is Less Than



Loop While Fahrenheit is Greater Than

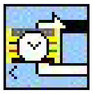
Use these to start a loop while the temperature is below or above a certain value. The loop for the Less Than icon will end when the temperature rises to or above that value. The loop for the Greater Than icon will end when the temperature drops to or below that value. If the temperature is above that value with the Less Than icon or below the value for the Greater Than icon when the program gets to this command, it will skip the loop. An End of Loop icon is needed somewhere after each icon.

Example

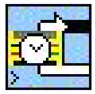


This piece of code turns on motor A in the forward direction then starts a loop which continuously flips the direction of the motor and waits one second until the temperature gets to or goes above 80 degrees Fahrenheit. When this happens, motor A is shut off.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



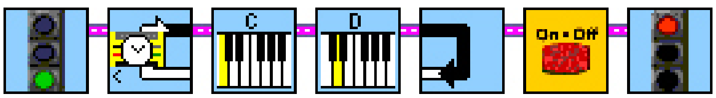
Loop While Clock Value is Less Than



Loop While Clock Value is Greater Than

Use these to start a loop while the RCX clock value is less than or greater than a certain number. The loop will end when the clock goes above (Less Than icon) or below (Greater Than icon) that number. An End of Loop icon is needed somewhere after each icon.

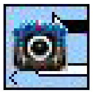
Example



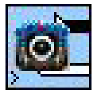
This piece of code plays notes C and D continuously until the RCX clock goes above 1 minute (the default). Then it turns the RCX power off.

educational division

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



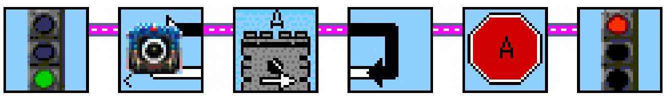
Loop While Camera Sensor is Less Than



Loop While Camera Sensor is Greater Than

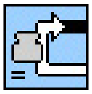
Use this to start a loop while the value of the camera container is less than or greater than some number. The loop will end when the container goes above (Less Than icon) or below (Greater Than icon) that number. An End of Loop icon is needed somewhere after each icon.

Example

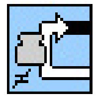


This piece of code starts a loop while the camera container value is less than 1 (the default). It turns on motor A. When the container value is greater than 1, the loop stops and turns off motor A.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



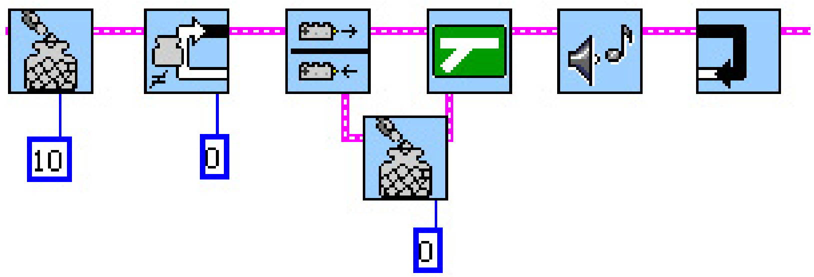
Loop While Container Value Is Equal To



Loop While Container Value Is Not Equal To

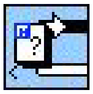
Use this to start a loop while the value of a container is equal to or not equal to a number. An End of Loop icon is needed somewhere after each icon.

Example



This piece of code fills the red container with the number 10. The loop runs (plays a sound) while the red container is not equal to zero. If the touch sensor is pressed the red container is set to zero and the loop stops running.

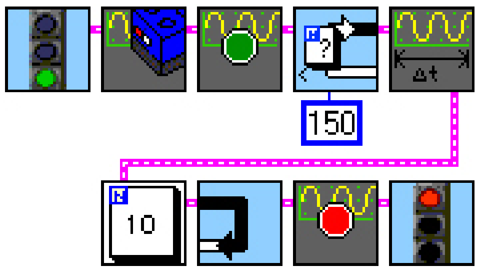
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Loop While Points in Data Set are Less Than

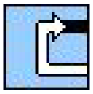
Use in Investigator. Use this to start a loop while the number of data points is less than some number. The loop will end when the number of data points gets to or goes above that number. An End of Loop icon is needed somewhere after this.

Example



This piece of code begins logging light sensor points and enters a loop. Inside the loop it adds a time stamp to the data set and waits for 10 more points. This continues until there are 150 points or more, then the data logging stops.

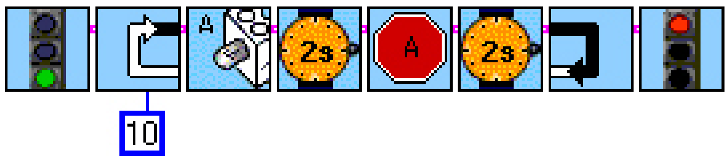
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Start of Loop

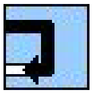
Use this piece of code to start a loop that is going to be repeated a certain number of times. An End of Loop icon is needed somewhere after this icon.

Example



This piece of code would turn lamp A on for 2 seconds and then off for 2 seconds and repeat this 9 more times. In total, the light blinks 10 times.

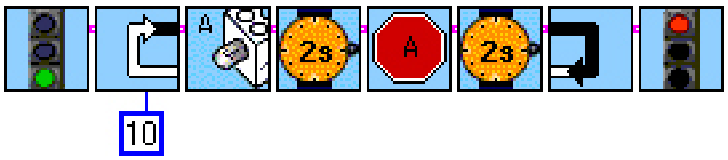
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



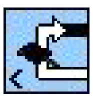
End of Loop

ALWAYS end loops of any kind with this icon.

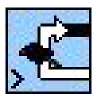
Example



This piece of code would turn lamp A on for 2 seconds and then off for 2 seconds and repeat this 9 more times. In total, the light blinks 10 times.



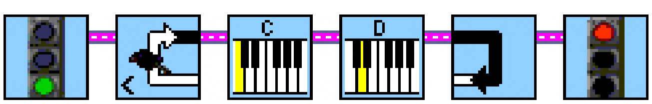
Loop While Value of Generic Sensor is Less Than



Loop While Value of Generic Sensor is Greater Than

Use this to start a loop while the value of the generic sensor adapter is less than or greater than some number. The loop will stop when the value goes above (Less Than icon) or below (Greater Than icon) that number. An End of Loop icon is needed somewhere after each icon.

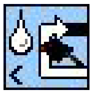
Example



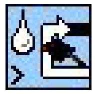
This piece of code will start a loop that plays notes C and D continuously while the value of the generic sensor adapter is less than 2V (the default). When the reading goes above this, the loop will stop.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



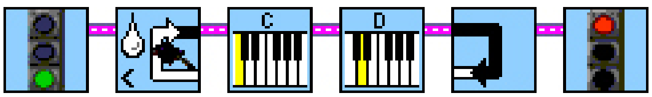
Loop While Humidity is Less Than



Loop While Humidity is Greater Than

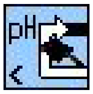
Use these to start a loop while the value of the humidity sensor adapter (LogIT) is less or greater than some number. The loop will stop when the value goes above (Less Than icon) or below (Greater Than icon) that number. An End of Loop icon is needed somewhere after each icon.

Example



This piece of code will start a loop that plays notes C and D continuously while the value of the humidity sensor adapter is less than 50% (the default). When the reading goes above this, the loop will stop.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

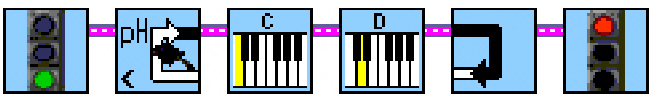


Loop While pH is Less Than

Loop While pH is Greater Than

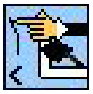
Use these to start a loop while the value of the pH sensor adapter (LogIT) is less than or greater than some number. The loop will stop when the value goes above (Less Than icon) or below (Greater Than icon) that number. An End of Loop icon is needed somewhere after each icon.

Example



This piece of code will start a loop that plays notes C and D continuously while the value of the pH sensor adapter is less than 7 (the default). When the reading goes above this, the loop will stop.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

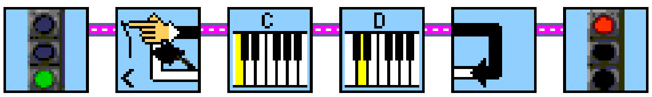


Loop While Pressure is Less Than

Loop While Pressure is Greater Than

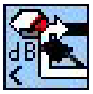
Use these to start a loop while the value of the pressure sensor adapter (LogIT) is less than or greater than some number. The loop will stop when the value goes above (Less Than icon) or below (Greater Than icon) that number. An End of Loop icon is needed somewhere after these icons.

Example



This piece of code will start a loop that plays notes C and D continuously while the value of the pressure sensor adapter is less than 100 kPa (the default). When the reading goes above this, the loop will stop.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

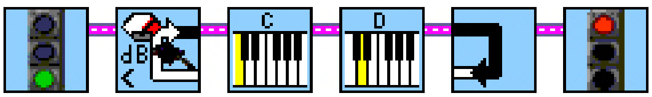


Loop While Sound Level is Less Than

Loop While Sound Level is Greater Than

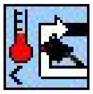
Use these to start a loop while the value of the sound level sensor adapter (LogIT) is less than or greater than some number. The loop will stop when the value goes above that number (Less Than icon) or below (Greater Than icon). An End of Loop icon is needed somewhere after these icons.

Example

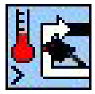


This piece of code will start a loop that plays notes C and D continuously while the value of the sound level sensor adapter is less than 60 dB (the default). When the reading goes above this, the loop will stop.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



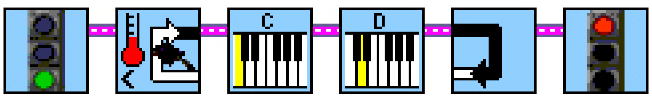
Loop While Temperature (C) is Less Than



Loop While Temperature (C) is Greater Than

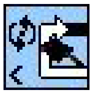
Use these icons to start a loop while the value of the temperature sensor adapter (LogIT) is less than or greater than some number. The loop will stop when the value goes above (Less Than icon) or below (Greater Than icon) that number. An End of Loop icon is needed somewhere after these icons.

Example

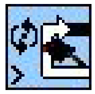


This piece of code will start a loop that plays notes C and D continuously while the value of the temperature sensor adapter is less than 30 degrees Celsius (the default). When the reading goes above this, the loop will stop.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



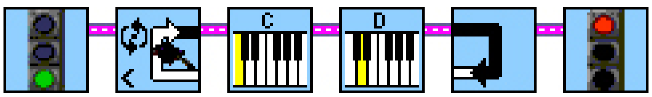
Loop While Position is Less Than



Loop While Position is Greater Than

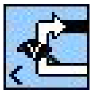
Use these icons to start a loop while the value of the position sensor adapter (LogIT) is less than or greater than some number. The loop will stop when the value goes above (Less Than icon) or below (Greater Than icon) that number. An End of Loop icon is needed somewhere after this icon.

Example

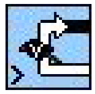


This piece of code will start a loop that plays notes C and D continuously while the value of the position sensor adapter is less than 180 degrees (the default). When the reading goes above this, the loop will stop.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



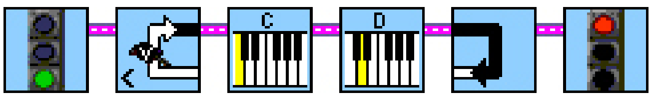
Loop While Voltage Sensor is Less Than



Loop While Voltage Sensor is Greater Than

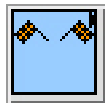
Use these icons to start a loop while the value of the voltmeter (LogIT) is less than or greater than some number. The loop will stop when the value goes above (Less Than icon) or below (Greater Than icon) that number. An End of Loop icon is needed somewhere after both icons.

Example

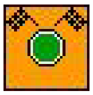


This piece of code will start a loop that plays notes C and D continuously while the value of the voltmeter is less than 2V (the default). When the reading goes above this, the loop will stop.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Events



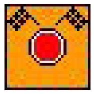
Start Monitoring for an Event

Use this command to begin watching for an event. This command must be used in every event program.

Example

This piece of code sets up a red event that is triggered when the touch sensor is pressed. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the touch sensor is pushed in. This will force the program out of the jump sequence and make it land where the Event Landing is located.

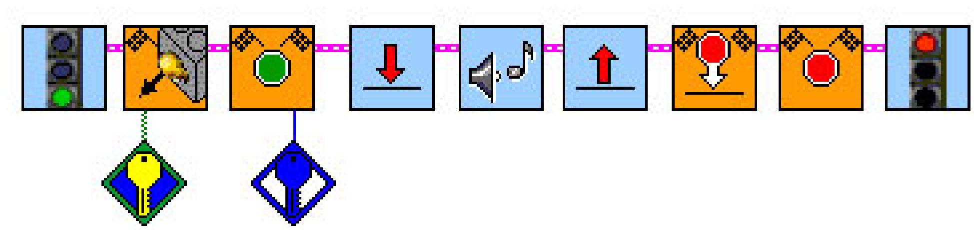
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Stop Event Monitoring

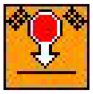
This command stops the program from watching for events.

Example



This piece of code sets up a blue event that is triggered when the touch sensor is touched and released. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the touch sensor is pushed in and released. This will force the program out of the jump sequence and make it land where the Event Landing is located. The program then stops monitoring for events because of the Stop Event Monitoring icon.

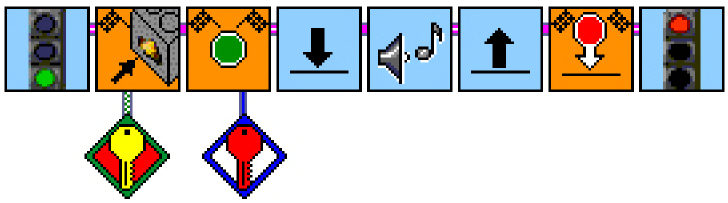
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Event Landing

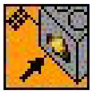
This command marks the place where your program will jump to when an event is triggered. Every event goes to the same land. If you are using more than one event, you will have to use the Event Register Container to determine what event was triggered.

Example



This piece of code sets up a red event that is triggered when the touch sensor is pressed. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the touch sensor is pushed in. This will force the program out of the jump sequence and make it land where the Event Landing is located.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



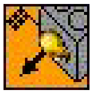
Set Up Pressed Event

This command sets up an event to occur when the touch sensor is pressed. The default waits for a touch sensor pressed connected to port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source, source of information, such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example

This piece of code sets up a red event that is triggered when the touch sensor is pressed. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the touch sensor is pushed in. This will force the program out of the jump sequence and make it land where the Event Landing is located.

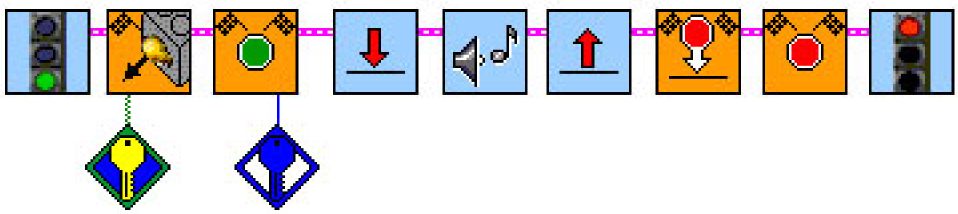
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Released Event

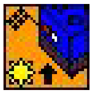
This command sets up an event to occur when the touch sensor is pressed and released. The default waits for a touch sensor pressed connected to port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source, source of information, such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a blue event that is triggered when the touch sensor is released. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the touch sensor is pushed in and released. This will force the program out of the jump sequence and make it land where the Event Landing is located.

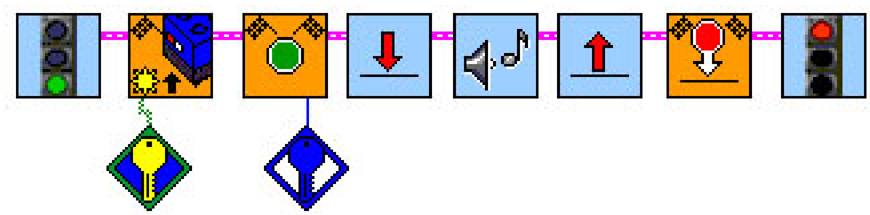
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Light Event

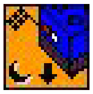
This command sets up an event to occur when the value of the light sensor goes above the threshold. The threshold can be modified with an event modifier. The default is a red event that waits for a light reading higher than 55 from port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source (source of information), such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a blue event that is triggered when the light value goes above the threshold. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the light value goes above the threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

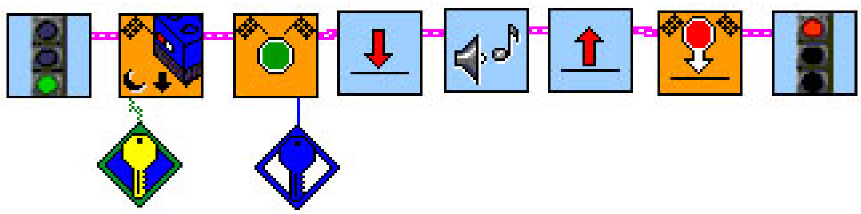
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Dark Event

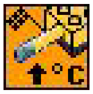
This command sets up an event to occur when the value of the light sensor goes below the threshold. The threshold can be modified with an event modifier. The default is a red event that waits for a light equal to or below 55 from port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source (source of information), such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a blue event that is triggered when the light value goes below the threshold. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the light value goes below the threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

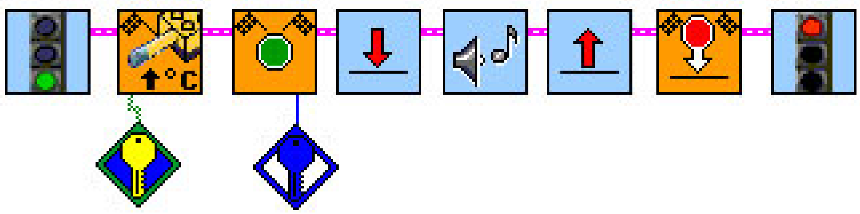
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Increase in Temp (C) Event

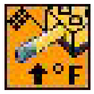
This command sets up an event to occur when the value of the temperature sensor goes above the threshold in Celsius. The threshold can be modified with an event modifier. The default is a red event that waits for a temperature reading above 30(C) from port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source (source of information), such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a blue event that is triggered when the temperature value goes above the threshold in Celsius. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the temperature value goes above the threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

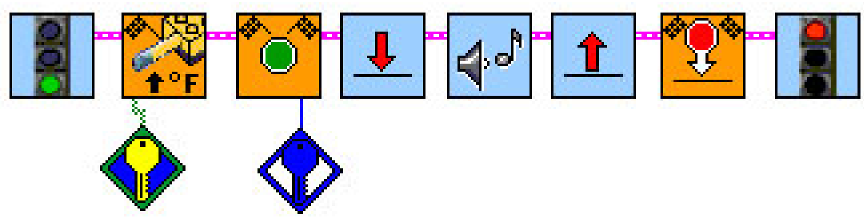
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Increase in Temp (F) Event

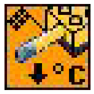
This command sets up an event to occur when the value of the temperature sensor goes above the threshold in Fahrenheit. The threshold can be modified with an event modifier. The default is a red event that waits for a temperature reading above 80(F) from port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source (source of information), such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a blue event that is triggered when the temperature value goes above the threshold in Fahrenheit. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the temperature value goes above the threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

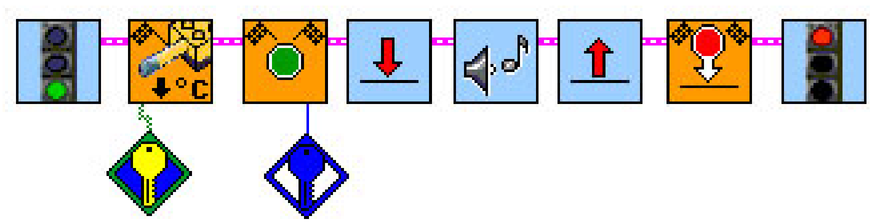
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Decrease in Temp (C) Event

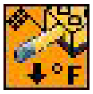
This command sets up an event to occur when the value of the temperature sensor goes below the threshold in Celsius. The threshold can be modified with an event modifier. The default is a red event that waits for a temperature reading below 30(C) from port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source (source of information), such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a blue event that is triggered when the temperature value goes below the threshold in Celsius. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the temperature value goes below the threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

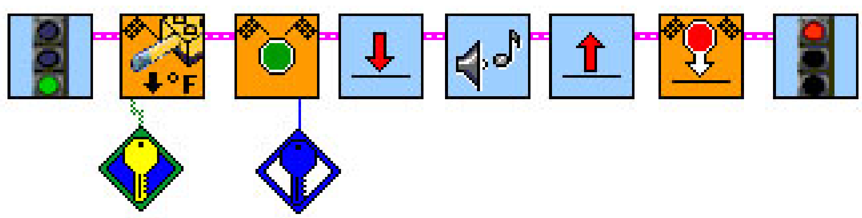
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Decrease in Temp (F) Event

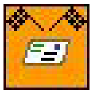
This command sets up an event to occur when the value of the temperature sensor goes below the threshold in Fahrenheit. The threshold can be modified with an event modifier. The default is a red event that waits for a temperature reading below 80(F) from port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source (source of information), such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a blue event that is triggered when the temperature value goes below the threshold in Fahrenheit. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the temperature value goes below the threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

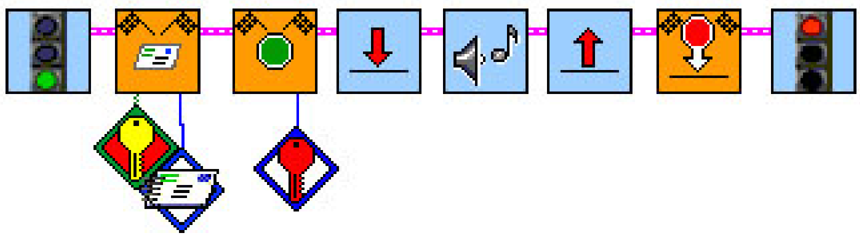
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Mail Event

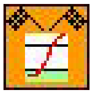
This command sets up an event to occur when the value of the mail equals 10. The default is a red event that waits for the mail to equal 10. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source (source of information), such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a red event that is triggered when the mail value being sent to the RCX equals that of the value of the mailbox. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the value of the mailbox equals the mail the RCX is receiving. This will force the program out of the jump sequence and make it land where the Event Landing is located.

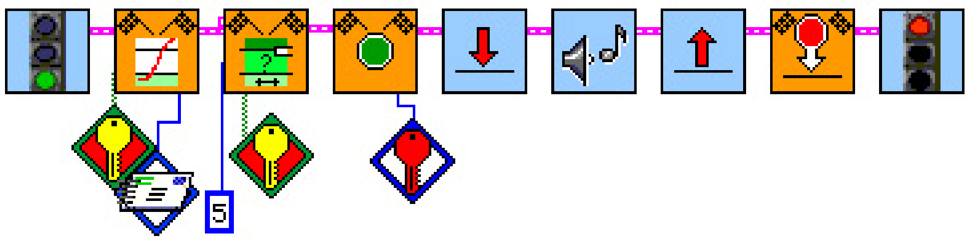
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Enter Low Event

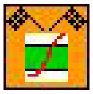
This command sets up an event to be triggered when the value of the Event Source goes below the lower threshold. This command should be followed by the Define Setting Command which is where you define your lower and upper thresholds. Event Sources can come from sensor values from ports 1,2 & 3, container values, mail values, and timer values. If using a sensor value, the sensor should be zeroed prior to setting up the event.

Example



This piece of code defines the thresholds for the Enter Low Event icon. It then sets up the red Enter Low Event and begins monitoring. The program will play a sound over and over again until the value of the mailbox goes below 5. This event will force the program to the place where the Event Landing icon is located. Make sure you define a lower, upper, or normal threshold event before you set them up.

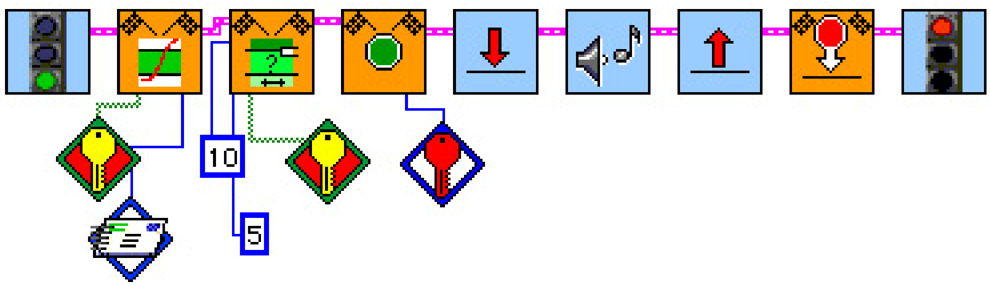
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Enter Normal Event

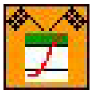
This command sets up an event to be triggered when the value of the Event Source is between the lower and upper threshold. This command should be followed by the Define Setting Command which is where you define your lower and upper thresholds. Event Sources can come from sensor values from ports 1,2,3, container values, mail values, and timer values. If using a sensor value, the sensor should be zeroed prior to setting up the event.

Example



This piece of code defines the thresholds for the Enter Low Event icon. It then sets up the red Enter Low Event and begins monitoring. The program will play a sound over and over again until the value of the mailbox goes above 10 or below 5. This event will force the program to the place where the Event Landing icon is located. Make sure you define a lower, upper, or normal threshold event before you set them up.

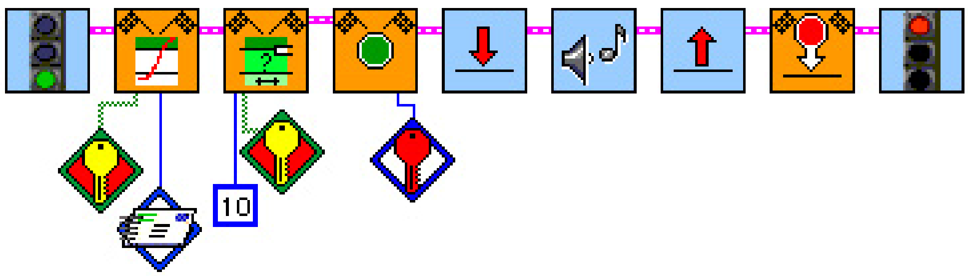
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Enter Hi Event

This command sets up an event to be triggered when the value of the Event Source is above the upper threshold. This command should be followed by the Define Setting Command which is where you define your lower and upper thresholds. Event Sources can come from sensor values from ports 1,2 & 3, container values, mail values, and timer values. If using a sensor value, the sensor should be zeroed prior to setting up the event.

Example



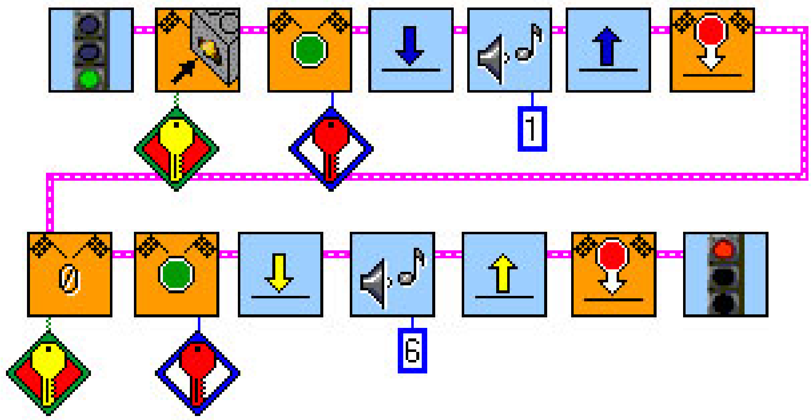
This piece of code defines the thresholds for the Enter Low Event icon. It then sets up the red Enter Low Event and begins monitoring. The program will play a sound over and over again until the value of the mailbox goes above 10. This event will force the program to the place where the Event Landing icon is located. Make sure you define a lower, upper, or normal threshold event before you set them up.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Reset Event

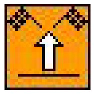
This command resets monitoring for an event. The default is to reset the red event with the value of port 1. Event Sources can come from sensor values from ports 1,2 & 3, container values, mail values, and timer values.

Example



This piece of code sets up a red event that is triggered when the touch sensor is pressed. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the touch sensor is pushed in. This will force the program out of the jump sequence and make it land where the Event Landing is located. The program then resets all events and begins monitoring for the red event to happen again. A different sound will keep playing until the touch sensor is pushed again.

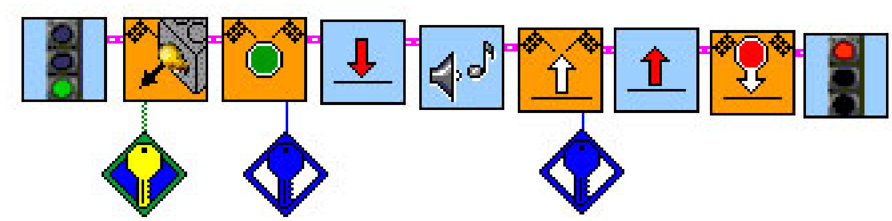
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Force An Event

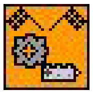
This command forces an event to be triggered. It basically overrides the system in order to make an event occur. String in the yellow, blue, or red event modifier in order to force that particular event.

Example



This piece of code sets up a blue event that is triggered when the touch sensor is touched and released. The Monitor Event icon begins monitoring for such an event to occur. In this case, the Force Event icon will the program into thinking that the event has really happened (even thought the touch sensor was never pressed) and force it out of the jump loop.

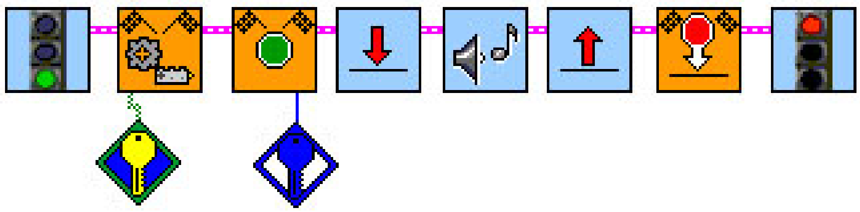
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Clicks Event

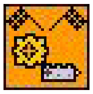
This command sets up an event to occur when the number of clicks equals the clicks threshold. The default is a red event that waits for 10 clicks on port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source, source of information, such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a blue event that is triggered when the number of clicks equals the clicks threshold. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the number of clicks equals the clicks threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

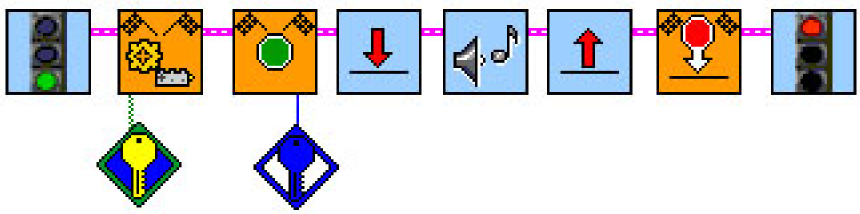
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Touch and Release Event

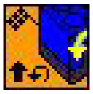
This command sets up an event to occur when the number of touch and releases equals the touch and release (TR) threshold. The default is a red event that waits for 10 touch and releases on port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source, source of information, such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a blue event that is triggered when the number of touch and releases of the touch sensor equals the clicks threshold. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the number of touch and releases of the touch sensor equals the clicks threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

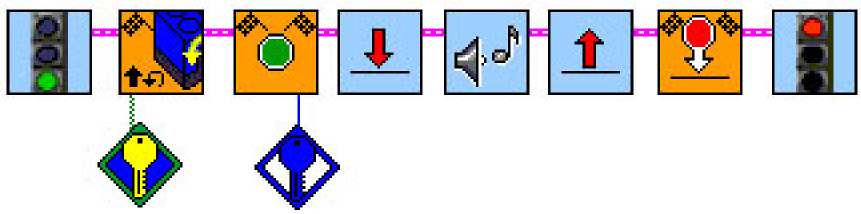
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Increase in Rotation Event

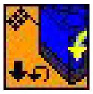
This command sets up an event to occur when the value of the rotation sensor goes above the threshold. The threshold can be modified with an event modifier. The default is a red event that waits for a rotation reading of 16 (rotation sensor is in 16ths of a rotation) from port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source (source of information), such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a blue event that is triggered when the rotation value goes above the threshold. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the rotation value goes above the threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

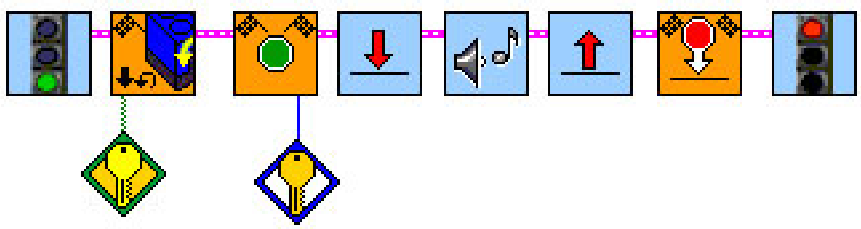
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Decrease in Rotation Event

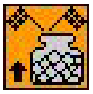
This command sets up an event to occur when the value of the rotation sensor goes below the threshold. The threshold can be modified with an event modifier. The default is a red event that waits for a rotation reading equal to or below 16 (rotation sensor is in 16ths of a rotation) from port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source (source of information), such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a yellow event that is triggered when the rotation value goes below the threshold. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the rotation value goes below the threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

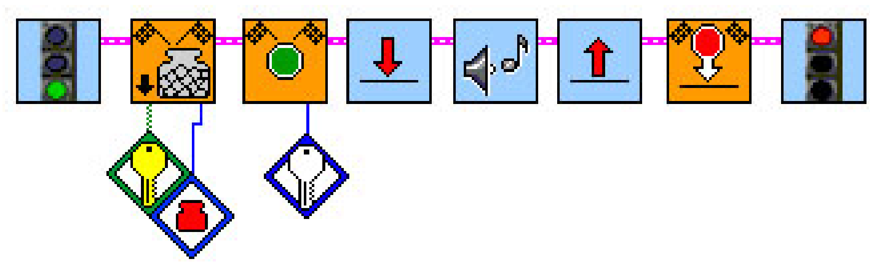
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Increase in Container Event

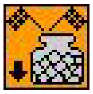
This command sets up an event to occur when the value of the container goes above the threshold. The threshold can be modified with an event modifier. The default is a red event that waits for a red container value greater than 1 from port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source (source of information), such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a generic event that is triggered when the red container value goes above the threshold. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the red container value goes above the threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

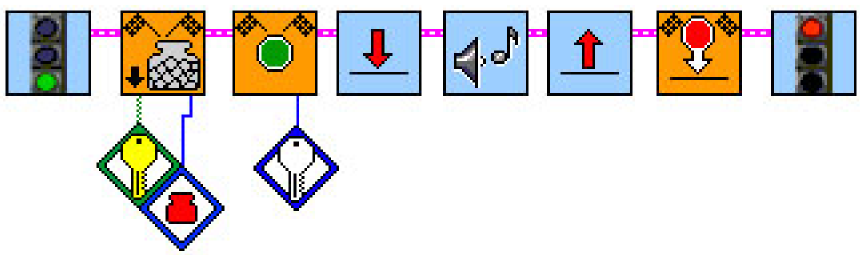
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Up Decrease in Container Event

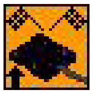
This command sets up an event to occur when the value of the container goes below the threshold. The threshold can be modified with an event modifier. The default is a red event that waits for a red container value equal to or less than 1 from port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source (source of information), such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a generic event that is triggered when the red container value goes below the threshold. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the red container value goes below the threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

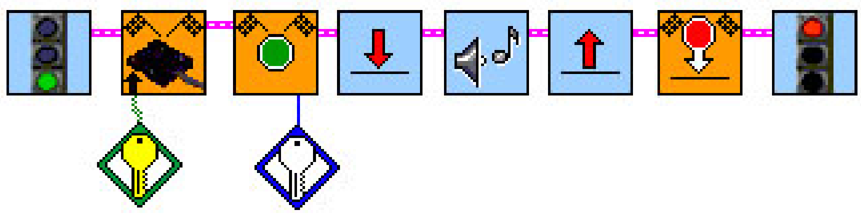
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Generic Up Event

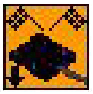
This command sets up an event to occur when the value of the sensor goes above the threshold. The threshold can be modified with an event modifier. The default is a red event that waits for a sensor value greater than 2V from port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source (source of information), such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a generic event that is triggered when the generic sensor value goes above the threshold. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the generic sensor value goes above the threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

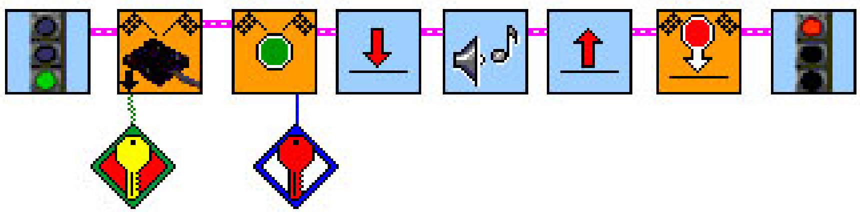
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set Generic Down Event

This command sets up an event to occur when the value of the sensor goes below the threshold. The threshold can be modified with an event modifier. The default is a red event that waits for a sensor value equal to or less than 2V from port one. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source (source of information), such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



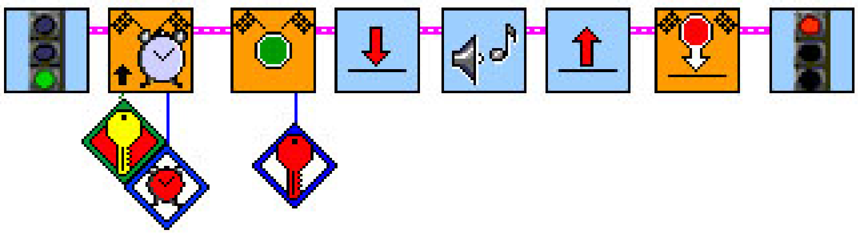
This piece of code sets up a red event that is triggered when the generic sensor value goes below the threshold. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the generic sensor value goes below the threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Set Up Increase in Timer Event

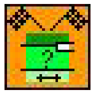
This command sets up an event to occur when the value of the timer goes above the threshold. The default is a red event that waits for a timer value of one second. This command must appear before the Begin Monitoring command in your program. You can string in a yellow, blue, red, or generic modifier to have several events. You can also string in an event source, source of information, such as sensor values of ports 1, 2, & 3, container values, or mail values.

Example



This piece of code sets up a red event that is triggered when the value of the red timer goes above the threshold. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the red timer value goes above the threshold. This will force the program out of the jump sequence and make it land where the Event Landing is located.

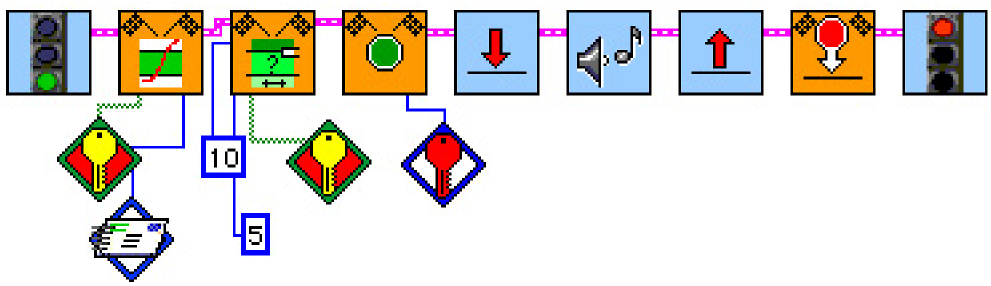
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Define Event

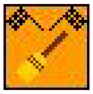
This command is used for defining the thresholds for the Enter Low, Normal, and High Events commands. If a certain specification is not being used, it will be ignored. For example, the upper threshold will be ignored if you only set up an Enter Low Event command. You can string in an event modifier to determine what event you are dealing with. Also, you can string in a duration modifier which sets a limit of the length of time a condition must be occurring to count as an event.

Example



This piece of code defines the thresholds for the Enter Low Event icon. It then sets up the red Enter Low Event and begins monitoring. The program will play a sound over and over again until the value of the mailbox goes above 10 or below 5. This event will force the program to the place where the Event Landing Icon is located. Make sure you define a lower, upper, or normal threshold event before you set them up.

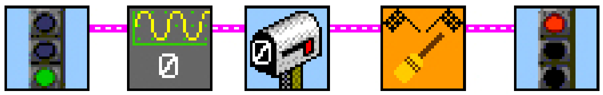
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Clear All Events

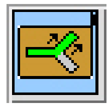
This command clears ALL events. Tasks that are actively monitoring or awaiting for events are not alerted to this fact.

Example

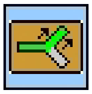


This piece of code empties the data buffer, zeros the mailbox, and clears all events currently configured on the RCX..

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



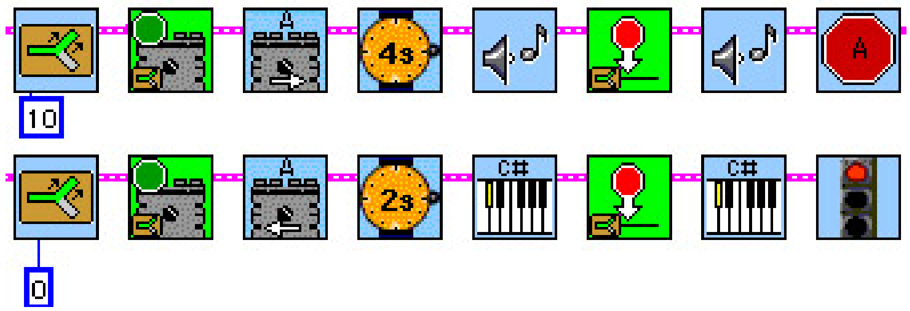
Task Priority



Task Priority

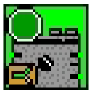
Use this icon to assign a task priority over other tasks. The highest priority is the default, 0. The lowest is 255. An access monitoring command should follow this icon

Example



This piece of code assigns the highest priority to the bottom task so that motor A will go in the backwards direction for 2 seconds. The top task will give control of port A to the bottom task and jump immediately to the Access land and then beeps.

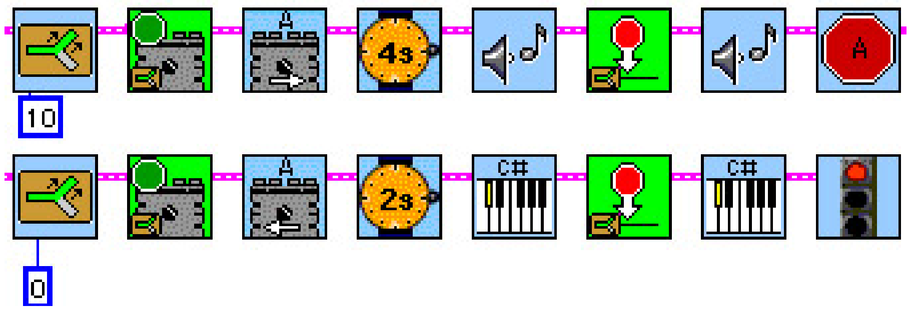
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Start Monitoring for Output Access Control

Use this to monitor ports for control over outputs. If there are multiple tasks each wanting to use the same output port at the same time, this command allows the task with highest priority to use that output. Other tasks will jump to the Access Landing

Example



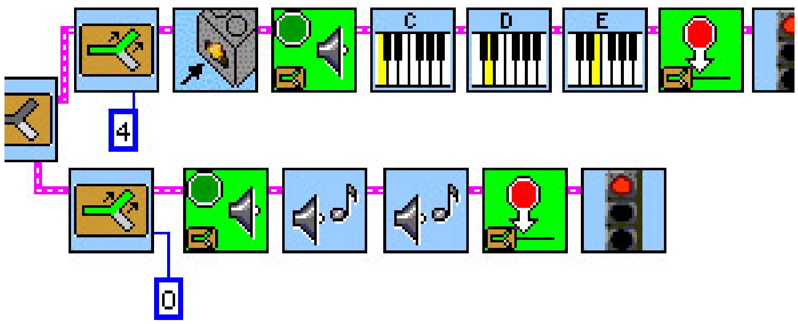
This piece of code assigns the highest priority to the bottom task so that motor A will go in the backwards direction for 2 seconds. The top task will give control of port A to the bottom task and jump immediately to the Access land and then beeps.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Start Monitoring for Sound Access Control

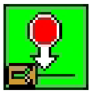
Use this to monitor ports for control over sound outputs. If there are multiple tasks wanting to use sound at the same time, this command allows the task with highest priority access to sound. Other tasks will jump to the Access Landing..

Example



This piece of code assigns the highest priority to the bottom task which plays two beeps. If the touch sensor is pressed before the bottom task has finished the program will jump immediately to the Access Land in the top task..

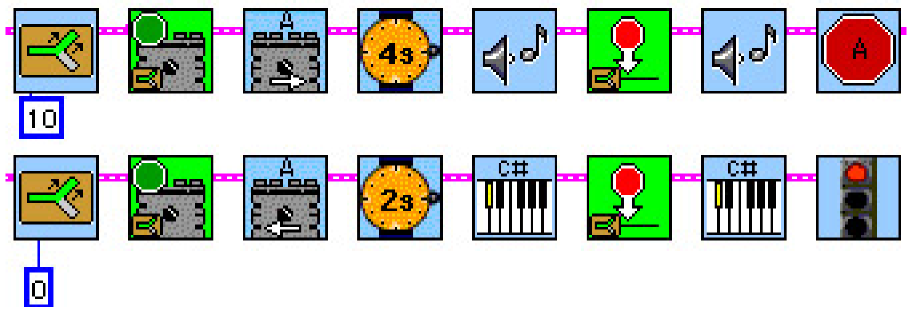
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Access Control Landing

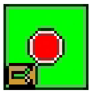
Use this in tasks of lower priority after an access monitoring command to tell the task where to jump to if a task of higher priority comes in conflict with it

Example



This piece of code assigns the highest priority to the bottom task so that motor A will go in the backwards direction for 2 seconds. The top task will give control of port A to the bottom task and jump immediately to the Access land and then beeps.

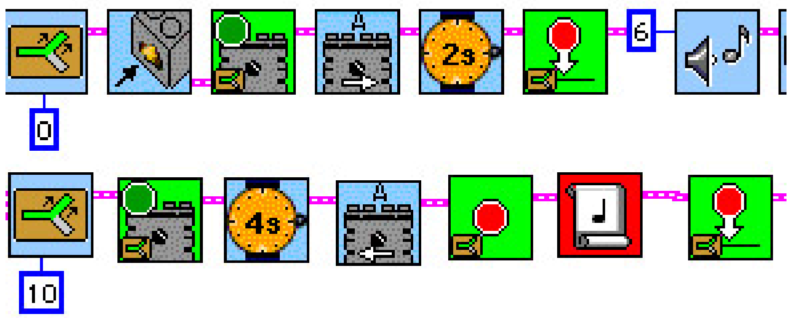
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Stop Access Control Monitoring

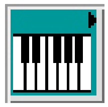
Use this to stop access monitoring.

Example

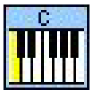


This piece of code assigns the highest priority to the top task so that motor A will go in the forward direction after the touch sensor is pressed. The bottom task will give control of port A to the top task and turn on lamp B if the touch sensor is pressed BEFORE the bottom task reaches the stop access monitoring. Once the red scroll has started playing, the bottom task will continue even if the touch sensor is pressed.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Music

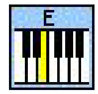


C

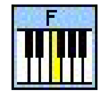
C#

D

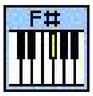
D#



E



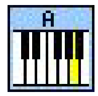
F



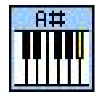
F#

G

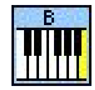
G#



A



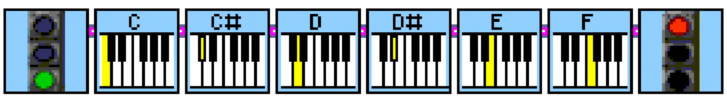
A#



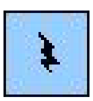
B

Use these icons to play individual notes.

Example



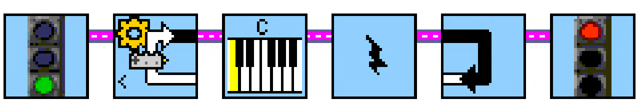
This piece of code plays C, C#, D, D#, E and F as quarter notes in the standard scale.



Use this to play a rest, or pause.

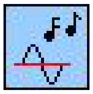
Rest

Example



This piece of code would start a loop that would continuously play the note C and a rest (the default is a quarter rest) until the number of touches and releases was 10 (the default).

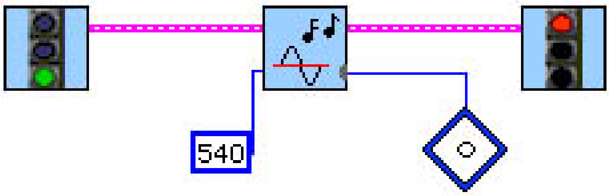
educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



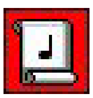
Play Any Note

Use this to play any note by choosing a frequency and duration.

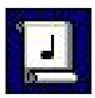
Example



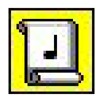
This piece of code plays a whole note A the frequency is specified by the constant (540) wired in.



Red Scroll



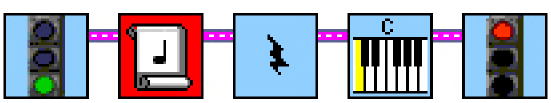
Blue Scroll



Yellow Scroll

Use this to play the music stored on the scrolls. If you haven't stored any music to the scrolls in piano player, the default songs will play. The red scroll plays Frere Jacques. The blue scroll plays Row, Row, Row your Boat. The yellow scroll plays Twinkle, Twinkle, Little Star.

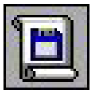
Example



This piece of code plays the music on the red scroll then a quarter rest and the note C.

educational division

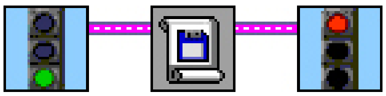
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



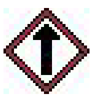
Load Scroll From File

Use this to play a song file you have saved.

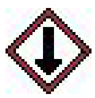
Example



This piece of code brings up a dialog box for you to choose a song. Then, it plays that song.



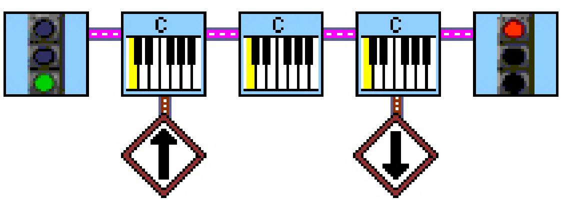
Up an Octave



Down an Octave

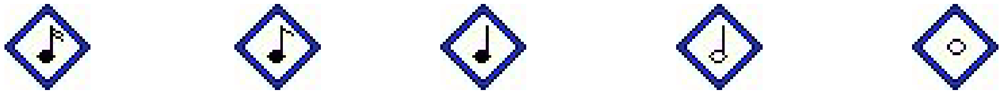
Use this to play a note in octaves above or below the standard octave. String several of these together to play several octaves above or below standard. These icons are used as modifiers.

Example



This piece of code plays C one octave above standard, then in the standard octave, then one octave below standard.

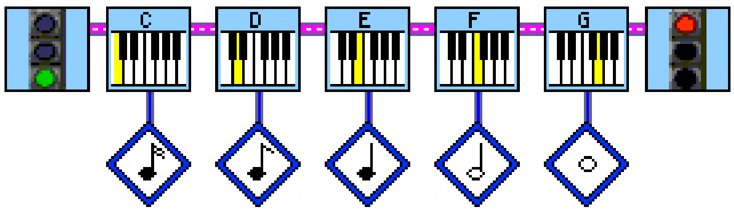
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Sixteenth Note | **Eighth Note** | **Quarter Note** | **Half Note** | **Whole Note**

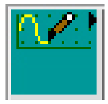
Use these to play notes. These icon are used as modifiers.

Example

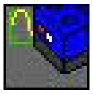


This piece of code plays C as a sixteenth note, D as an eighth note, E as a quarter note, F as a half note, and G as a whole note in the standard scale.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



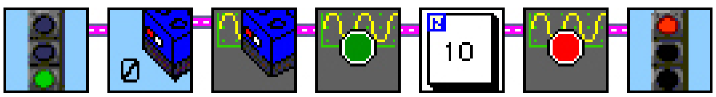
Investigator



Initialize Light Sensor Logging

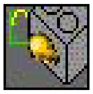
Use this command to collect light data using the light sensor. The data returned is a number between 1 (dark) and 100 (bright).

Example



This piece of code resets the light sensor and then logs 10 light data points.

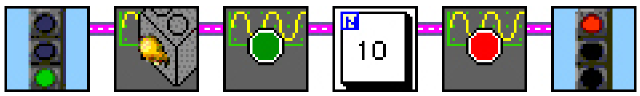
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Initialize Touch Sensor Logging

Use this command to collect data from the touch sensor. If pressed, the sensor returns a 1, otherwise, it returns a zero.

Example



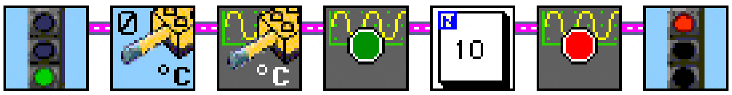
This piece of code would take 10 data points from the touch sensor.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

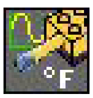
Initialize Temperature Sensor Logging

Use this command to collect temperature data in Celsius from the temperature sensor.

Example



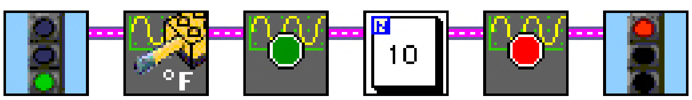
This piece of code resets the temperature sensor and then logs 10 temperature data points.



Initialize Temperature Sensor Logging

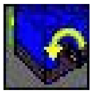
Use this command to collect temperature data in Fahrenheit from the temperature sensor.

Example



This piece of code would take 10 data points from the temperature sensor.

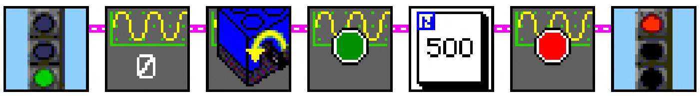
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



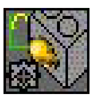
Initialize Rotation Sensor Logging

Use this command to collect rotation data from the rotation sensor. Data is in sixteenths of a rotation so a reading of 16 is equal to one rotation. Rotation can either be in either direction.

Example



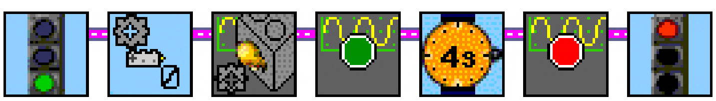
This piece of code clears the data memory, then logs 500 points of rotation sensor data.



Initialize Click Sensor Logging

Use this command to collect data from the click sensor. It might be helpful to zero the click sensor at the beginning of the code.

Example



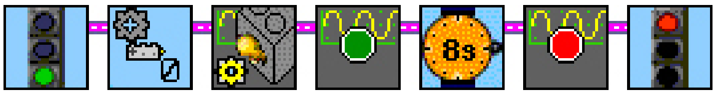
This piece of code zeros the click sensor. Then it logs click sensor data for 4 seconds and stops.

educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Initialize Touch and Release Sensor Logging

Use this command to collect data from the touch and release sensor. It might be helpful to zero the touch and release sensor at the beginning of the code.

Example

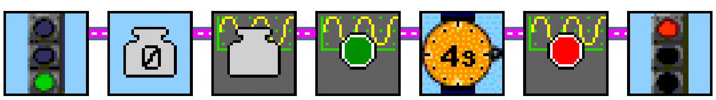


This piece of code zeros the touch and release sensor. Then it logs touch and release sensor data for 8 seconds and stops.

Initialize Container Logging

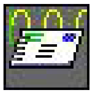
Use this command to collect data from a container.

Example



This piece of code empties the container, then logs container data for 4 seconds and then stops.

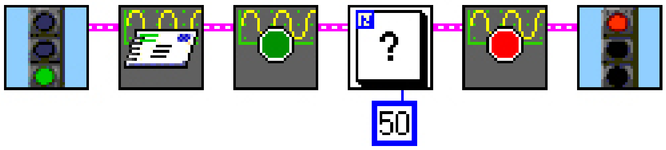
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Initialize Mail Logging

Use this command to collect data from the mailbox.

Example

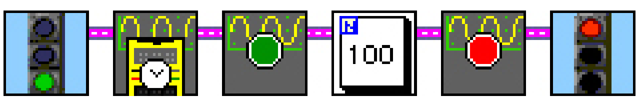


This piece of code would take 50 data points from the mailbox.

Initialize Clock

Use this command to initialize logging of the RCX clock.

Example



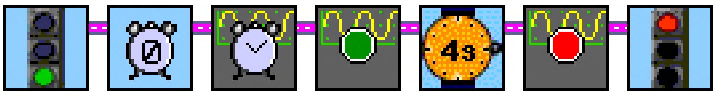
This piece of code logs 100 data points from the clock.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Initialize Timer Logging

This initializes logging of the regular timer (timer that increases in .1 second increments versus the fast timer which increases in .01 second increments).

Example

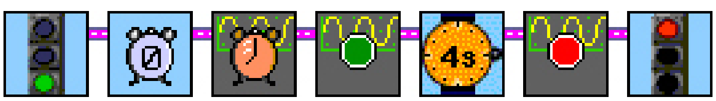


This piece of code initializes the red (default) timer to zero, initializes regular timer logging, starts data logging every .1 seconds, logs for 4 seconds, and then stops data logging.

Initialize Fast Timer Logging

This initializes logging of the fast timer (timer that increases in .01 second increments versus the regular timer that increases in .1 second increments).

Example



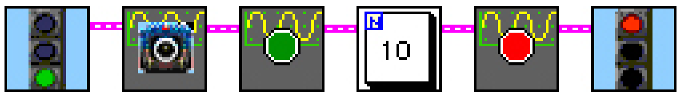
This piece of code initializes the red (default) timer to zero, initializes fast timer logging, starts data logging every .01 seconds, logs for 4 seconds, and then stops data logging.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

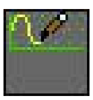
Initialize Camera Sensor Logging

Use this command to collect data from the camera sensor.

Example



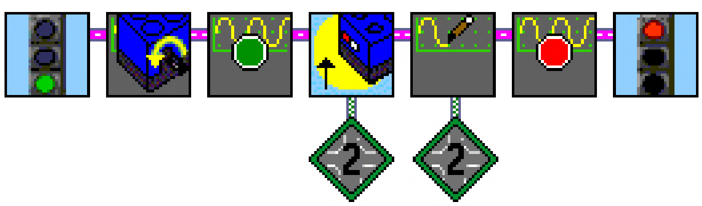
This piece of code would take 10 data points from the camera sensor.



Write Data Point to Data Set

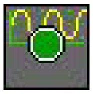
Use this command to add a data point to a data set. MUST have both Initialize Logging and Start Logging commands before this command.

Example



This piece of code would log rotation sensor data until the light sensor reading increases by 5. It would then add the light sensor reading to the data set and stop logging data.

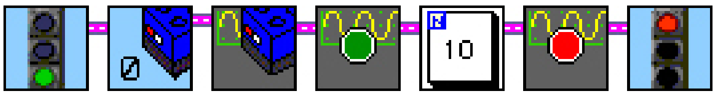
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Start Data Logging

Use this to start taking data points. There MUST be an initialize command before this.

Example

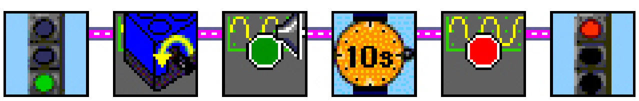


This piece of code resets the light sensor and then logs 10 light data points.

Start Data Logging with Clicks

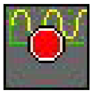
Use this command to log click data along with another type of data. An Initialize Logging command must come before this icon. This command WILL start logging data.

Example



This piece of code would log rotation data with clicks for 10 seconds and then stop.

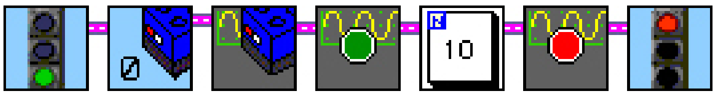
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Stop Logging

Use this to stop taking data points.

Example

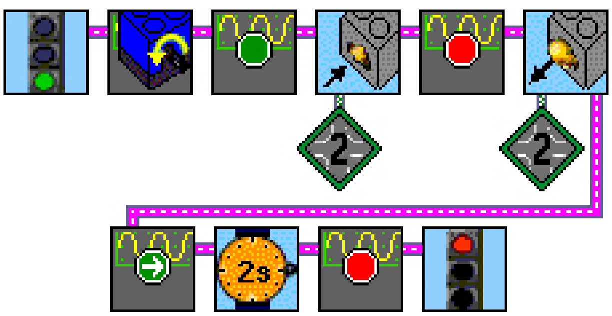


This piece of code resets the light sensor and then logs 10 light data points.

Resume Logging

Use this command to start data logging again after it has been stopped.

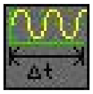
Example



This piece of code would log data from the rotation sensor until the touch sensor was pushed in. Then it would stop until the touch sensor was released. Then it would resume data logging for 2 seconds.

educational division

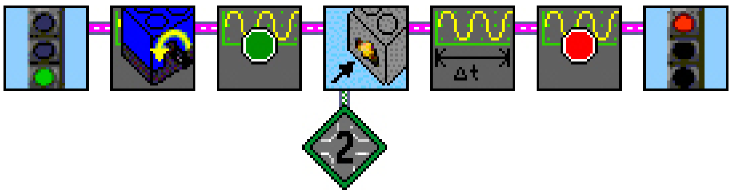
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



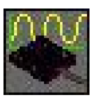
Log Time Stamp

Use this command to find out how much time has passed since you began taking data.

Example



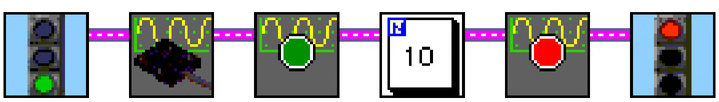
This piece of code would log data from the rotation sensor until the touch sensor on port 2 is pressed. Then it would log how much time has passed since data logging began, and stop logging data.



Initialize Generic Sensor Logging

Use this command to take voltage data from a generic powered sensor.

Example



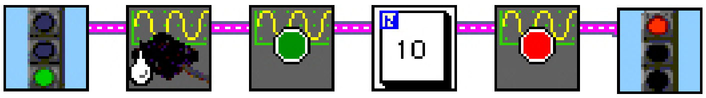
This piece of code would take 10 data points from a generic powered sensor.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Initialize Humipro Sensor Logging

Use this command to take humidity data from a humidity sensor adapter (LogIT). The data is returned as a percentage between 1 and 100.

Example

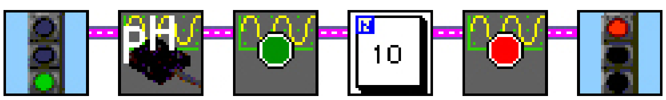


This piece of code would take 10 data points from the humidity sensor.

Initialize pH Sensor Logging

Use this command to take pH data from the pH sensor (LogIT). The data is returned as a number between 1 and 14.

Example



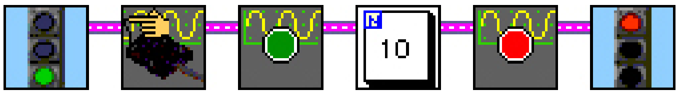
This piece of code would take 10 data points from the pH sensor.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Initialize Pressure Sensor Logging

Use this command to take pressure data from the pressure sensor adapter (LogIT). The data is returned in kPa.

Example

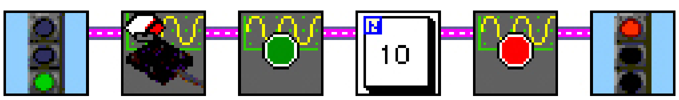


This piece of code would take 10 data points from the pressure sensor.

Initialize Sound Sensor Logging

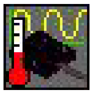
Use this command to take sound data from the sound level sensor (LogIT). The data is returned in dB.

Example



This piece of code would take 10 data points from the sound level sensor.

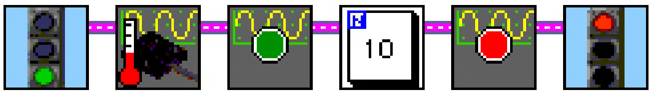
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Initialize proTemp Sensor Logging

Use this command to take temperature data in Celsius from the temperature sensor adapter (LogIT). You cannot take data in Fahrenheit with this sensor.

Example

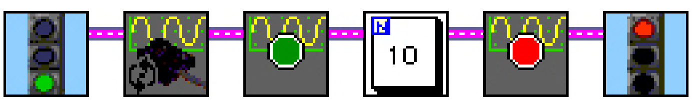


This piece of code would take 10 data points from the temperature sensor.

Initialize Position Sensor Logging

Use this command to take position data in degrees from the position sensor adapter (LogIT).

Example



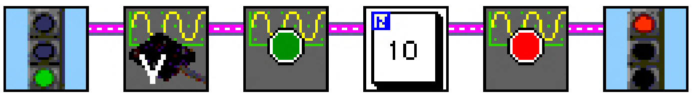
This piece of code would take 10 data points from the position sensor.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Initialize Voltmeter Sensor Logging

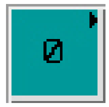
Use this command to take voltage data from the voltmeter (LogIT).

Example

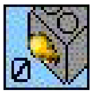


This piece of code would take 10 data points from the voltmeter.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



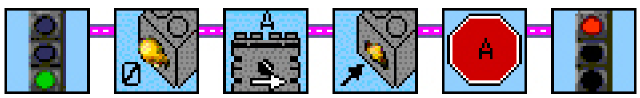
Reset



Zero Touch Sensor

Use this to reset the touch sensor. Helpful if taking touch sensor data.

Example

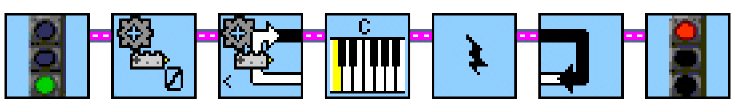


This piece of code resets the touch sensor then turns on motor A. When the touch sensor is pressed, motor A is shut off. This program could run without the reset, however, the reset works to ensure accurate touch sensor data.

Zero Clicks Sensor

Use this to reset the click count sensor. Helpful if taking clicks data.

Example



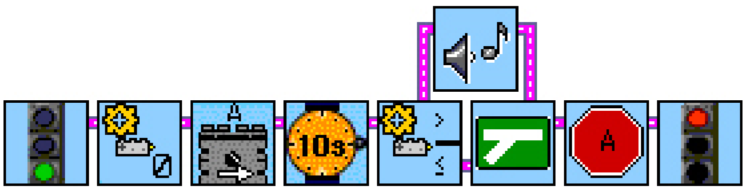
This piece of code zeros the clicks sensor and then starts a loop that would continuously play the note C and a rest until the number of clicks was 10 (the default).

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Zero Touch and Release Sensor

Use this to reset the touch and release sensor. Helpful if taking touch and release data.

Example

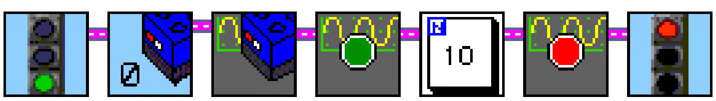


This piece of code turns on motor A for 10 seconds, and then shuts it off. If after 10 seconds the touch sensor has been touched and released more than 10 times (the default), it plays a sound.

Zero Light Sensor

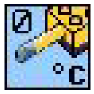
Use this to reset the light sensor. Helpful if taking light sensor data.

Example



This piece of code resets the light sensor and then logs 10 data points.

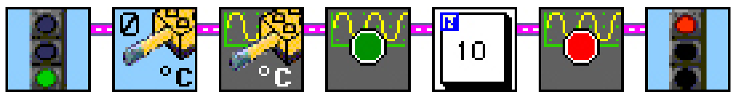
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



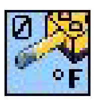
Zero Temperature (C) Sensor

Use this to reset the temperature sensor. Helpful if taking temperature data.

Example



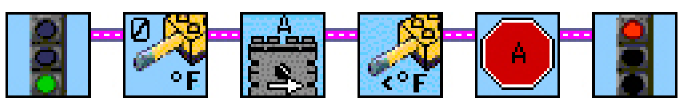
This piece of code resets the temperature sensor and then logs 10 data points.



Zero Temperature (F) Sensor

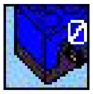
Use this to reset the temperature sensor. Helpful if taking temperature data.

Example



This piece of code resets the temperature sensor. Then it turns on motor A. When the temperature falls below 80 degrees Fahrenheit (the default), motor A is shut off.

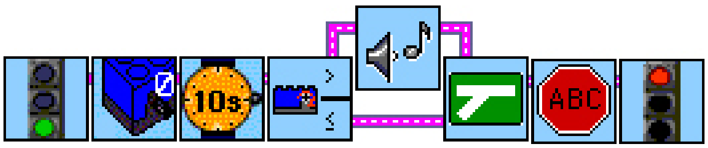
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Use this to reset the angle sensor. Helpful if taking angle or rotation data.

Zero Angle Sensor

Example

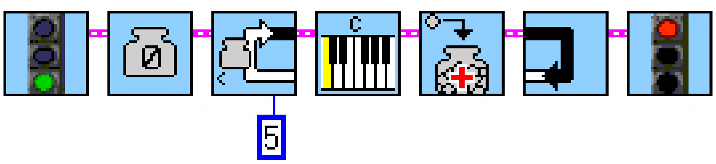


This piece of code zeros the angle sensor then waits 10 seconds. If at that point the rotation sensor reads greater than 1 rotation (a reading of 16, the default), then the RCX plays a sound.

Use this to empty a container. Helpful whenever you are placing new values in a container.

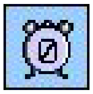
Zero Container

Example



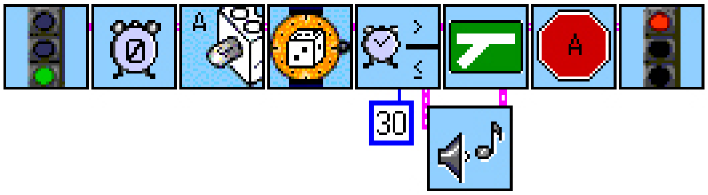
This piece of code would zero the container. Then it would start a loop that plays the note C and adds one to the container. The loop would end when the container reaches 5. In total, it will play the note C five times.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.


Zero Timer

Use to reset timer. Helpful whenever you use the timers.

Example

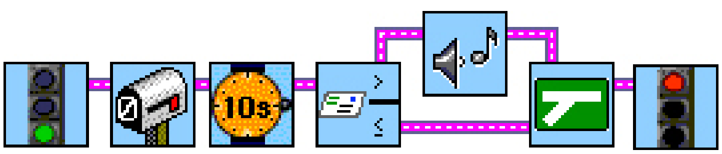


This piece of code turns on lamp A for a random amount of time, and then turns it off. If the time is less than 3 seconds (the timer measures tenths of seconds, so the modifier 30 is equivalent to 3 seconds), the RCX plays a sound.


Zero Mailbox

Use this to empty the mailbox. Helpful when you are planning to receive new mail or want to check for new mail.

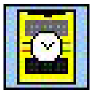
Example



This piece of code would empty the mail box. After 10 seconds, it checks to see if it has received mail whose value is greater than 1 (the default). If it has, the RCX plays a sound.

LEGO educational division

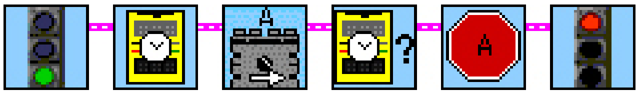
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



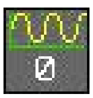
Zero Clock

Use this to reset the RCX clock.

Example



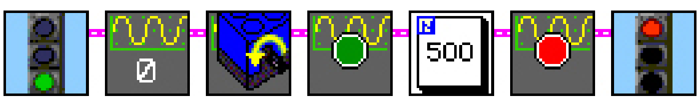
This piece of code resets the clock to zero and turns on motor A. When the clock gets to 1 minute (the default), motor A turns off.



Clear Data Logging Memory

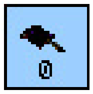
Clears data logging memory. Helpful if you're taking lots of data or have a lengthy program.

Example



This piece of code clears the data memory, then logs 500 points of rotation sensor data.

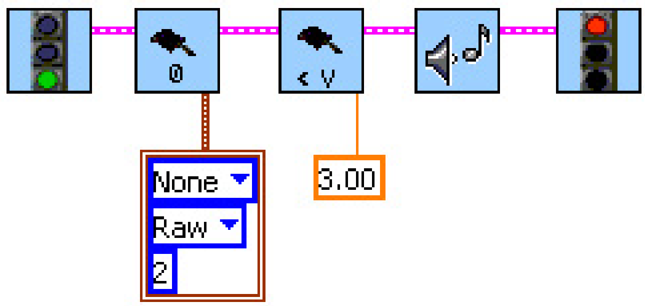
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



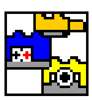
Configure Generic Sensor

Allows you to programmatically configure the type, mode and slope of the Generic Sensor (settings of which are specified in 101Generic.sa sensor file) Calibrations must be altered in the actual sensor file.

Example



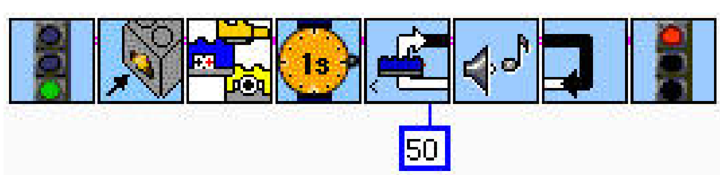
This piece of code sets up a sensor with as type 'None', mode 'Raw', with a slope of 2. It waits for the value to read below 3 and then beeps.



Reset Sensor Ports

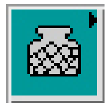
Allows you to reset the sensor ports so you can use two different sensors on the same port. Note: This should be used carefully by advanced users and can only be implemented with compatible sensors.

Example



This piece of code waits for the touch sensor to be pressed and the resets the sensor ports, waits for one second, and then beeps until the light value is greater than 50.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

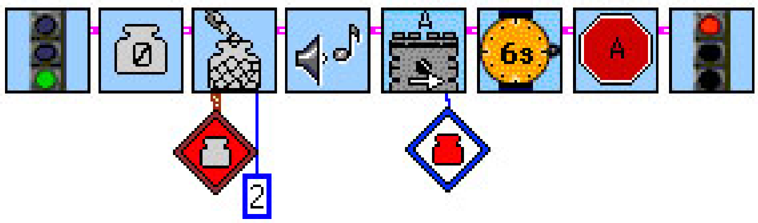


Containers

Fill Container

Use this command to fill a container with a value of your choice which can be controlled by a modifier. The default is to set the Red Container to a value of 1.

Example

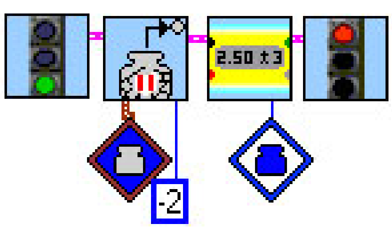


This piece of code first zeroes the container, then fills the Red Container with a value of 2. A sound is played, and then the motor is run at the speed of the Red Container, which is 2.

ABS Container

Use this command to fill a container with the absolute value of a number. The default is to put the absolute value of zero into the Red Container.

Example

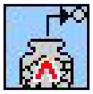


This piece of code takes absolute value of whatever is put into the container. In this case, the absolute value of -2

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

is put into the Blue Container, and the RCX will display 2.

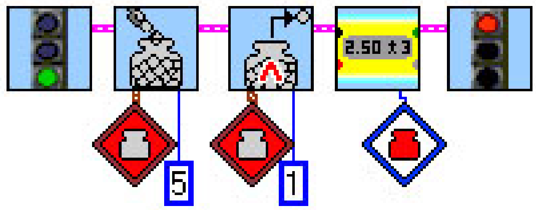
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



AND Container

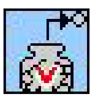
This command ANDs a bitwise value to the bitwise value of the container. This is NOT a Boolean operation.

Example



This piece of code fills the Red Container with a value of 5. It then ANDs that number with 1. The resulting number after the bitwise operation is 1.

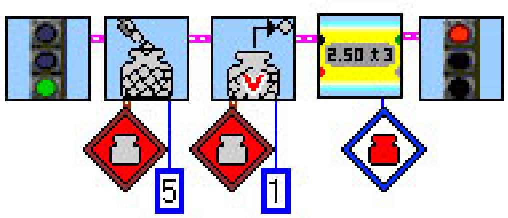
5 = 0101
1 = 0001
1 = 0001



OR Container

This command ORs a bitwise value to the bitwise value of the container. This is NOT a Boolean operation.

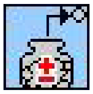
Example



This piece of code fills the Red Container with a value of 5. It then ORs that number with 1. The resulting number after the bitwise operation is 5.

5 = 0101
1 = 0001
5 = 0101

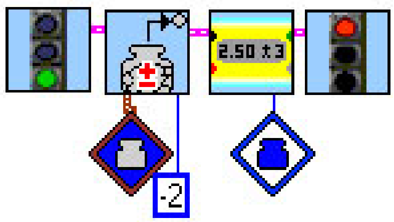
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



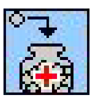
Sign Container

Use this command to put the sign (positive or negative) into a container. The default is to put a zero in the Red Container.

Example



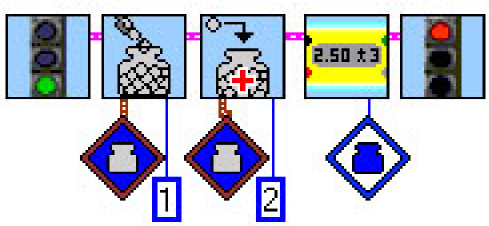
This piece of code takes the sign of the number you put into the container. In this case, -2 is put into the Blue Container. The RCX will then display -1. If the number was 2, then 1 would be put into the container.



Add to Container

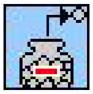
Use this command to add a value to the current value of the container. The default is to add one to the Red Container.

Example



This piece of code fills the Blue Container with a value of 1. It then adds 2 to the Blue Container. The value of the Blue Container would then be displayed on the RCX as 3.

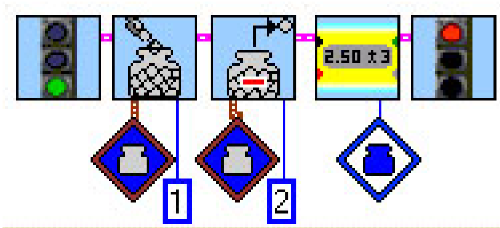
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



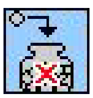
Remove from Container

Use this command to subtract a value from the current value of the container. The default is to subtract one from the Red Container.

Example



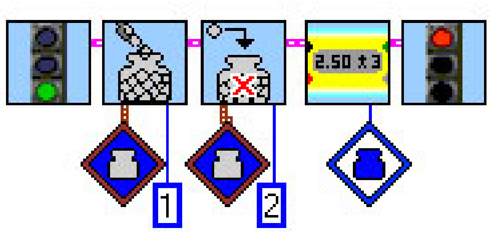
This piece of code fills the Blue Container with a value of 1. It then subtracts 2 from the Blue Container. The value of the Blue Container would then be displayed on the RCX as -1.



Multiply to Container

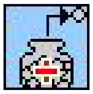
Use this command to multiply the value of a container by a certain number. The default is to multiply the Red Container by two.

Example



This piece of code fills the Blue Container with a value of 1. It then multiplies 2 by the value of the Blue Container. The value of the Blue Container would then be displayed on the RCX as 2.

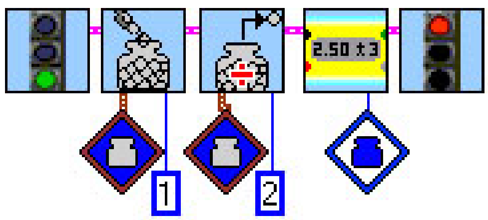
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Divide Container

Use this command to divide the value of a container by a certain number. The default is to divide the Red Container by two. User can only use integers (1,2,3....). Non-integer numbers will be rounded to the nearest integer. The result also can only be an integer. Non-integer results will be truncated (the decimal parts of the number will be removed).

Example



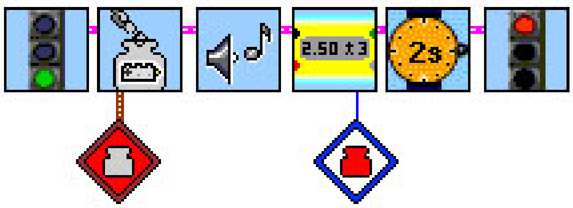
This piece of code fills the Blue Container with a value of 1. It then divides the value of the Blue Container by 2. The value of the Blue Container would then be displayed on the RCX as 0 (0.5 truncated = 0).

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Touch Container

Use this command to fill a container with the value of the touch sensor. This value will either be a zero or a one. The default is to set the Red Container to the value of the touch sensor on port 1. In your program, be sure to specify what port your touch sensor is connected to with a modifier.

Example



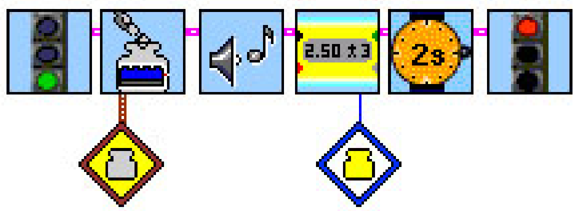
This piece of code fills the Red Container with the value of the touch sensor. A sound is played and the value of the Red Container is shown on the RCX. If the value of the touch sensor was 1 (meaning it was pressed in), the RCX would read 1.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Light Container

Use this command to fill a container with the value of the light sensor. The default is to set the Red Container to the value of the light sensor on port 1. In your program, be sure to specify what port your light sensor is connected to with a modifier.

Example



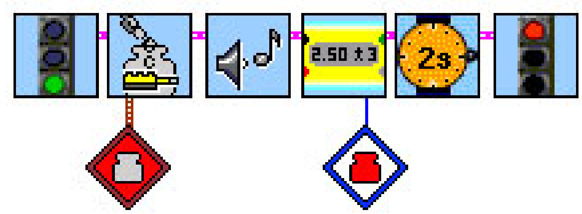
This piece of code fills the Yellow Container with the value of the light sensor. A sound is played and the value of the Yellow Container is shown on the RCX. If the value of the light sensor was 41, the RCX would read 41.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Temp (C) Container

Use this command to fill a container with the value of the temperature (Celsius) sensor. The default is to set the Red Container to the value of the temperature sensor on port 1. In your program, be sure to specify what port your temperature sensor is connected to with a modifier. Rounds value to nearest integer value (20.7 goes to 21)

Example



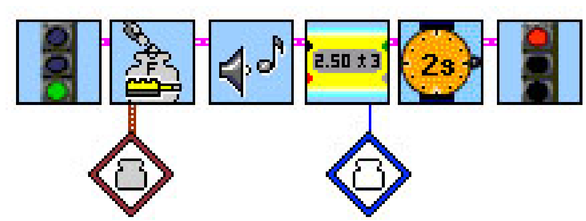
This piece of code fills the Red Container with the value of the Temperature sensor in Celsius. A sound is played and the value of the Red Container is shown on the RCX. If the value of the sensor was 25 deg, the RCX would read 25.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Temp (F) Container

Use this command to fill a container with the value of the temperature (Fahrenheit) sensor. The default is to set the Red Container to the value of the temperature sensor on port 1. In your program, be sure to specify what port your temperature sensor is connected to with a modifier. Rounds value to nearest integer value (70.7 goes to 71)

Example



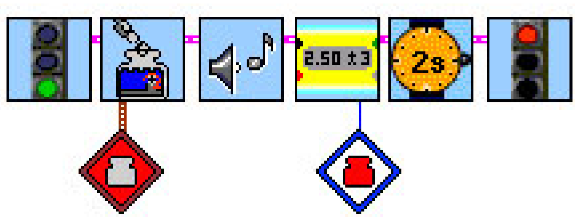
This piece of code fills the Generic Container with the value of the Temperature sensor in Fahrenheit. A sound is played and the value of the Generic Container is shown on the RCX. If the value of the sensor was 75 deg, the RCX would read 75.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Angle Container

Use this command to fill a container with the value of the rotation sensor. The default is to set the Red Container to the value of the rotation sensor on port 1. In your program, be sure to specify what port your rotation sensor is connected to with a modifier.

Example



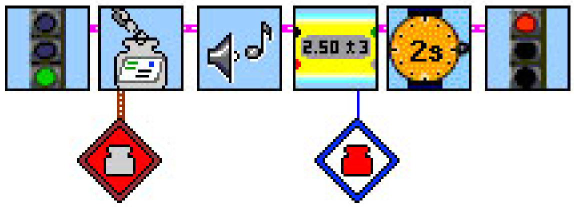
This piece of code fills the Red Container with the value of the Rotation sensor. A sound is played and the value of the Red Container is shown on the RCX. If the value of the sensor was 8, the RCX would read 8.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Mail Container

Use this command to fill a container with the value of the mailbox. The default is to set the Red Container to the value of the mailbox.

Example



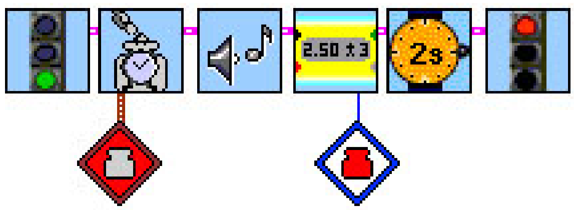
This piece of code fills the Red Container with the value of the mailbox. A sound is played and the value of the Red Container is shown on the RCX. If the value of the mail was 2, the RCX would read 2.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Timer Value Container

Use this command to fill a container with the value of the timer. The default is to set the Red Container to the value of the red timer. You can use modifiers to determine what container you're using as well as what timer value you're using.

Example



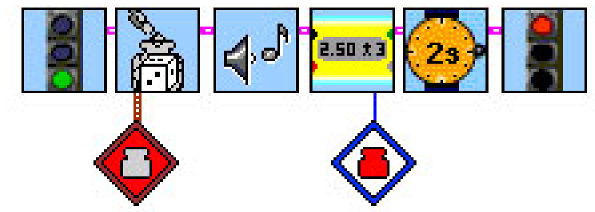
This piece of code fills the Red Container with the value of the timer. A sound is played and the value of the Red Container is shown on the RCX. If the value of the sensor was 10, the RCX would read 10.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Random Fill Container

Use this command to fill the container with a random value between zero and a max random number (determined by a modifier). The default is to fill the Red Container with a random number between 0 and 8.

Example



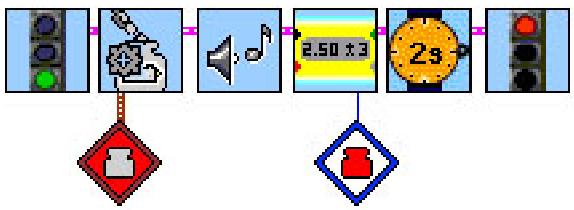
This piece of code fills the Red Container with a random number from 0 to 8. A sound is played and the value of the Red Container is shown on the RCX. If the value of the sensor was 8, the RCX would read 8.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Clicks Container

Use this command to fill a container with the number of clicks of the touch sensor. The default is to set the Red Container to the number of clicks from the touch sensor on port 1. In your program, be sure to specify what port your touch sensor is connected to with a modifier.

Example



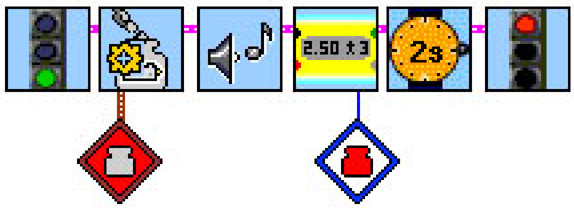
This piece of code fills the Red Container with the number of clicks of the touch sensor. A sound is played and the value of the Red Container is shown on the RCX. If the number of clicks was 3, the RCX would read 3.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Touch and Release Container

Use this command to fill a container with the number of touch and releases of the touch sensor. The default is to set the Red Container to the number of touch and releases from the touch sensor on port 1. In your program, be sure to specify what port your touch sensor is connected to with a modifier.

Example



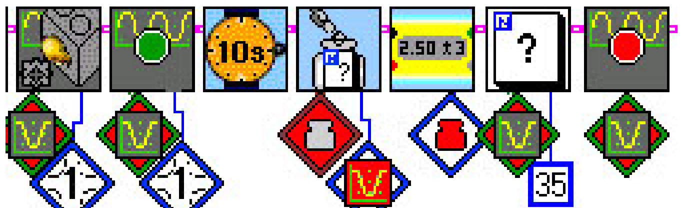
This piece of code fills the Red Container with the number of touch and releases of the touch sensor. A sound is played and the value of the Red Container is shown on the RCX. If the number of touch and releases was 3, the RCX would read 3.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Points Container

Use this command to fill a container with the number of data points collected in a data set. The default is to set the Red Container to the number of data points from the Red Data Set.

Example



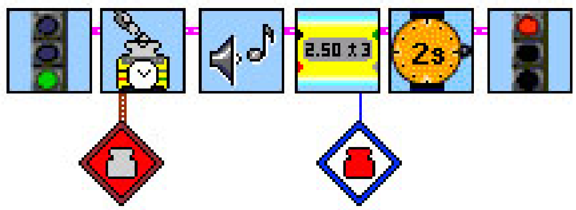
This piece of code initializes the touch sensor on Port 1 on the Red Data Set then begins taking data. It waits 10 seconds and then fills the Red Container with the number of points on the Red Data Set after that time. It then displays this number on the RCX and continues to wait until 35 data points are collected on the Red Data Set.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Clock Container

Use this command to fill a container with the value of the clock. The default is to set the Red Container to the value of the clock.

Example



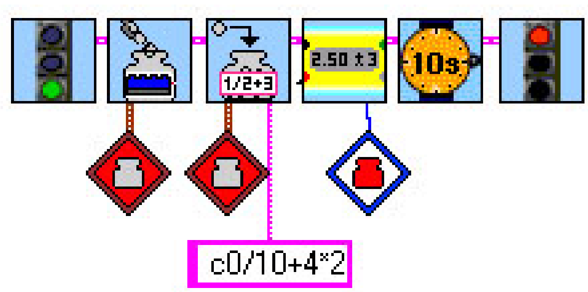
This piece of code fills the Red Container with the value of the clock. A sound is played and the value of the Red Container is shown on the RCX. If the value of the clock was 30 minutes (the amount of time the RCX was on), the RCX would read 30.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Formula Container

Use this command to fill a container with the result of a mathematical formula. The default is to set the Red Container to a value of one. In order to create a string (the area where the formula is written), you must right click on the lower right hand corner of the icon and click on create→constant. You can then type in your formula in the pink box.

Example



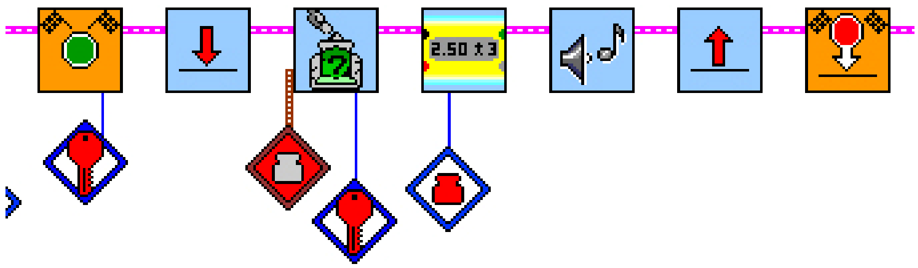
This piece of code fills the Red Container with the value of the light sensor on port 1. It then divides the value of the Red Container by 10, adds 8 (4 times 2), and then puts the resulting value into the Red Container. This value is then displayed on the RCX.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Event Register Container

This command sets the container to an event state. The event states are as follows: 0-low, 1-medium, 2-high, 3-undefined, 4-start calibrating, 5-calibrating in process.

Example



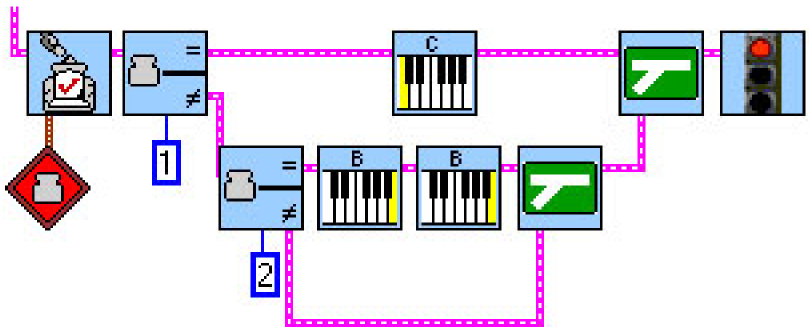
This piece of code (where an event was configured earlier) takes the current event state, puts it in the Red Container and displays it on the RCX's LCD and then beeps. This process loops continuously until the event has been triggered.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Event State Container

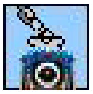
Set the container to a copy of the bit register of the successful event(s) of the current task. The default is to set the Red Container to the register.

Example



This piece of code is a part of a larger program. The Event State icon lets the user discover what event was triggered and react accordingly.

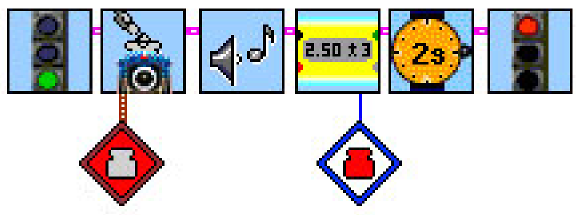
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Camera Sensor Container

This command sets the container to the value of the camera sensor. The value of the camera sensor is defined by the user in Vision Center.

Example



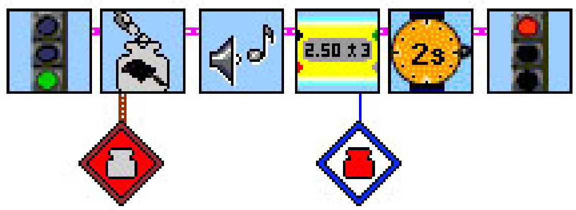
This piece of code fills the Red Container with the value that is returned from the camera and displays it on the RCX. In order for this icon to work, you must be running Vision Center which can be found under the Project menu on the ROBOLAB toolbar.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Generic Sensor Container

This command sets the container to the value of the generic sensor adapter. The default is to set the value of the generic sensor adapter to the Red Container.

Example



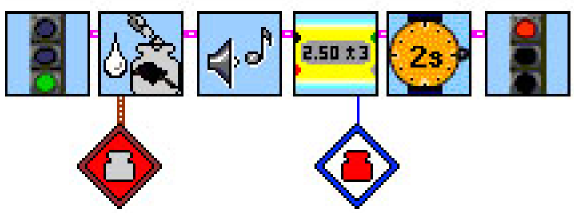
This piece of code fills the Red Container with the value of the generic sensor. A sound is played and the value of the Red Container is shown on the RCX. If the value of the generic sensor was 32, the RCX would read 32.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Humidity Container (HumiPro LogIT)

Use this command to fill a container with the value of the humidity sensor. The default is to set the Red Container to the value of the humidity sensor on port 1. In your program, be sure to specify what port your humidity sensor is connected to with a modifier.

Example



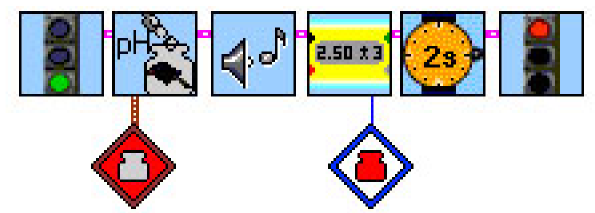
This piece of code fills the Red Container with the value of the humidity sensor. A sound is played and the value of the Red Container is shown on the RCX. If the value of the humidity sensor was 32, the RCX would read 32.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

pH Container (pH LogIT)

Use this command to fill a container with the value of the pH sensor. The default is to set the Red Container to the value of the pH sensor on port 1. In your program, be sure to specify what port your pH sensor is connected to with a modifier.

Example



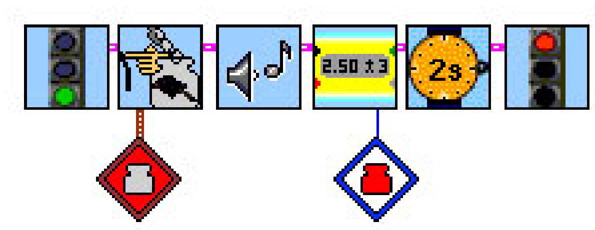
This piece of code fills the Red Container with the value of the pH sensor. A sound is played and the value of the Red Container is shown on the RCX. If the value of the pH sensor was 8, the RCX would read 8.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Pressure Container (Pressure LogIT)

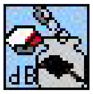
Use this command to fill a container with the value of the pressure sensor adapter. The default is to set the Red Container to the value of the pressure sensor adapter on port 1. In your program, be sure to specify what port your pressure sensor adapter is connected to with a modifier.

Example



This piece of code fills the Red Container with the value of the pressure sensor. A sound is played and the value of the Red Container is shown on the RCX. If the value of the pressure sensor was 4, the RCX would read 4.

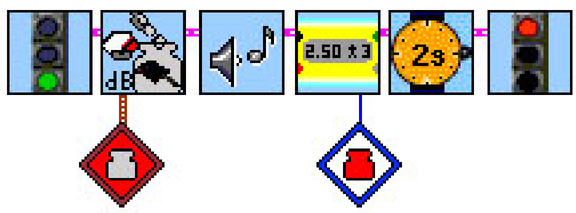
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Sound Level Container (Sound LogIT)

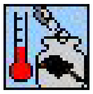
Use this command to fill a container with the value of the sound level sensor. The default is to set the Red Container to the value of the sound level sensor on port 1. In your program, be sure to specify what port your sound level sensor is connected to with a modifier.

Example



This piece of code fills the Red Container with the value of the sound sensor. A sound is played and the value of the Red Container is shown on the RCX. If the value of the sound sensor was 8, the RCX would read 8.

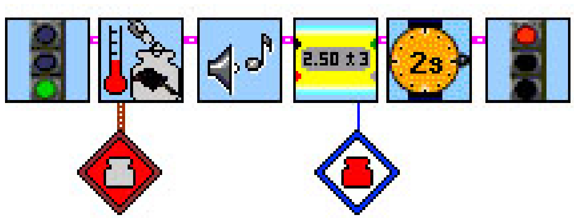
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Temperature Container (TempPro LogIT)

Use this command to fill a container with the value of the temperature sensor adapter. The default is to set the Red Container to the value of the temperature sensor adapter on port 1. In your program, be sure to specify what port your temperature sensor adapter is connected to with a modifier.

Example



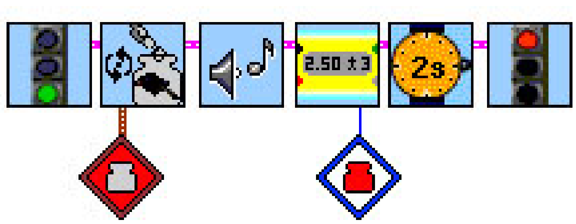
This piece of code fills the Red Container with the value of the ProTemp sensor. A sound is played and the value of the Red Container is shown on the RCX. If the value of the ProTemp sensor was 8, the RCX would read 8.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Position Container (Position LogIT)

Use this command to fill a container with the value of the position sensor adapter. The default is to set the Red Container to the value of the position sensor adapter on port 1. In your program, be sure to specify what port your position sensor adapter is connected to with a modifier.

Example



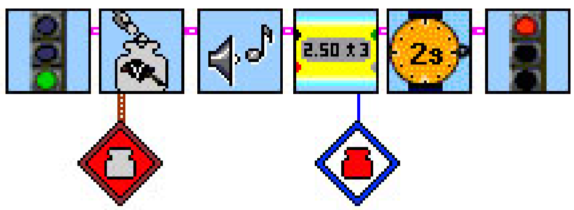
This piece of code fills the Red Container with the value of the position sensor. A sound is played and the value of the Red Container is shown on the RCX. If the value of the position sensor was 8, the RCX would read 8.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Voltage Container (Voltmeter LogIT)

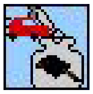
Use this command to fill a container with the value of the voltage sensor adapter. The default is to set the Red Container to the value of the voltage sensor adapter on port 1. In your program, be sure to specify what port your voltage sensor adapter is connected to with a modifier.

Example



This piece of code fills the Red Container with the value of the voltage sensor. A sound is played and the value of the Red Container is shown on the RCX. If the value of the voltage sensor was 8, the RCX would read 8.

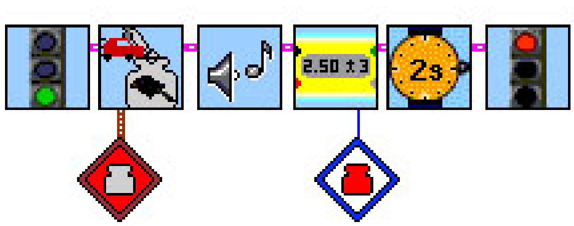
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Acceleration Sensor Container

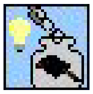
Use this command to fill a container with the value of the acceleration sensor. The default is to set the Red Container to the value of the acceleration sensor on port 1. In your program, be sure to specify what port your acceleration sensor is connected to with a modifier.

Example



This piece of code fills the Red Container with the value of the accelerometer. A sound is played and the value of the Red Container is shown on the RCX. If the value of the accelerometer was 8, the RCX would read 8.

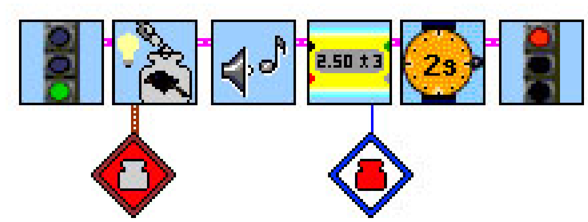
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Lux Sensor Container

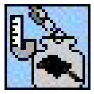
Use this command to fill a container with the value of the Lux sensor. The default is to set the Red Container to the value of the Lux sensor on port 1. In your program, be sure to specify what port your Lux sensor is connected to with a modifier.

Example



This piece of code fills the Red Container with the value of the LUX sensor. A sound is played and the value of the Red Container is shown on the RCX. If the value of the LUX sensor was 8, the RCX would read 8.

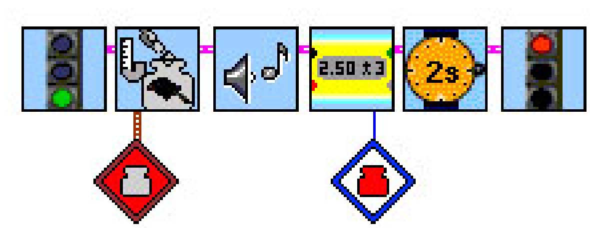
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Barometric Sensor Container

Use this command to fill a container with the value of the barometric sensor. The default is to set the Red Container to the value of the barometric sensor on port 1. In your program, be sure to specify what port your barometric sensor is connected to with a modifier.

Example



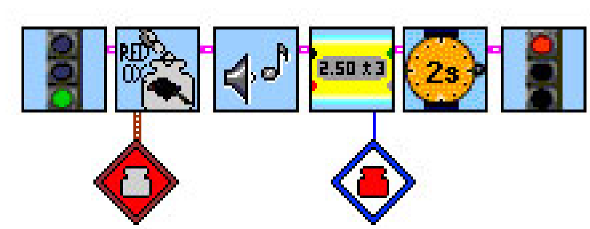
This piece of code fills the Red Container with the value of the barometer sensor. A sound is played and the value of the Red Container is shown on the RCX. If the value of the barometer sensor was 8, the RCX would read 8.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Redox Sensor Container

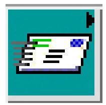
Use this command to fill a container with the value of the Redox sensor. The default is to set the Red Container to the value of the Redox sensor on port 1. In your program, be sure to specify what port your Redox sensor is connected to with a modifier.

Example

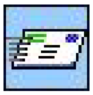


This piece of code fills the Red Container with the value of the Redox sensor. A sound is played and the value of the Red Container is shown on the RCX. If the value of the Redox sensor was 8, the RCX would read 8.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



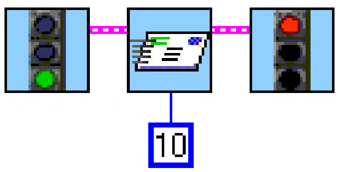
RCX Communications



Send Mail

Use this icon to send mail (numbers) to another RCX. There must be a continuous line of site between the 2 RCXs when the program is run.

Example

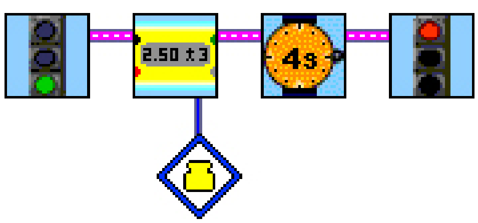


This piece of code would send a value of 10 to the mailbox of another RCX.

Set Display

Use the icon to control the LCD display on your RCX. You can display any modifier in this area. This can be useful for debugging mail or container issues as well as monitoring sensors. The display is only set to view the modifier while the program is running.

Example



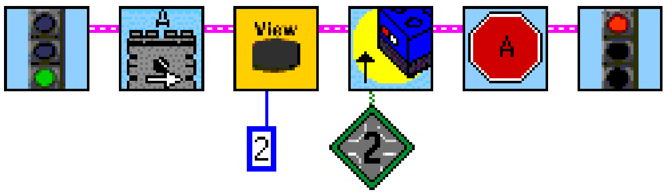
This piece of code would display the value of the yellow container for 4 seconds.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Change View on RCX

Use this to change the display on the RCX to information from the inputs (1,2,3), outputs (4,5,6), or clock (0). The default is to display the clock.

Example

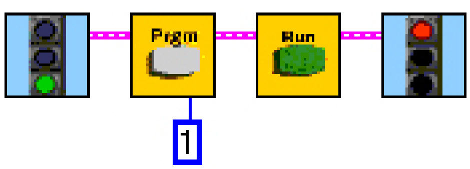


This piece of code will turn on motor A, then change the RCX display to input 2. Then it will wait for the value of light sensor on port 2 to increase by 5 (the default). Then the motor will be shut off.

Change Program on RCX

Use this to change the number of the program in the RCX.

Example



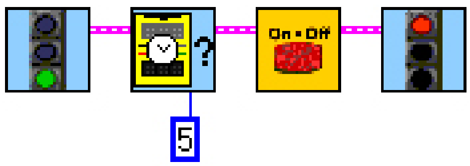
This piece of code would run program 1.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Use this to turn off the RCX's power.

Turn RCX Power Off

Example

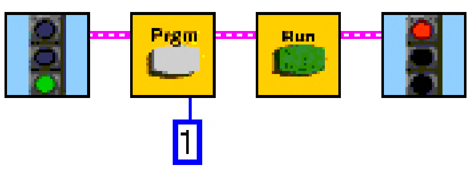


This piece of code would wait for the RCX clock to reach 5 minutes, and then it would turn off power to the RCX.

Use this to run a program.

Run Program on RCX

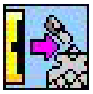
Example



This piece of code would run program 1.

educational division

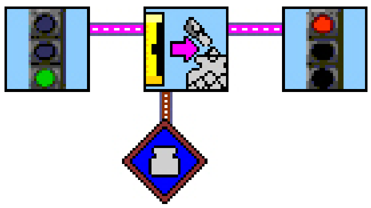
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



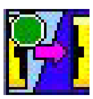
Fill Remote Container

This sets the value of a container on another RCX to the specified value. This is similar to the send mail command. It is useful if you have multiple values you want to send to another RCX.

Example



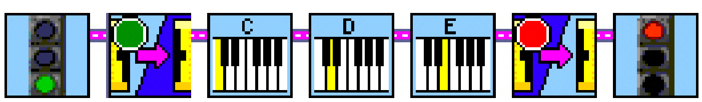
This piece of code will set the Blue Container on another RCX to 1 (the default).



Start Direct RCX Communication

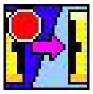
This icon is used to start direct communication between 2 RCXs. Commands that appear between this icon and the End Direct RCX Communication icon are sent to another RCX in direct mode (there must be a continuous line of site between the 2 RCXs).

Example



This piece of code would tell another RCX (in line of site) to play each of the 3 notes. Because the RCXs are talking in direct mode the other RCX would immediately play the notes.

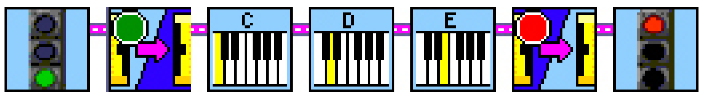
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



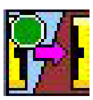
End Direct RCX Communication

This icon is used to end direct communication between 2 RCXs. Commands that appear between this icon and the Begin Direct RCX Communication icon are sent to another RCX in direct mode (there must be a continuous line of site between the 2 RCXs).

Example



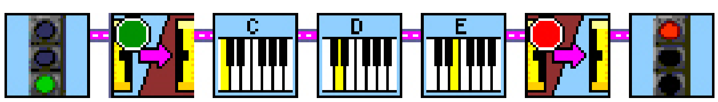
This piece of code would tell another RCX (in line of site) to play each of the 3 notes. Because the RCXs are talking in direct mode the other RCX would immediately play the notes.



Start Remote Program

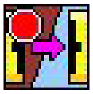
This icon is used to start the downloading of a program from one RCX to another. The commands that are between this icon and the Download Remote Program icon would be downloaded to a 2nd RCX as a program (there must be a continuous line of site between the 2 RCXs during the download).

Example



This piece of code would download a program containing only the three notes to another RCX. The 2nd RCX would NOT play the notes until the Run button was pressed (or the program was told to run programmatically).

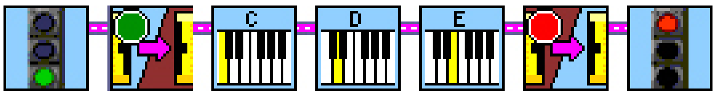
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



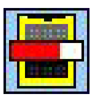
End Remote Program

This icon is used to end the downloading of a program from one RCX to another. The commands that are between this icon and the Start Remote Program icon would be downloaded to a 2nd RCX as a program (there must be a continuous line of site between the 2 RCXs during the download).

Example



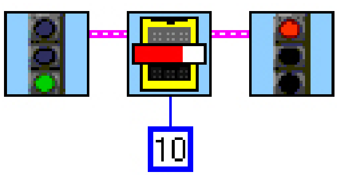
This piece of code would download a program containing only the three notes to another RCX. The 2nd RCX would NOT play the notes until the Run button was pressed (or the program was told to run programmatically).



Set RCX Powerdown Time

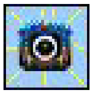
Use this piece of code to change the powerdown time on the RCX. This can also be done with the Robolab Administrator.

Example



This piece of code sets the powerdown time to 10 minutes.

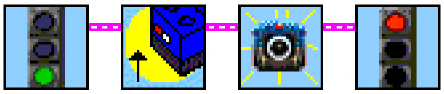
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



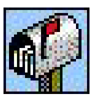
Snap Image

Use this to snap an image during a program.

Example



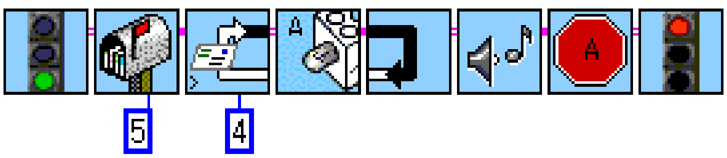
This piece of code waits until the light sensor reading increases by 5 (the default), and then it snaps an image.



Fill Mailbox

Use this icon to fill the mailbox on an RCX with a specific value. Can be used in a program or in direct mode.

Example



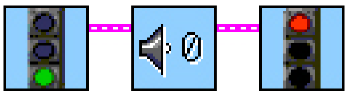
This piece of code fills the mailbox with the value of 5 then starts a loop which turns on lamp A. The loop ends when the RCX receives mail whose value is 4 or less. Then the RCX plays a sound and turns off the lamp.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Clear Sound Buffer

Use this command to clear sounds and reset sound settings. This icon can be helpful before events requiring sound input.

Example



This piece of code would clear the RCX sound buffer.

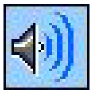
Mute Sound

Use the Mute Sound icon to put your RCX into silent mode. Mute Sound suppresses all sounds (notes or sounds in a running program, download sounds) except for the sounds made when pressing run, stop, view and On Off buttons. Useful when you want to programmatically create a stealth RCX or when you want to program your RCX in the library.

Example

This piece of code, when run, will mute all further sound (except of those made by pressing RCX buttons). This mute applies to other programs (existing or newly download). The only way to get sound is to include or run the Unmute Sound command.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



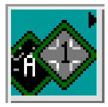
Unmute Sound

Use the Unmute Sound icon to restore the sound capabilities to your RCX. (This is the opposite of Mute Sound).

Example

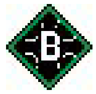
This piece of code, when run, will unmute sound on the RCX.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

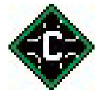


Modifiers

Output A



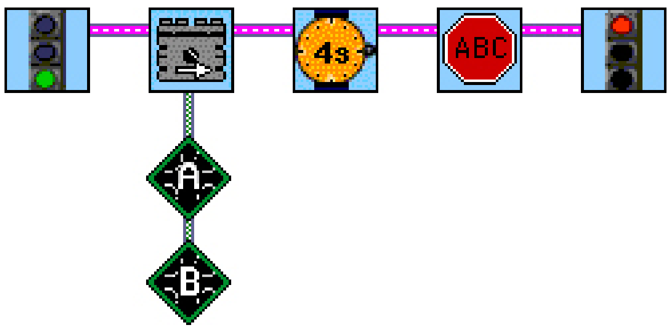
Output B



Output C

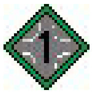
Use these modifiers to indicate to which port your output device is connected.

Example

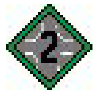


This piece of code would turn on motors connected to Ports A and B in the forward direction. After 4 seconds, it would turn them off.

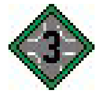
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Input 1



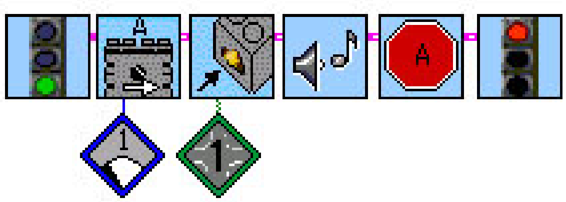
Input 2



Input 3

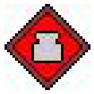
Use these modifiers to indicate to which ports your input device is connected.

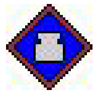
Example



This piece of code runs motor A forward at power level one until the touch sensor that is connected to Port 1 is pressed in. It then plays a sound and stops motor A.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

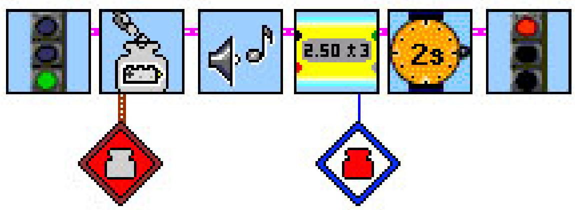

Red Container


Blue Container


Yellow Container

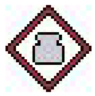
Use these modifiers to indicate that you are using values associated with the specific Containers.

Example



This piece of code fills the Red Container with the value of the touch sensor. A sound is played and the value of the red container is shown on the RCX. If the value of the touch sensor was 1 (meaning it was pressed in), the RCX would read 1.

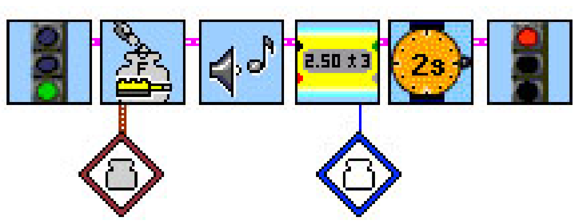
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Generic Container

Use this modifier to indicate that you are using values associated with the Generic Container. Zero (0) corresponds with the Red Container, 1 with the Yellow Container, and 2 with the Blue Container. Containers 3-20 can be used by string in a numeric constant for the container #.

Example



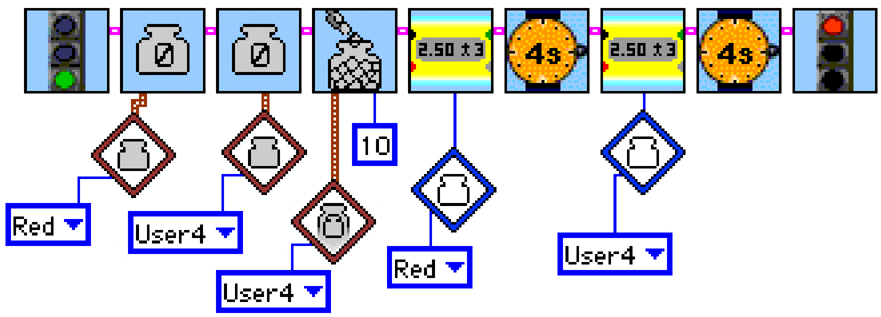
This piece of code fills the Generic Container with the value of the temperature sensor in Fahrenheit. A sound is played and the value of the Generic Container is shown on the RCX. If the value of the sensor was 75 degrees, the RCX would read 75.

educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Container's Container

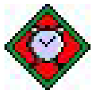
This command takes the value in the container specified (Container #) and uses that to choose the container specified. This is the ROBOLAB equivalent of a C Pointer.

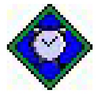
Example



This piece of code zeroes both the generic container (3) and the Red Container. It then takes the value of the generic container(0) and uses that to determine what container should be filled with the value of 4. Zero (0) corresponds to the Red Container so the Red Container is filled with a value of 4 and that is then displayed on the RCX. The program then waits 4 seconds and then shows the value of the Generic Container that is still 0.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

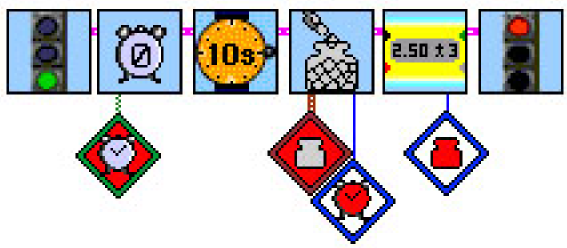

Red Timer


Blue Timer


Yellow Timer

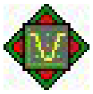
Use this modifier to indicate that you are using values associated with the Red Timer.

Example



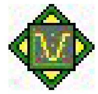
This piece of code zeroes the Red Timer and then waits for 10 seconds. The value of the Red Timer is put into the Red Container and then displayed on the RCX.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Red Data Set

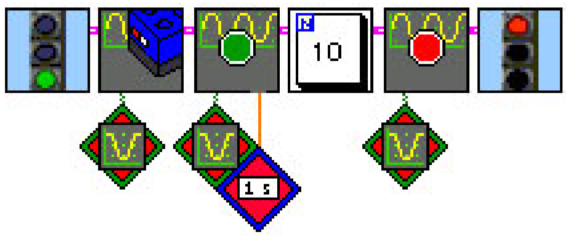
Blue Data Set



Yellow Data Set

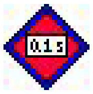
Use this modifier to indicate that you are using values associated with the Red, Blue, or Yellow Data Set. You can wire in additional Data Sets.

Example

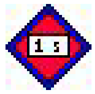


This piece of code will initialize the light sensor on port one, and will begin data logging on the Red Data Set for 10 points. This code captures data every second.

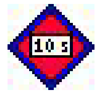
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



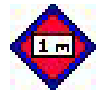
Sample one tenth sec



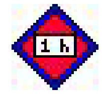
Sample one sec



Sample ten sec



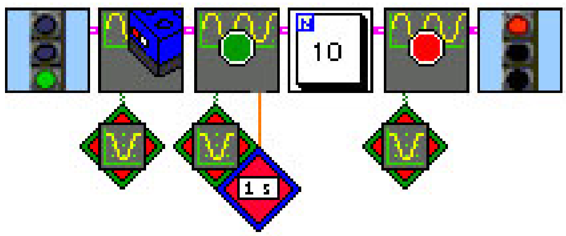
Sample one minute



Sample one hour

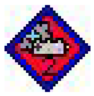
Use this modifier in a data acquisition program to take a sample for a specified amount of time.

Example



This piece of code will initialize the light sensor on port one, and will begin data logging on the Red Data Set for 10 points. This code captures data every second.

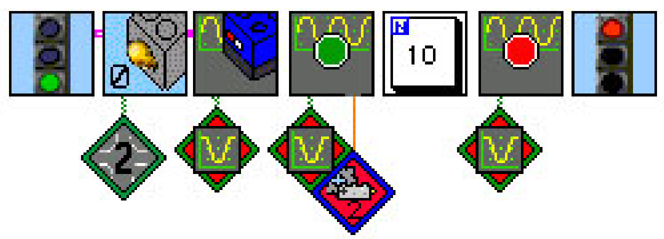
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Touch Sampling

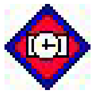
Use this modifier in a data acquisition program to take a sample every time the touch sensor is pressed on port two.

Example



First, this piece of code zeroes the touch sensor on Port 2. It then initializes the light sensor on port one and begins to take data points in the Red Data Set every time the touch sensor on port 2 is pressed. The program stops data logging when the user gets 10 points.

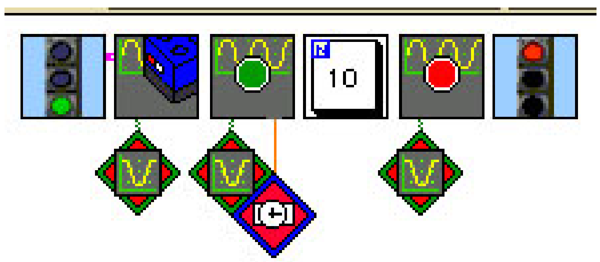
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Free Sampling with Time Stamp

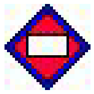
String this modifier to a Start Data Logging subroutine to capture data points every time you write to the data set and marks the data with time. The time will rollover every 2^16 tenths of a second.

Example



This piece of code initializes the light sensor on port one, takes 10 data points on the Red Data Set, and then stops taking data points. It captures data every time you write to the data set and marks the data with time.

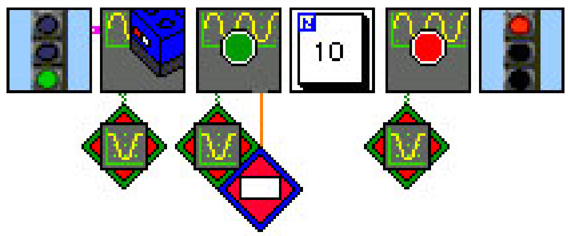
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



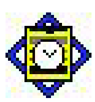
Free Sampling

String this modifier to a Start Data Logging subroutine to capture data points every time you write to the data set.

Example



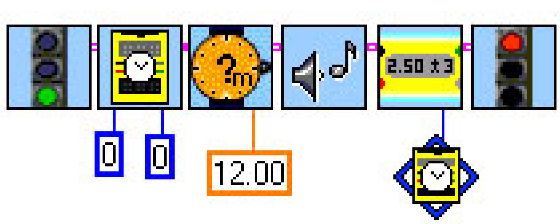
This piece of code initializes the light sensor on port one, takes 10 data points on the Red Data Set, and then stops taking data points. It captures data every time you write to the data set.



Value of Clock

Use this modifier to use the value of the clock since 00:00 in minutes.

Example



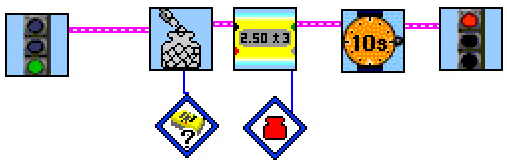
This piece of code sets the clock to 0 hours and 0 minutes and then waits 12 minutes. A sound is played and the value of the clock is displayed on the RCX.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

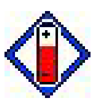
Value of Firmware

Use this command when you want to see the value of the firmware in your RCX.

Example



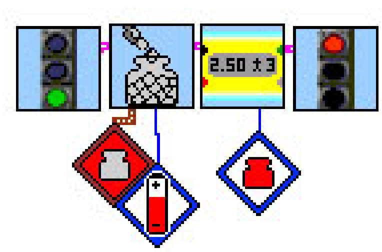
This piece of code fills the Red Container with the value of the firmware in your RCX and then displays it on your RCX screen.



Value of Battery

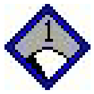
Use this command when you want to use or monitor the battery value in your program. (Battery Power is represented as 1000 times the battery power so 7.654 = 7654).

Example

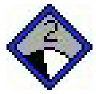


This piece of code fills the Red Container with the value of the battery and then displays the value on the RCX.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



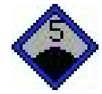
Power Level 1



Power Level 2

Power Level 3

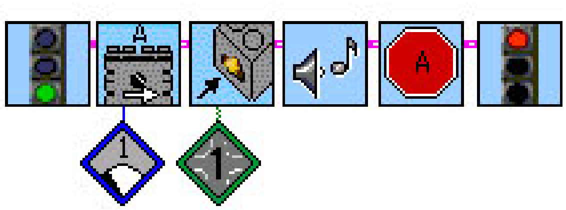
Power Level 4



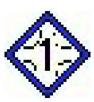
Power Level 5

String these modifiers to a motor command to control the speed of the motor. Speed 1 is the slowest and speed 5 is the fastest.

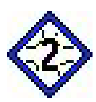
Example



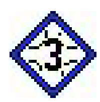
This piece of code runs motor A forward at power level one until the touch sensor that is connected to Port one is pressed in. It then plays a sound and stops motor A.



Value of Port 1



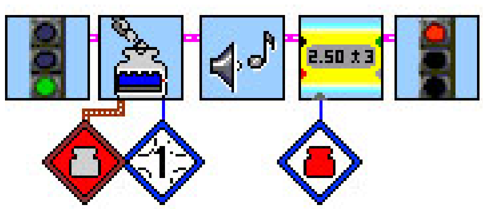
Value of Port 2



Value of Port 3

Use these modifiers to use the value of ports 1, 2, and 3.

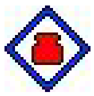
Example



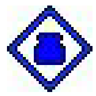
This piece of code fills the Red Container with the value

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

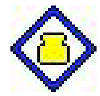
of the light sensor on Port 1. It then plays a sound and displays the value of the red container on the RCX.



Value of Red Container



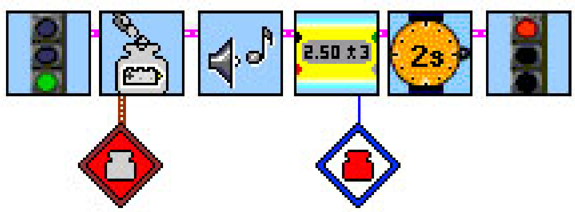
Value of Blue Container



Value of Yellow Container

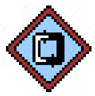
Use these modifiers to use the value of the red, blue, and yellow containers.

Example



This piece of code fills the Red Container with the value of the touch sensor. A sound is played and the value of the red container is shown on the RCX. If the value of the touch sensor was 1 (meaning it was pressed in), the RCX would read 1.

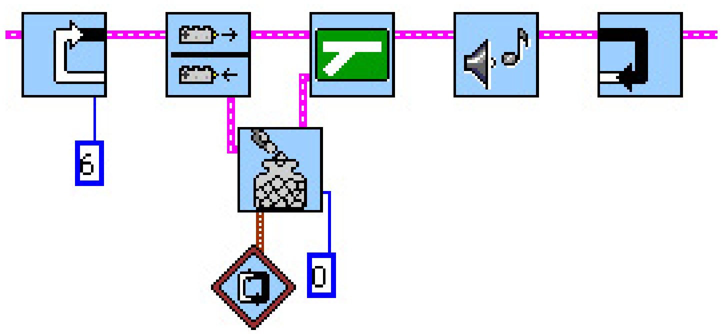
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Loop Counter Container

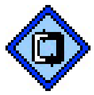
The loop iteration is stored in a variable. Use this modifier, as you would use the Red Container, to set or manipulate the loop iteration value.

Example



This piece of code fills the Red Container with the value of the firmware in your RCX and then displays it on your RCX screen.

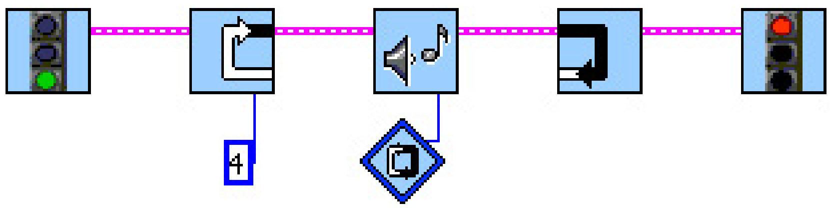
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



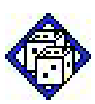
Value of the Loop Counter Container

The loop iteration is stored in a variable . Use this modifier, as you would use the Red Container Value, to view the loop iteration value.

Example



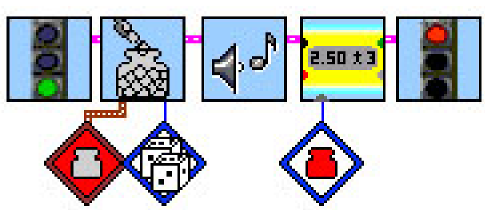
This piece of code sets a loop to iterate 4 times. The value of the loop iteration is wired into the Play Sound icon (which plays different sounds depending which number is wired into it). As the loop iteration changes different sounds are played (Play Sound does not have a sound for the 0 state so the sound for the 1 state is played twice).



Random Number

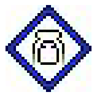
Use this modifier to generate a random number from 0-8.

Example



This piece of code will fill the Red Container with a random number from 0-8. The RCX will then play a sound and display the value of the red container on its screen.

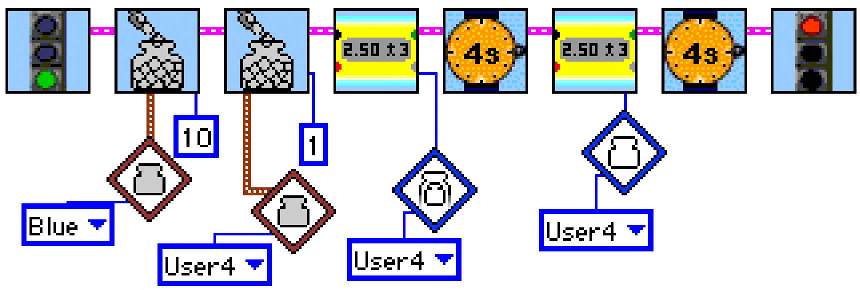
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Value of Container's Container

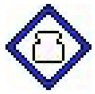
The value in the container specified (Container #) is used to choose a container and the value in that container is passed out.

Example



This piece of code fills the blue container (also known as container #1) with the value 10 and fills the User 4 container with the value 1. It then displays the value of the container specified in User 4. User 4 contains the value 1 (which indicates the Blue container) and the Blue container contains the value 10 -- hence 10 is displayed for 4 seconds. Then the value of User 4 (which is 1) is displayed for 4 seconds.

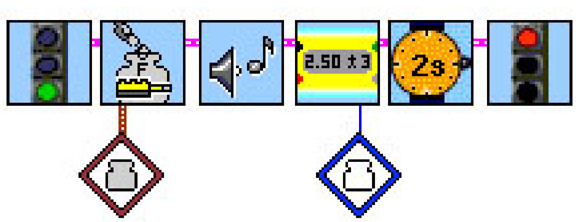
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Value of Generic Container

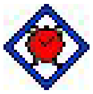
This modifier represents the value of the generic container. You can have up to 21 containers. Zero (0) is red, 1 is yellow, 2 is blue, and 3-20 are user defined.

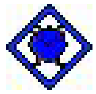
Example

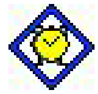


This piece of code fills the Generic Container with the value of the temperature sensor in Fahrenheit. A sound is played and the value of the Generic Container is shown on the RCX. If the value of the sensor was 75 degrees, the RCX would read 75.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

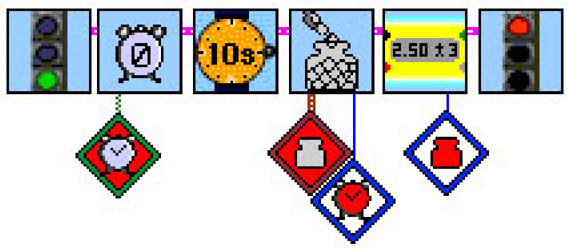

Value of Red Timer


Value of Blue Timer


Value of Yellow Timer

Use this modifier to use the value of the red, blue, and yellow timers.

Example



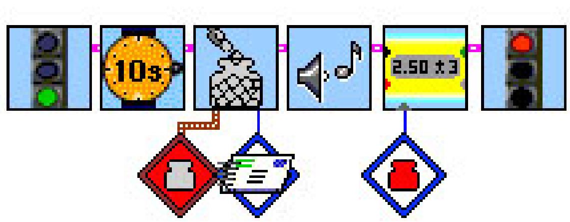
This piece of code zeroes the Red Timer and then waits for 10 seconds. The value of the Red Timer is put into the Red Container and then displayed on the RCX.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

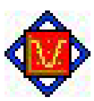
Value of Mail

Use this modifier to use the value of the mail in the mailbox.

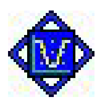
Example



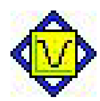
This piece of code will fill the Red Container with the value of the mailbox. The RCX will then play a sound and display the value of the red container on its screen.



Value of Red Data Set



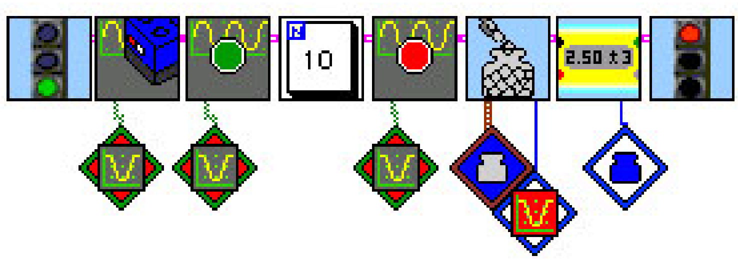
Value of Blue Data Set



Value of Yellow Data Set

Use these modifiers to use the value of the red data set.

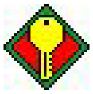
Example

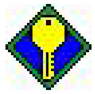


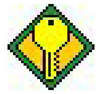
This piece of code initializes the light sensor on Port 1, begins data logging in the Red Data Set for 10 points, and then fills the Blue Container with the last value taken in the Red Data Set. The value of the blue container is then

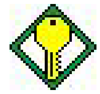
educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

displayed on the RCX.


Red Event


Blue Event


Yellow Event

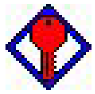

Generic Event

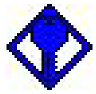
String these modifiers to an event definition or monitor command to select the Red, Blue, Yellow, or Generic Event.

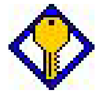
Example

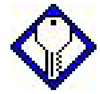
This piece of code sets up a red event that is triggered when the touch sensor is pressed. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the touch sensor is pushed in. This will force the program out of the jump sequence and make it land where the Event Landing is located.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.


Value of Red Event

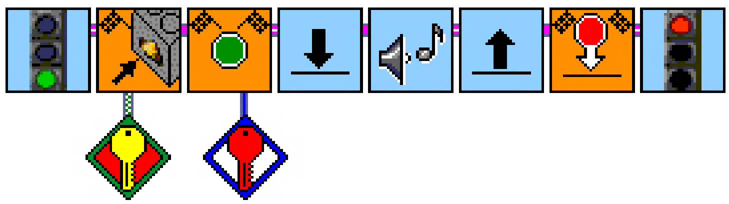

Value of Blue Event


Value of Yellow Event


Value of Generic Event

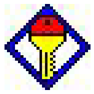
Use these modifiers to represent the value of the Red, Blue, Yellow, or Generic Event.

Example

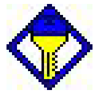


This piece of code sets up a red event that is triggered when the touch sensor is pressed. The Monitor Event icon begins monitoring for such an event to occur. A sound will be played over and over again until the touch sensor is pushed in. This will force the program out of the jump sequence and make it land where the Event Landing is located.

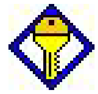
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



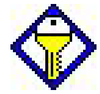
Value of Red Upper Threshold



Value of Blue Upper Threshold



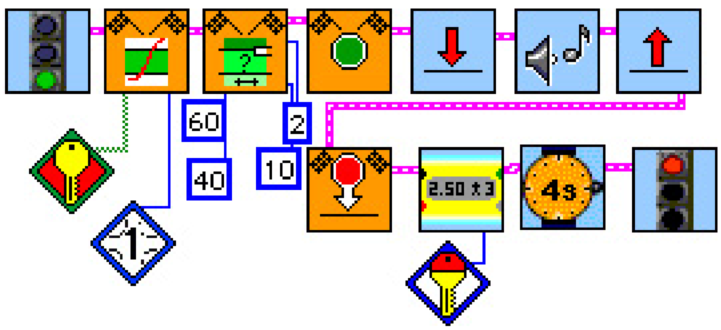
Value of Yellow Upper Threshold



Value of Generic Upper Threshold

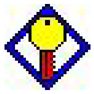
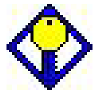
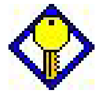
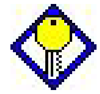
Use this modifier to represent the value of the upper threshold of the Red, Blue, Yellow, or Generic Event.

Example



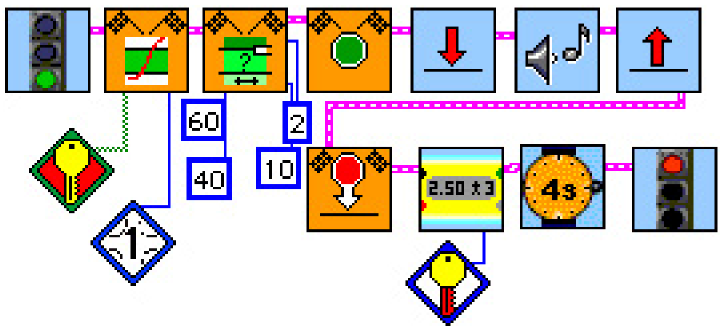
This piece of code sets the red event to be a normal event with an upper threshold of 60, a lower threshold of 40, duration of 10 seconds, and hysteresis of 2. The program will beep repeatedly until the sensor on channel 1 fulfills the event requirements. It will then display the value of the red upper threshold on the RCX LCD for 4 seconds.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

			
Value of Red Lower Threshold	**Value of Blue Lower Threshold**	**Value of Yellow Lower Threshold**	**Value of Generic Lower Threshold**

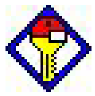
Use these modifiers to represent the value of the lower threshold of the Red, Blue, Yellow, or Generic Event.

Example

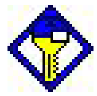


This piece of code sets the red event to be a normal event with an upper threshold of 60, a lower threshold of 40, duration of 10 seconds, and hysteresis of 2. The program will beep repeatedly until the sensor on channel 1 fulfills the event requirements. It will then display the value of the red lower threshold on the RCX LCD for 4 seconds.

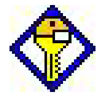
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



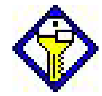
Value of Red Hysteresis



Value of Blue Hysteresis



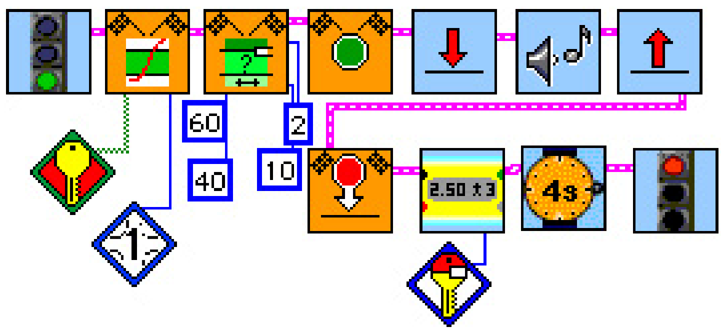
Value of Yellow Hysteresis



Value of Generic Hysteresis

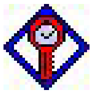
Use these modifiers to represent the value of the Hysteresis of the Red, Blue, Yellow, or Generic Event.

Example

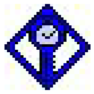


This piece of code sets the red event to be a normal event with an upper threshold of 60, a lower threshold of 40, duration of 10 seconds, and hysteresis of 2. The program will beep repeatedly until the sensor on channel 1 fulfills the event requirements. It will then display the value of the red hysteresis on the RCX LCD for 4 seconds.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

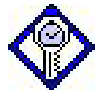


Value of Red Duration



Value of Blue Duration

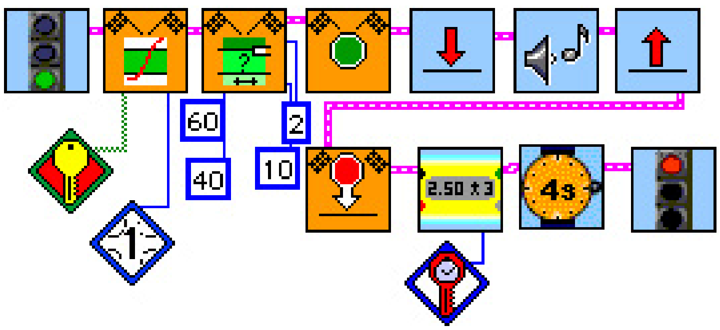
Value of Yellow Duration



Value of Generic Duration

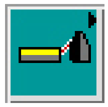
Use these modifiers to represent the value of the duration of the Red, Blue, Yellow, or Generic Event.

Example

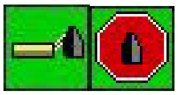


This piece of code sets the red event to be a normal event with an upper threshold of 60, a lower threshold of 40, duration of 10 seconds, and hysteresis of 2. The program will beep repeatedly until the sensor on channel 1 fulfills the event requirements. It will then display the value of the red duration on the RCX LCD for 4 seconds.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



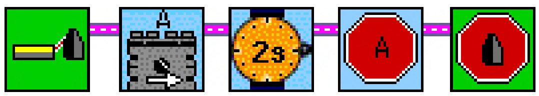
Direct Functions



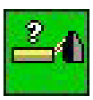
Begin, End direct mode

These two icons are used to start and end direct communication between the RCX and the tower. Commands that appear between the two icons are sent to the RCX in direct mode (there must be continuous line of site between the RCX and the tower).

Example



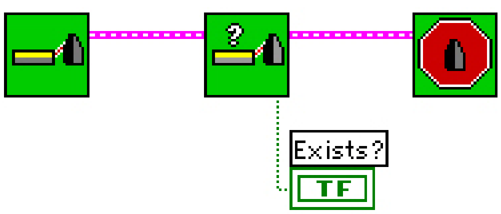
This piece of code would turn motor A on for 2 seconds, and then turn motor A off.



Is RCX in view?

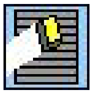
If you are running a direct mode program it can be useful to know if the RCX is in view of the tower and can receive commands. This command updates a Boolean (true / false) indicator with a true if the tower can talk to the RCX and a false if the tower cannot talk to the RCX.

Example



This piece of code looks to see if the RCX is in view of the tower. It updates the Boolean indicator Exists? with the current status.

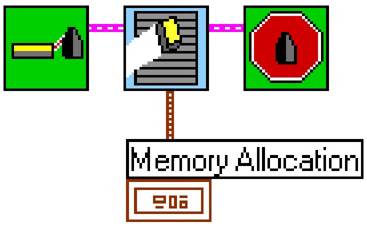
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



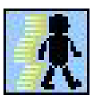
Memory Map

The memory map icon gives you a reading of the memory contents of the RCX. It MUST be used in direct mode and outputted to the front panel.

Example



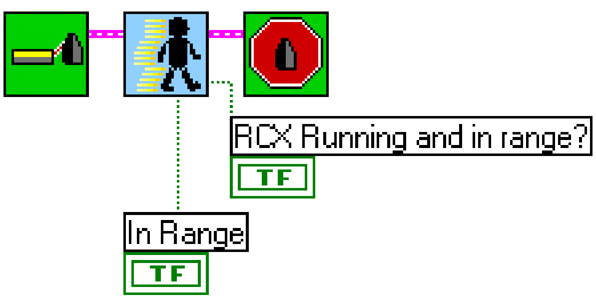
This piece of code will read the RCX memory contents and display the readings on an indicator on the front panel.



Read Run Status

Use this icon to determine whether your RCX is still running a program (i.e. is the person running?). Should be used in direct mode.

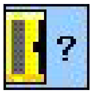
Example



This piece of code displays on the front panel whether the RCX is running and whether it is in range.

LEGO educational division

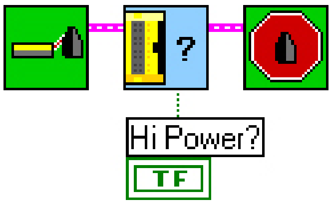
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



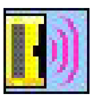
Read tower power

Use this icon to determine the power setting of the tower (little and big arrows equal high and low power mode). Most usable in direct mode programming.

Example



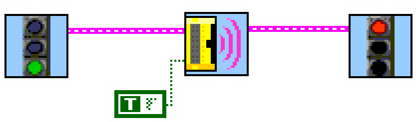
This piece of code reads the tower power and displays a true state on the front panel if it is in high state (and a false state if it is in low power mode).



Set RCX Tower Power

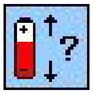
Use this icon to set the IR power of your RCX (high or low - usually done in Administrator). Can be done in any (direct, remote, local) programming mode. The Power modifier is a Boolean (true / false) control. If true is selected, the RCX IR power is set to High. If false is selected, it is set to Low.

Example



This piece of code sets the RCX IR power to High when the program is run.

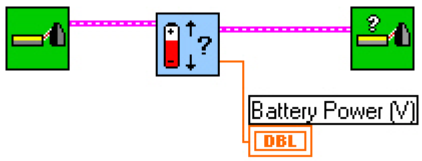
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Read Battery Power

Use this command in direct mode to determine the remaining battery power of your RCX. Useful if your battery power significantly impacts RCX performance.

Example

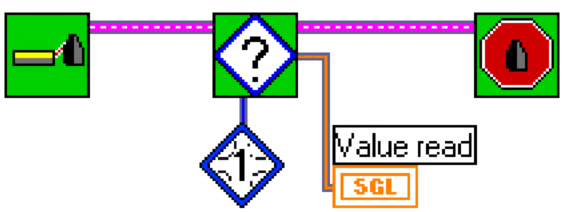


This piece of code will display the Battery Power level on the front panel (between 0-9 Volts).

Read Value

Use this command in direct mode to read the value of any memory location. This command will NOT automatically display that value. To display automatically use the Read and Display Value command. To display using the Read Value command, string an indicator to the Value read terminal. Also, you can choose which port to poll using the modifier to read terminal.

Example



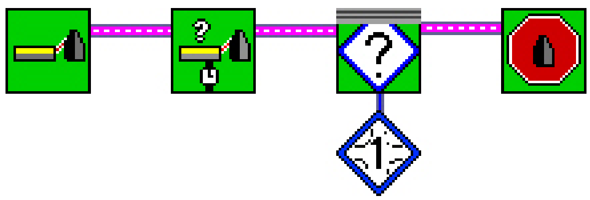
This piece of code reads the value of sensor 1 in direct mode and displays it to the control panel.

educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

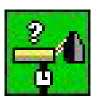
Read and Display Value

Use this command in direct mode to read and display the value of any memory location. This command displays automatically. Choose which port to poll using the modifier to read terminal.

Example



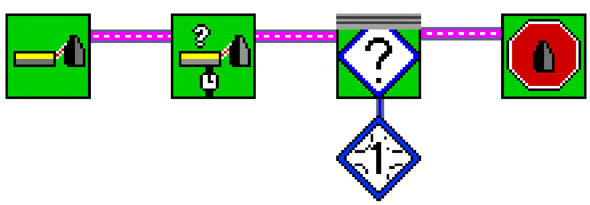
This piece of code waits until the RCX is in view of the tower, then reads and displays the value of sensor 1.



Wait for RCX to be in view

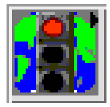
Use this command if you need the RCX to be in view of the tower before moving on.

Example

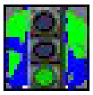


This piece of code waits until the RCX is in view of the tower, then reads and displays the value of sensor 1.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



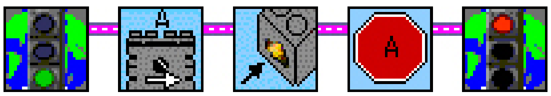
Internet



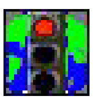
Internet Begin

ALWAYS start an Internet program with this icon.

Example



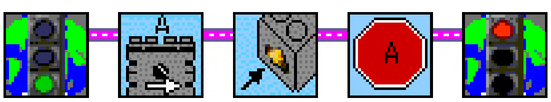
If an IP number was strung onto the Begin icon, this piece of code could be sent over the Internet. It would be stored as a program in a remote RCX that would turn on motor A until a touch sensor was pressed, and then shut it off.



Internet End

ALWAYS end an Internet program with this icon. Each task needs its own End icon.

Example



If an IP number was strung onto the Begin icon, this piece of code could be sent over the Internet. It would be stored as a program in a remote RCX that would turn on motor A until a touch sensor was pressed, and then shut it off.

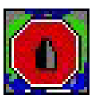
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Begin Internet Direct Mode

ALWAYS start a direct Internet program with this icon.

Example

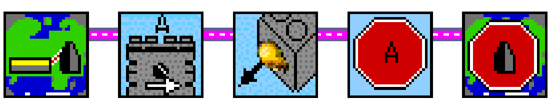
If an IP number was strung onto the Begin Direct icon, this piece of code could control an RCX in direct mode over the Internet. It would turn on motor A on the remote RCX until a touch sensor was released and then turn it off.



End Internet Direct Mode

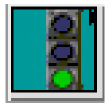
ALWAYS end a direct Internet program with this icon. Each task needs its own Esnd icon.

Example

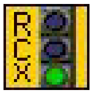


If an IP number were strung onto the Begin Direct icon, this piece of code could control an RCX in direct mode over the Internet. It would turn on motor A on the remote RCX until a touch sensor was released and then turn it off.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



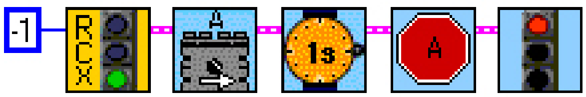
Advanced



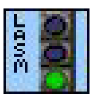
Begin RCX

Use this icon when you want to send a program to a specific tower (if you have multiple towers) on your computer and an RCX is in front of that tower.

Example



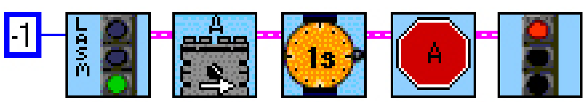
This piece of code would send the program to the default port specified.



Begin LASM

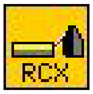
Use this icon when you want to initiate the LASM interface (text based program representation) before downloading.

Example



This piece of code would launch the LASM interface with these commands.

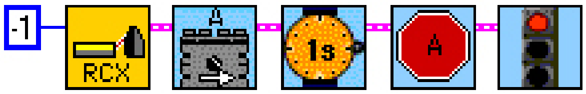
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Begin direct mode

Use this icon to send direct mode commands to a specific tower (if you have multiple towers) on your computer and an RCX is in front of that tower.

Example

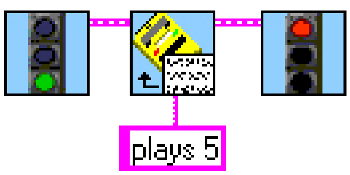


This piece of code would run motor A for 1 second in direct mode (RCX in front of tower).

Generate LASM command

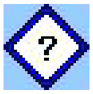
Use this icon to send direct mode commands to a specific tower (if you have multiple towers) on your computer and an RCX is in front of that tower.

Example



This piece of code would send the LASM command 'plays 5' to the RCX.

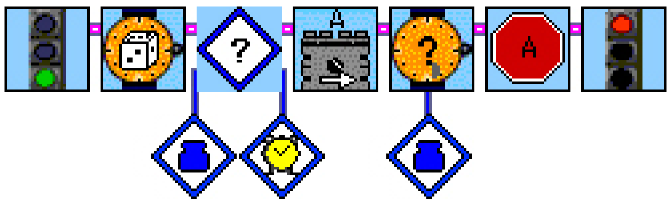
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Set modifier value

Use this to change the value of any modifier to any value you want.

Example



This piece of code waits for a random amount of time, then sets the blue container to the value of the yellow timer. It then turns on motor A and waits for the value of the blue container. It then turns off motor A.

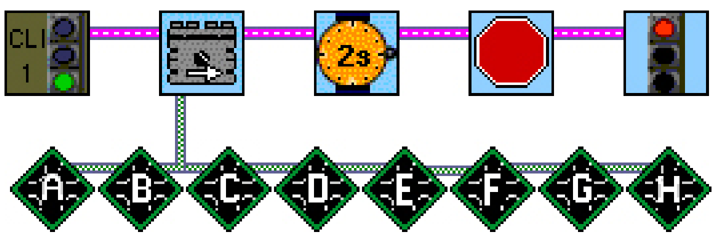
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Control Lab Interface

Output A **Output B** **Output C** **Output D** **Output E** **Output F**
Output G **Output H**

These are output modifiers for the Control Lab interface. String them onto motors, lamps, etc. that are connected to the Control Lab board.

Example



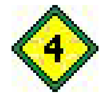
This piece of code would turn on motors connected to all 8 outputs for 2 seconds, and then turn them off.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.


Input 1

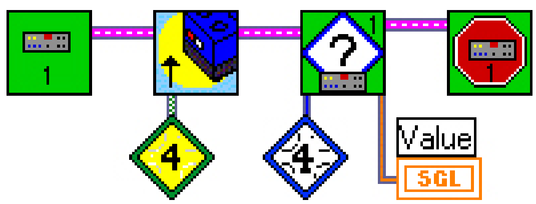

Input 2


Input 3

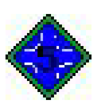

Input 4

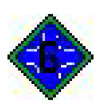
These are input modifiers for the Control Lab interface. String them onto sensors, etc. that are connected to the Control Lab board.

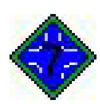
Example

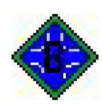


This piece of code would wait until the reading on the light sensor attached to port 4 increased by 5. Then it would display the value of port 4 on the control panel.


Input 5

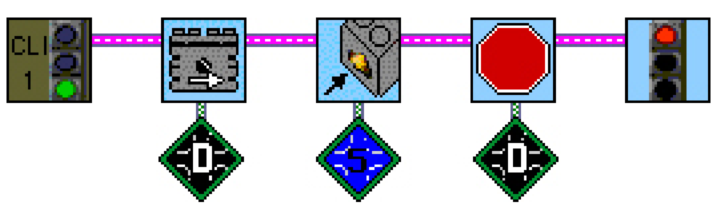

Input 6


Input 7


Input 8

These are input modifiers for the Control Lab interface. String them onto sensors, etc. that are connected to the Control Lab board.

Example

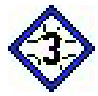


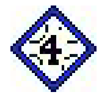
This piece of code would turn on motor D and wait until a touch sensor connected to port 5 was pushed. Then motor D would turn off.

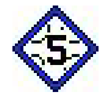
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.


Value of Port 1

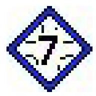

Value of Port 2

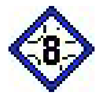

Value of Port 3


Value of Port 4


Value of Port 5

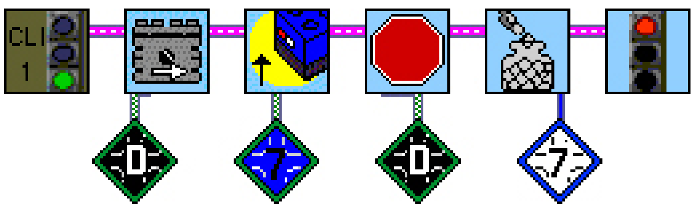

Value of Port 6


Value of Port 7


Value of Port 8

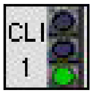
These are input value modifiers. String them onto containers, events, etc.

Example



This piece of code would turn on motor D, wait until the reading on the light sensor on port 7 increased by 5, then turn off motor D. Then it would take the light sensor reading from port 7 and put it in a container.

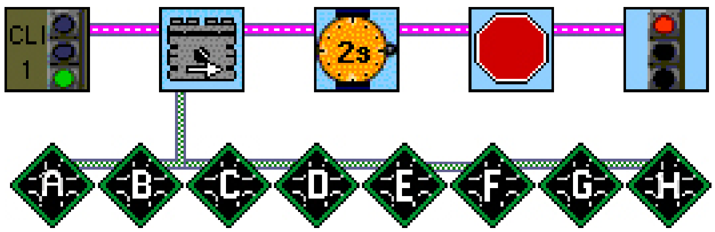
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



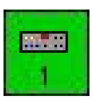
Begin Control Lab Interface

ALWAYS use this to begin a Control Lab program when not in direct mode. You need to string in a number that corresponds to the computer port the Control Lab interface is connected to.

Example



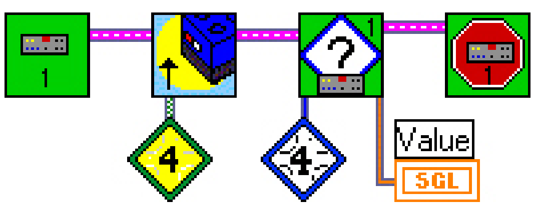
This piece of code would turn on motors connected to all 8 outputs for 2 seconds, and then turn them off.



Begin Direct Mode

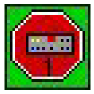
ALWAYS use this to begin a direct mode Control Lab program. You need to string in a number that corresponds to the computer port the Control Lab interface is connected to.

Example



This piece of code would wait until the reading on the light sensor attached to port 4 increased by 5. Then it would display the value of port 4 on the control panel.

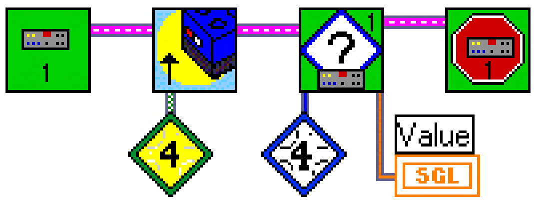
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



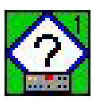
End Direct Mode

ALWAYS use this to end a direct mode Control Lab program. Each task will need its own End icon.

Example



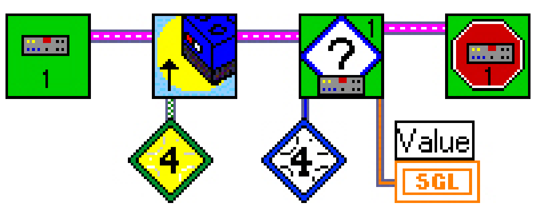
This piece of code would wait until the reading on the light sensor attached to port 4 increased by 5. Then it would display the value of port 4 on the control panel.



Poll Interface Box

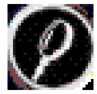
Use this to read a value during direct mode while using the Control Lab interface.

Example

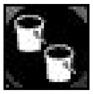


This piece of code would wait until the reading on the light sensor attached to port 4 increased by 5. Then it would display the value of port 4 on the control panel.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



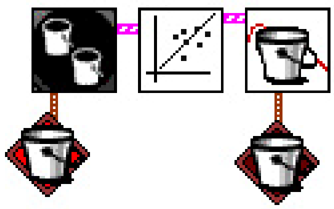
G-Code



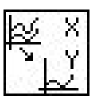
View All

This command is used at the start of G Code programs (Compute Level 4 & 5) in which you want to manipulate data sets that have already been uploaded to your project.

Example



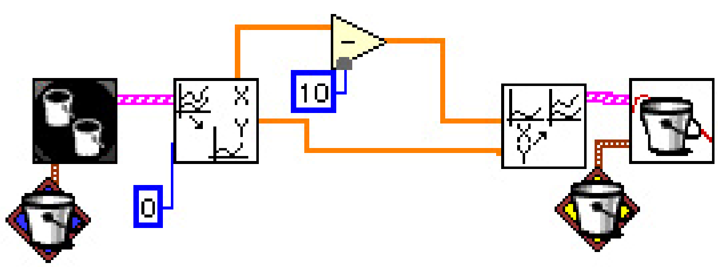
This piece of code takes data from the Red Bin, fits a line to it, and puts the plot in the Brown Bin.



Extract

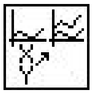
Use this command to extract the X and Y axis from a data set so that you can perform operations on the independent data.

Example



This piece of code extracts the first (0) data set from the blue bin, subtracts 10 from the X axis, recombines the 2 axes, and plots the result in the Yellow Bin.

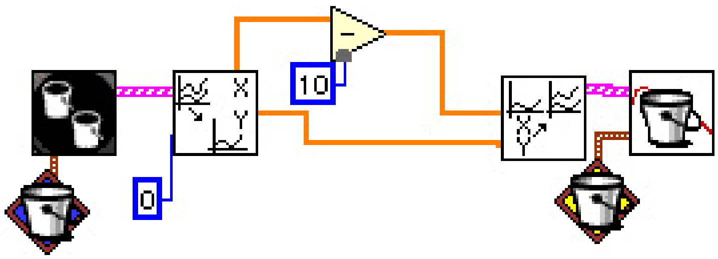
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



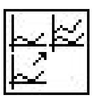
Combine

Use this to combine independent axes or arrays into a graph.

Example



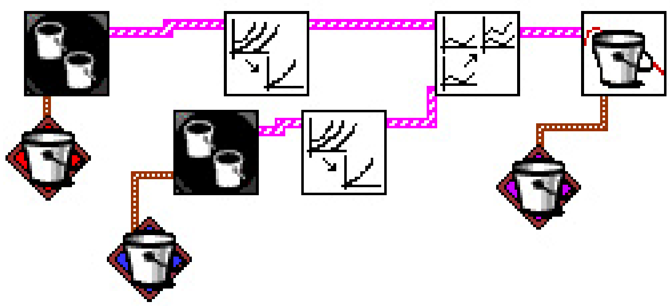
This piece of code extracts the first (0) data set from the blue bin, subtracts 10 from the X axis, recombines the 2 axes, and plots the result in the Yellow Bin.



Combine Bins

Use this command to put to separate plots on the same graph. This is useful for combining data averages collected from different sensors or for creating custom views of data.

Example



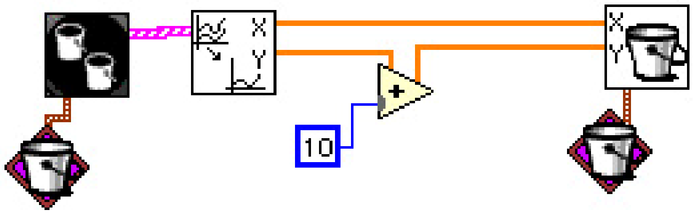
This piece of code averages all the lines in the Red Bin and averages all the lines in the blue bin. It then combines them into one plot that is plotted in the lavender bin.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

XY Plot

Use this to combine independent axes or arrays into a graph.

Example

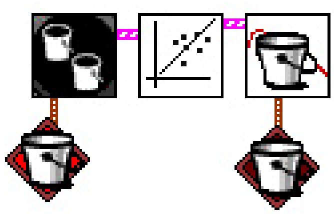


This piece of code separates the data in the lavender bin into separate X and Y arrays. 10 is added to the Y array and the result is plotted to the lavender bin. When viewed in the View and Compare area, the lavender bin will have 2 plots -- the original data and the adjusted data created with the code above.

Bin Plots

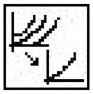
Use this command to put a plot into a bin.

Example



This piece of code would take data from the Red Bin, fit a line to it, and put the plot in the Brown Bin.

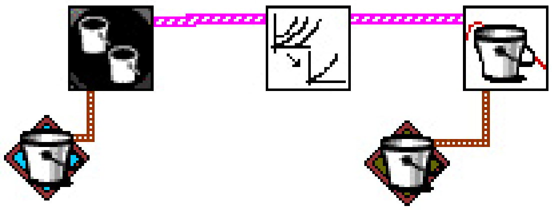
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



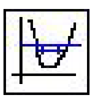
Average Lines

This command takes a bin with multiple plots and averages them into a single plot.

Example



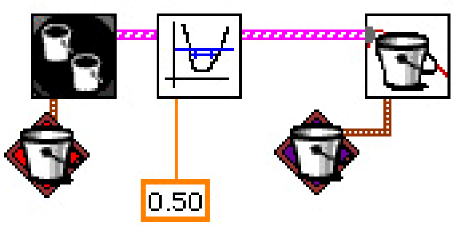
This piece of code creates a graph that is the average of all the lines in the light blue bin and plots it to the olive bin. Use the Compare template in the View and Compare area to see the original data in light blue bin and the averaged data in the olive bin.



Well Time

Well Time is the time between two peaks (peaks are defined as values above a threshold or cutoff value). Use this command to determine the amount of time between peaks (below the cutoff value) of a data set. If you have multiple peaks on your graph, this command will return a graph of the well times between each one. Useful for measuring for time between touch sensor presses or light sensor spikes.

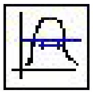
Example



This piece of code creates a graph of all the well times below the cutoff value for the red data set and plots it in

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

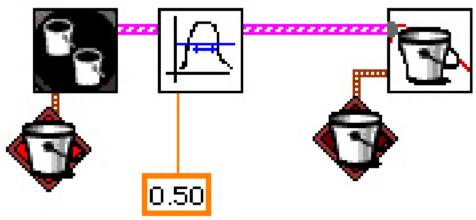
the purple bin.



Peak Time

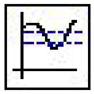
Peak time is the time between two wells (wells are defined as values below a threshold or cutoff value). Use this command to determine the amount of time between peaks (above the cutoff value) of a data set. If you have multiple peaks on your graph, this command will return a graph of the well times between each one. Useful for measuring for time between touch sensor presses or light sensor spikes.

Example



This piece of code creates a graph of all the peak times above the cutoff value for the red data set and plots it in the purple bin.

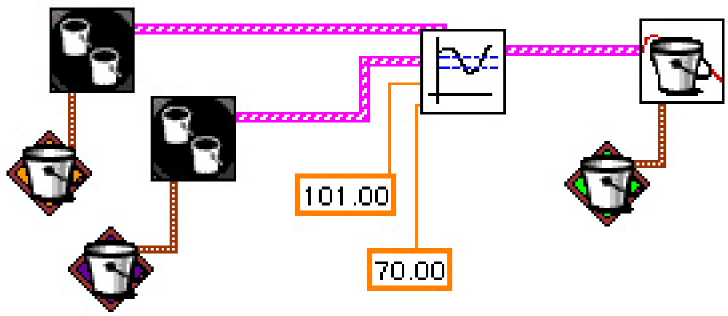
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



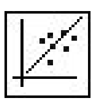
Threshold

The threshold command allows you to extract data from a data set based on bounds within the data set or bound of a second data set.

Example



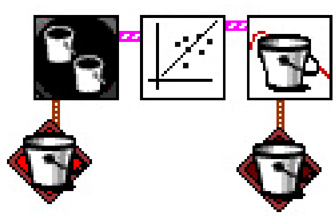
This piece of code extracts the values from the data set in the purple bin if the data in the orange bin is between 70 and 101 and plots the result to the green bin.



Fit Line

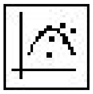
Use this command to fit a line to your data. If you need other information about your line (Offsets and Slopes) these can be displayed on the front panel.

Example



This piece of code takes data from the Red Bin, fits a line to it, and puts the plot in the Brown Bin.

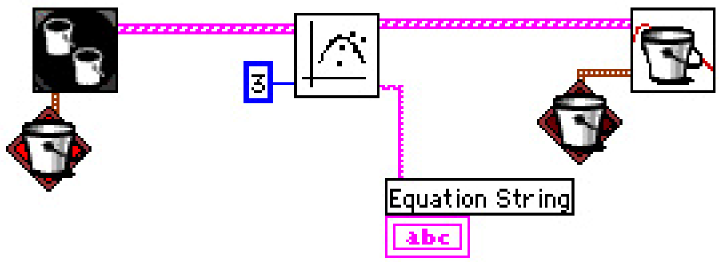
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Fit Curve

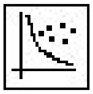
Use this command to fit a curve to your data. You can specify the curve order fit (using a numeric constant). The coefficients and equation of the line can be displayed on the front panel.

Example



This piece of code takes the data in the Red Bin, fits a 3rd order curve to it, and plots the result in the Brown Bin. The equation of the curve is displayed on the front panel using an indictor.

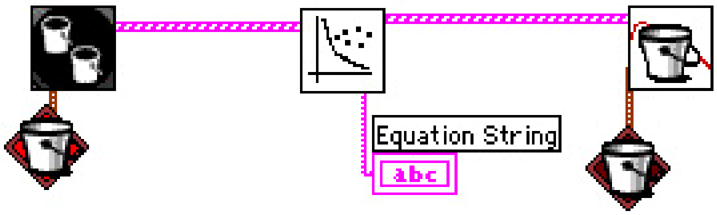
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



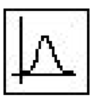
Fit Exponential

Use this command to fit an exponential curve to your data. The damping, amplitude, and equation of the line can be displayed on the front panel.

Example



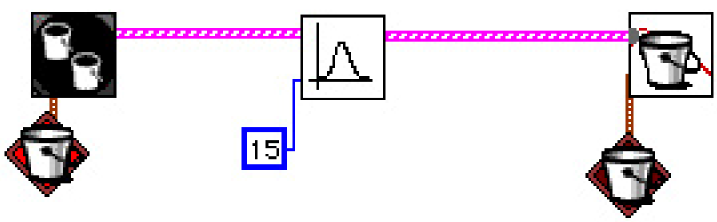
This piece of code takes the data in the Red Bin, fits an exponential curve to it, and plots the result in the Brown Bin. The equation of the curve is displayed on the front panel using an indicator.



Histogram

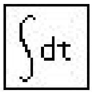
Histograms are useful for plotting the distribution of your data (how often you measured certain values or certain ranges of values).

Example



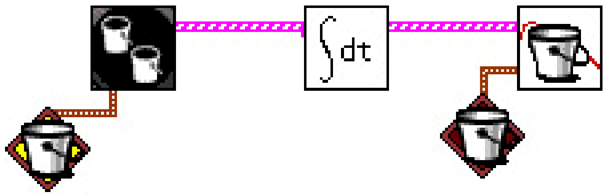
This piece of code takes the data from the Red Bin, creates a histogram of the data (using 15 bins), and plots the result to the Brown Bin.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

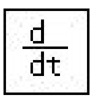


Integrate Use this command to integrate data. If multiple plots are input each one will be integrated separately.

Example

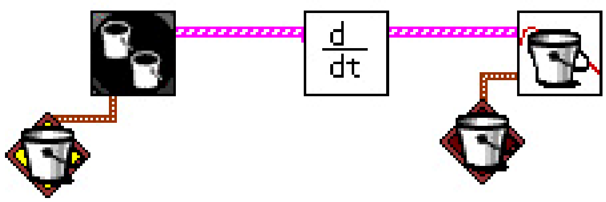


This piece of code takes the data from the Yellow Bin, integrates it, and plots the result in the Brown Bin.



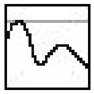
Differentiate Use this command to differentiate data. If multiple plots are input each one will be differentiated separately.

Example



This piece of code takes the data from the Yellow Bin, differentiates it, and plots the result in the Brown Bin.

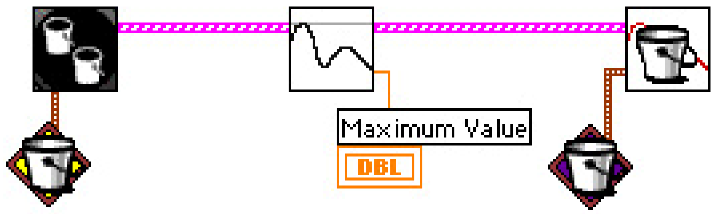
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



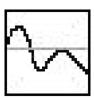
Maximum

Use this command to create a plot of the maximum value in a data set. The value can also be displayed on an indicator on the front panel.

Example



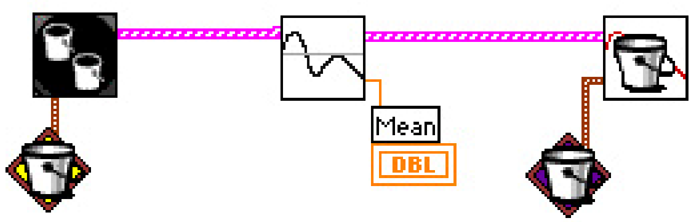
This piece of code takes data from the Yellow Bin, creates a plot of the maximum value, and plots the result in the purple bin. The maximum value is also displayed in numeric form as an indicator on the front panel.



Mean

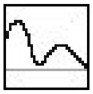
Use this command to create a plot of the mean value in a data set. The value can also be displayed on an indicator on the front panel.

Example



This piece of code takes data from the Yellow Bin, creates a plot of the mean value, and plots the result in the purple bin. The mean value is also displayed in numeric form as an indicator on the front panel.

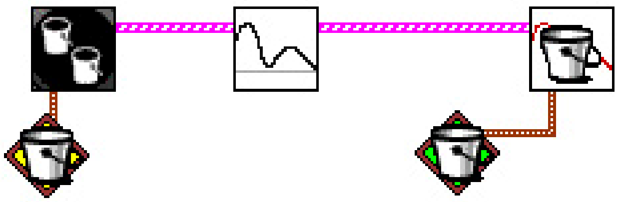
educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



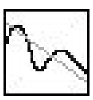
Minimum

Use this command to create a plot of the minimum value in a data set. The value can also be displayed on an indicator on the front panel.

Example



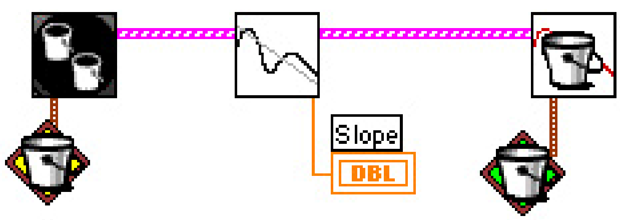
This piece of code takes data from the Yellow Bin, creates a plot of the minimum value, and plots the result in the purple bin.



Line Slope

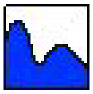
Use this command to create a plot of the slope of the data set. The value can also be displayed on an indicator on the front panel.

Example



This piece of code takes data from the Yellow Bin, creates a plot of the slope, and plots the result in the purple bin. The slope value is also displayed in numeric form as an indicator on the front panel.

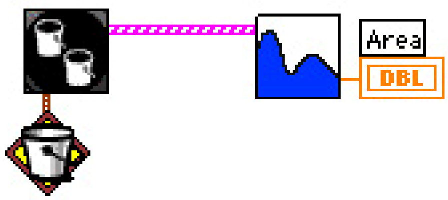
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



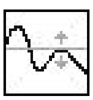
Area Under Curve

Use this command to find the area under the curves of a graph. The area is displayed best as an indicator on the front panel.

Example



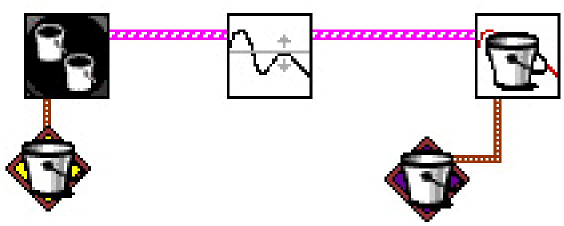
This piece of code takes data from the Yellow Bin and calculates the area under the curves. The result is displayed as an indicator on the front panel.



Standard Deviation

Use this command to find the standard deviation of data.

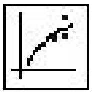
Example



This piece of code takes data from the Yellow Bin, calculates the standard deviation, and plots the result to the purple data bin. To view the original data and the standard deviation at the same time use the Compare template in the View and Compare section of Investigator.

LEGO educational division

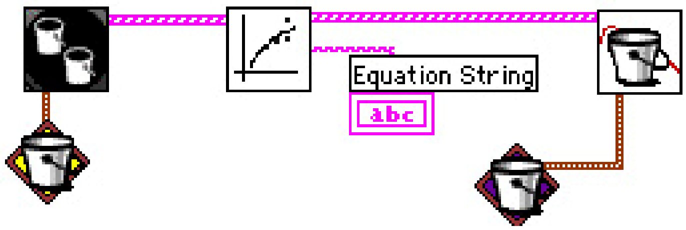
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Fit Ln

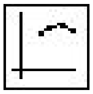
Use this command to fit a y=AlnX + B line to the desired Data Set. The equation of the fit line, offsets, and slopes can be displayed on the front panel using indicators.

Example



This piece of code takes data from the Yellow Bin, fits a y=AlnX + B line to the data, plots the result in the purple bin and displays the equation of the fit on the front panel. To view the original data and the fit data at the same time use the Compare template in the View and Compare section of Investigator.

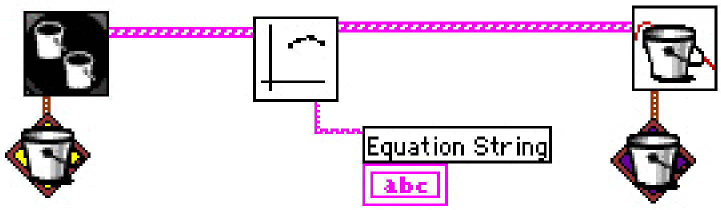
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



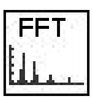
Fit Spline

Use this command to fit a spline to the desired Data Sets. Choose 'Spline fit to Data Sets' to display ONLY the spline fit or choose Original and fitted curve to display both. The coefficients and equation string can be displayed on the front panel using indicators.

Example



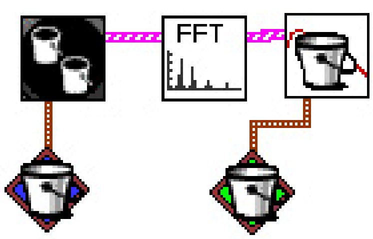
This piece of code takes data from the Yellow Bin, fits a spline to the data, plots the result in the purple bin and displays the equation of the fit on the front panel. To view the original data and the fit data at the same time use the Compare template in the View and Compare section of Investigator



Spectrum

Use this command to find the Power Spectrum of a data set. Useful for determining the frequency of a waveform in a data set.

Example



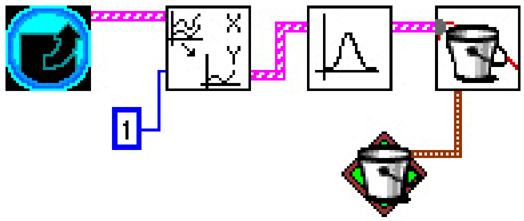
This piece of code takes data from the blue bin, finds the power spectrum of it and plots the result in the green bin

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

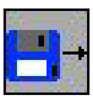
Upload Data

Use this command to upload data directly from the RCX. Useful if you often work in levels 4 & 5 of G-Code or if you are using data logging functions in Inventor.

Example



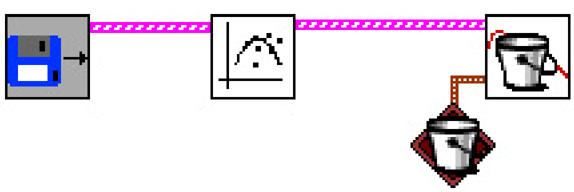
This piece of code uploads data from the RCX, takes the second data set, finds the histogram, and plots the result in the green bin.



Load Data

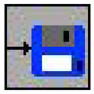
Use this command to load data into Investigator from an external file. When a program is run containing this command, you will be prompted to find the location of the file you wish to load. This can be useful if you want to transfer data from one Investigator project to another (export the data from one project and load it into another).

Example



This piece of code prompts the user to specify a file to load, fits a curve to it and plots the result in the Brown Bin.

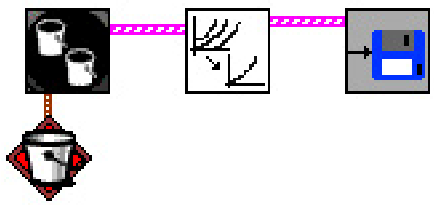
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Save Data

Use this command to save Investigator into an external file. When a program is run containing this command, you will be prompted to specify a name and location for the file you wish to save. This can be useful if you want to transfer data from one Investigator project to another (export the data from one project and load it into another).

Example



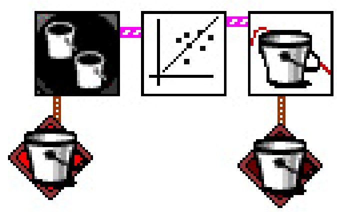
This piece of code takes data from the Red Bin, averages the line, and then saves the result to a file specified by the user.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.


All Bins

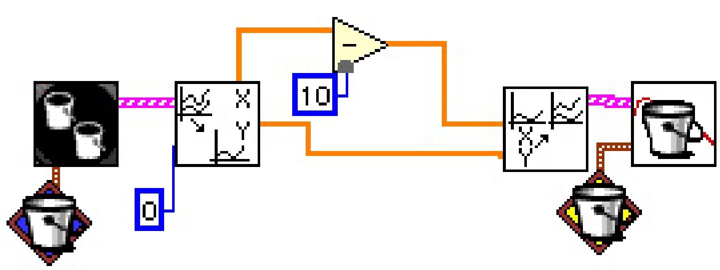
Use the bins to organize and retrieve data. Bins can contain multiple plots. It is best to use one bin per type of data you are collecting (Use the Red Bin for light data, the Blue Bin for temperature data etc...)

Example



This piece of code takes data from the Red Bin, fits a line to it, and puts the plot in the Brown Bin.

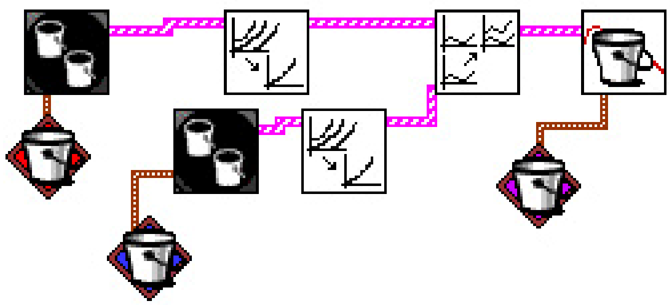
Example



This piece of code extracts the first (0) data set from the blue bin, subtracts 10 from the X axis, recombines the 2 axes, and plots the result in the Yellow Bin.

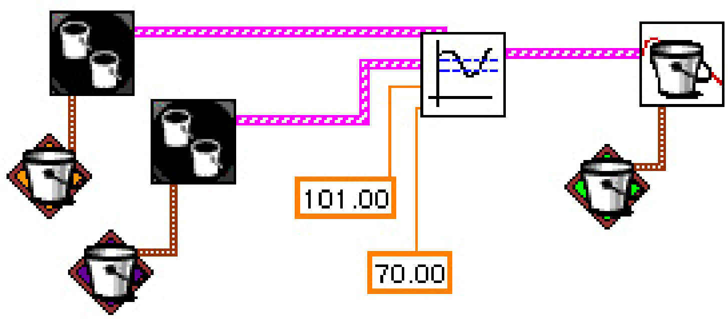
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Example



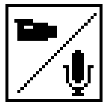
This piece of code averages all the lines in the Red Bin and averages all the lines in the blue bin. It then combines them into one plot that is plotted in the lavender bin.

Example

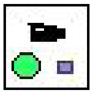


This piece of code extracts the values from the data set in the purple bin if the data in the orange bin is between 70 and 101 and plots the result to the green bin.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



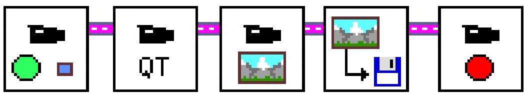
Multimedia



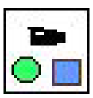
Init Small Image

Use this like a Start icon. This does NOT take pictures, it just initializes the camera. Use this if you want a smaller image.

Example



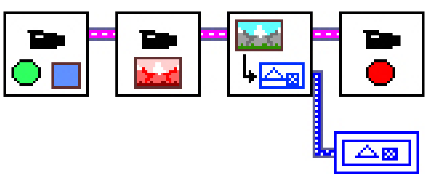
This piece of code initializes a small image, allows the user to adjust camera options, takes a color image, and saves it as a .bmp file.



Init Large Image

Use this like a Start icon. This does NOT take pictures, it just initializes the camera. Use this if you want a larger image.

Example



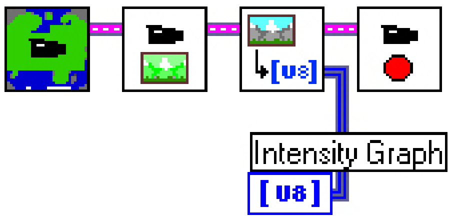
This piece of code initializes a medium image, snaps a red image, and converts that image to a picture to be displayed on the control panel.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

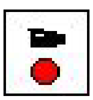
Init Internet Image

Use this like an Internet Start icon. This does NOT take pictures, it just initializes the camera. Use this if you want to take pictures over the Internet.

Example



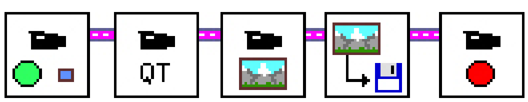
This piece of code initializes an Internet image, takes a green image from a remote camera, and converts it to an array which is displayed as an intensity graph on the front panel.



Close Camera

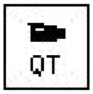
Use this like an End icon. This is required at the end of all camera program strings.

Example



This piece of code initializes a small image, allows the user to adjust camera options, takes a color image, and saves it as a .bmp file.

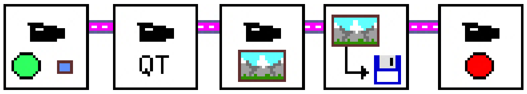
educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



Use this to set up camera properties, options, etc.

QT Setup

Example

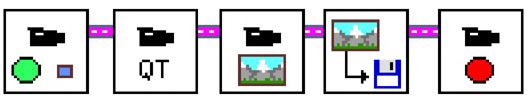


This piece of code initializes a small image, allows the user to adjust camera options, takes a color image, and saves it as a .bmp file.

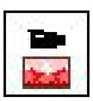
Use this to grab a full color image.

Grab RGB

Example



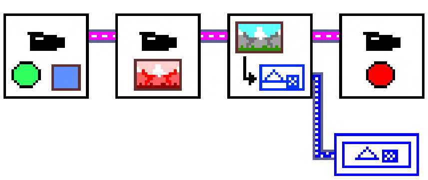
This piece of code initializes a small image, allows the user to adjust camera options, takes a color image, and saves it as a .bmp file.



Use this to grab a red image.

Grab Red

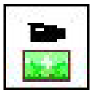
Example



This piece of code initializes a medium image, snaps a red

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

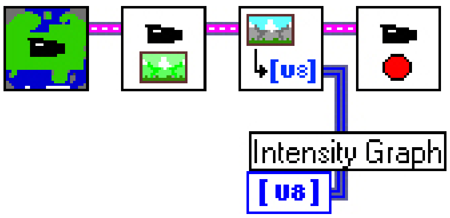
image, and converts that image to a picture to be displayed on the control panel.



Use this to grab a green image.

Grab Green

Example

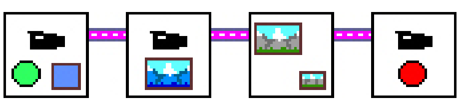


This piece of code initializes an Internet image, takes a green image from a remote camera, and converts it to an array which is displayed as an intensity graph on the front panel.

Use this to grab a blue image.

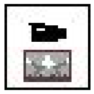
Grab Blue

Example



This piece of code initializes a medium image, takes a blue image, and shrinks it to a small image.

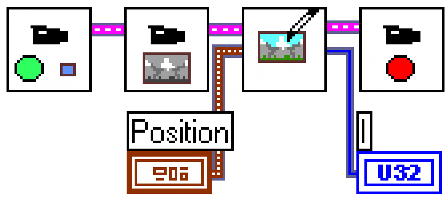
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



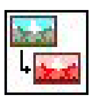
Use this to grab an image in grayscale.

Grab Grey

Example



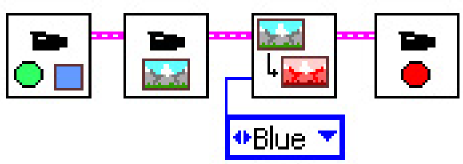
This piece of code takes a small gray image and then determines the value of the pixel in the specified position and prints it as a number to the I indicator.



Use this to extract a single plane (red, green, blue or grayscale) from a color image.

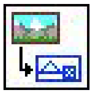
Extract Plane

Example



This piece of code takes a medium color image and extracts the blue plane from it.

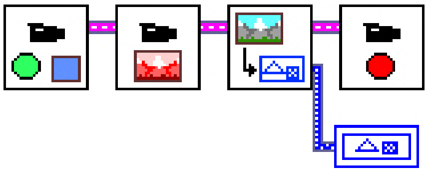
educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



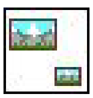
Convert to Picture

Use this to convert your image to a picture you can place on the control panel, etc.

Example



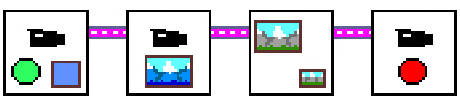
This piece of code initializes a medium image, snaps a red image, and converts that image to a picture to be displayed on the control panel.



Shrink Image

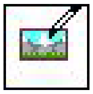
Use this to shrink image. Go from medium size to small size.

Example



This piece of code initializes a medium image, takes a blue image, and shrinks it to a small image.

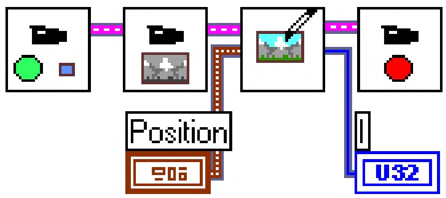
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



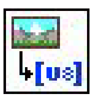
Get Pixel Value

Use this to determine the color of an individual pixel. If image is black and white the number will be 0 or 1. If image is color, the number will range from 0 to 255.

Example



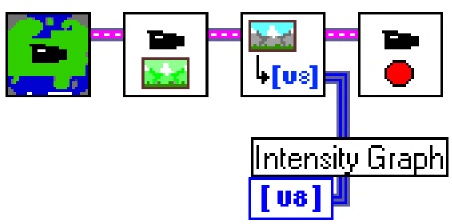
This piece of code takes a small gray image and then determines the value of the pixel in the specified position and prints it as a number to the I indicator.



Convert to Array

Images are strings of numbers (which represent colors) put together. Use this to turn an image into an array of numbers.

Example



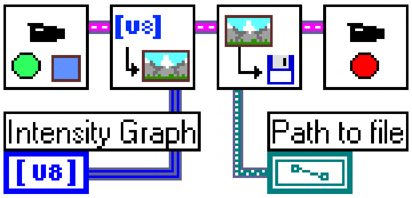
This piece of code initializes an Internet image, takes a green image from a remote camera, and converts it to an array which is displayed as an intensity graph on the front panel.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

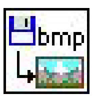
Convert to Image

Images are strings of numbers (which represent colors) put together. Use this to turn an array of numbers into an image.

Example



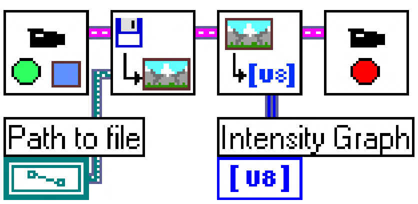
This piece of code reads an array from an intensity graph and converts it into an image. That image is then saved as a .bmp file along the indicated file path.



Read BMP

Use this to read a bitmap (.bmp) file and create an image.

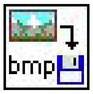
Example



This piece of code reads a .bmp file from the designated file path and converts it to an image. It then converts that image into an array which is displayed on the front panel as an intensity graph.

LEGO educational division

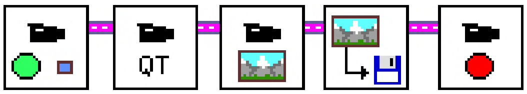
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



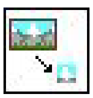
Save BMP

Use this to save your image as a bitmap (.bmp) file.

Example



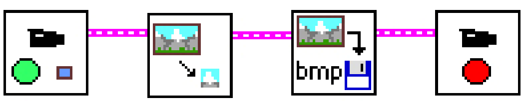
This piece of code initializes a small image, allows the user to adjust camera options, takes a color image, and saves it as a .bmp file.



Get Image Subset

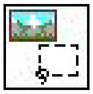
Use this icon to select a subset of an image you have grabbed.

Example



This piece of code saves a subset of the image to a .bmp file.

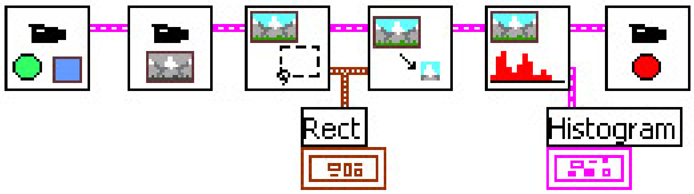
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



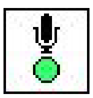
Select ROI

Use this icon to select a region of interest in an image you have grabbed.

Example



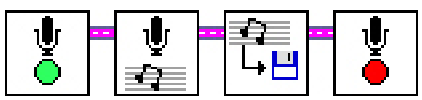
This piece of code grabs a grayscale image from the camera, allows the user to select a region of interest (ROI), extracts that subset from the image and displays the histogram of that subset on the front panel.



Init Mic

Use this like a Begin icon. This does NOT grab sound, it only initializes the microphone.

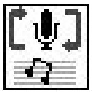
Example



This piece of code initializes the microphone, grabs a sound, and saves it as a .wav file.

educational division

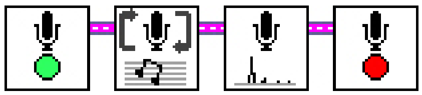
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



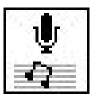
Grab Sound Continuously

Use this to continuously grab sounds from the microphone. This is the same as running a grab sound command in continuous mode.

Example



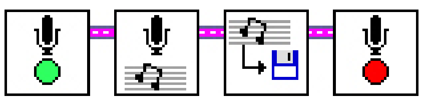
This piece of code initializes the microphone, continuously grabs a sound and analyzes its frequency.



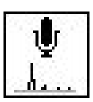
Grab Sound

Use this to grab a sound from the microphone.

Example



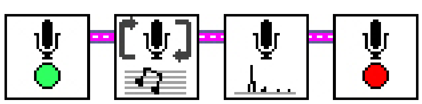
This piece of code initializes the microphone, grabs a sound, and saves it as a .wav file.



Frequency Analysis

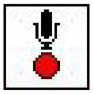
Use this to analyze a sound you have taken.

Example



This piece of code initializes the microphone, continuously grabs a sound and analyzes its frequency.

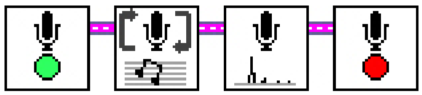
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



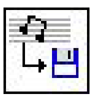
Close Mic

Use this like an End icon. Each string of a microphone program will need one of these.

Example



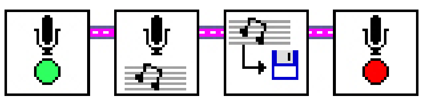
This piece of code initializes the microphone, continuously grabs a sound and analyzes its frequency.



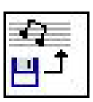
Save Sound

Use this to save a sound from the microphone as a .wav file.

Example



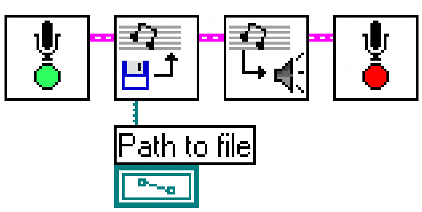
This piece of code initializes the microphone, grabs a sound, and saves it as a .wav file.



Load Sound

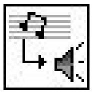
Use this icon to load a sound from a .wav file.

Example



©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

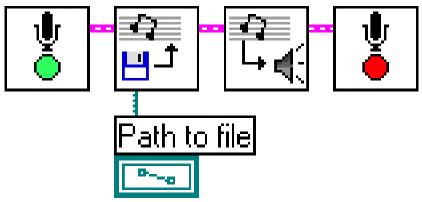
This piece of code initializes the microphone, loads a sound from the indicated file path, and then plays it.



Play Sound

Use this icon to play a sound you have grabbed or saved.

Example



This piece of code initializes the microphone, loads a sound from the indicated file path, and then plays it.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

3. How To…

The sections of this chapter explain how to perform a selection of more advanced ROBOLAB tasks and features.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

3.1 Using Vision Sensors Outside of Vision Center

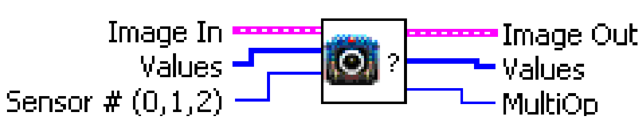
Using Vision Center, you can create custom vision sensors. These sensors use various image processing functions to reduce an image down to a single number. This number may represent one of many image properties (e.g. the area of white space in a portion of an image, the number of blobs in an image, etc.). With Vision Center running, you can assign a vision sensor (one of the 8 default sensors, or a custom sensor you create) to a container on the RCX. With the RCX in view, you can continually update this container value with the value of your vision sensor as the image changes. This way the RCX's actions can be linked to what the Lego camera sees while connected to your computer.

Another useful tool is to implement vision sensors without having Vision Center running. Once you understand how Vision Center works, you can create an Inventor program that grabs an image, applies your sensor to that image, then updates the value on the RCX using direct mode. This method allows you to process images from within your own program, without using Vision Center.

Vision Sensors

Vision sensors are saved the *Vision → Video_Sensors* (by default - see section 3.3 for the location of your Vision folder). To include a vision sensor in your Inventor program, use **Select a vi…** and browse to the folder where your sensor is saved (you will have to look for "All File Types"). When you have dropped the sensor on the diagram of your program, it will look like this:

Select a vi…



©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Of the available connections, you will need to use **Image In, Image Out, Sensor # (0,1,2)** and **Values.** The image on which the sensor operation will be performed should be connected to **Image In**, and the processed image is returned in **Image Out.**

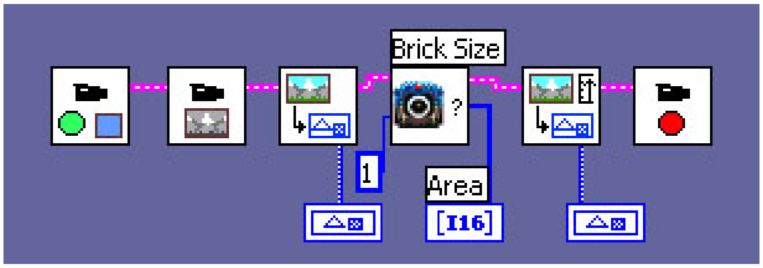
Values is an array of three numbers that contains the result of the vision sensor. If your sensor measures either **Max Info** or **Min Info** the array will be filled with the three values that represent this result:

1. Pixel Intensity (0-255)
2. Row number where the maximum (or minimum) value occurred
3. Column number where the maximum (or minimum) value occurred

If the sensor is of the type that returns only one number (pixel sum, pixel average, blob count, etc.), **Sensor # (0,1,2)** will determine which index of the **Values** array gets updated with the result.

Example 1

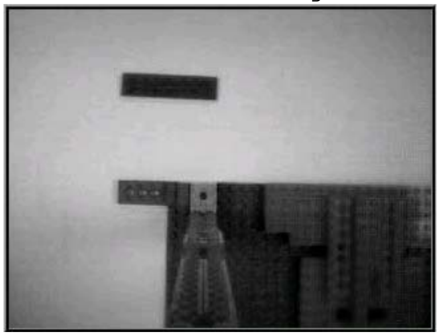
This program demonstrates how to include a custom vision sensor in an Inventor program. The custom sensor used here is called "Brick Size," was created in vision center, and performs a blob area operation on a masked portion of the image where a Lego brick is known to be. The image is also converted to a ROBOLAB picture before and after the vision sensor to clarify what is going on.



On the front panel of this program, we therefore have a picture of the grabbed image, the processed image and the sensor result (in this case, the area of the blob located within the mask). Here a 2x8 brick is placed in front of the camera, and the resulting area is shown, in pixels:

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Grabbed Image

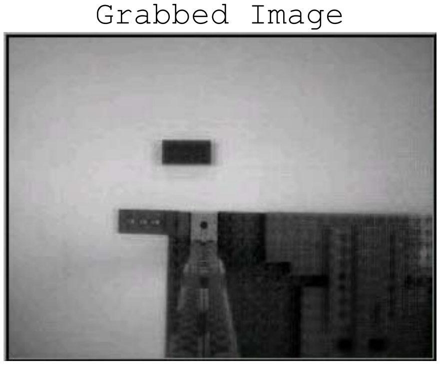


Processed Image

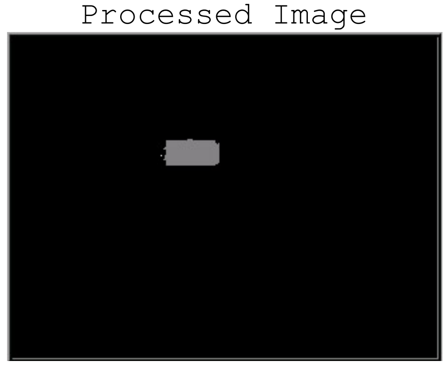
Result

If we place a smaller brick (2x4) in front of the camera, we get the corresponding images and result for the area, which we would expect to be approximately half that of the 2x8 brick:

Grabbed Image



Processed Image

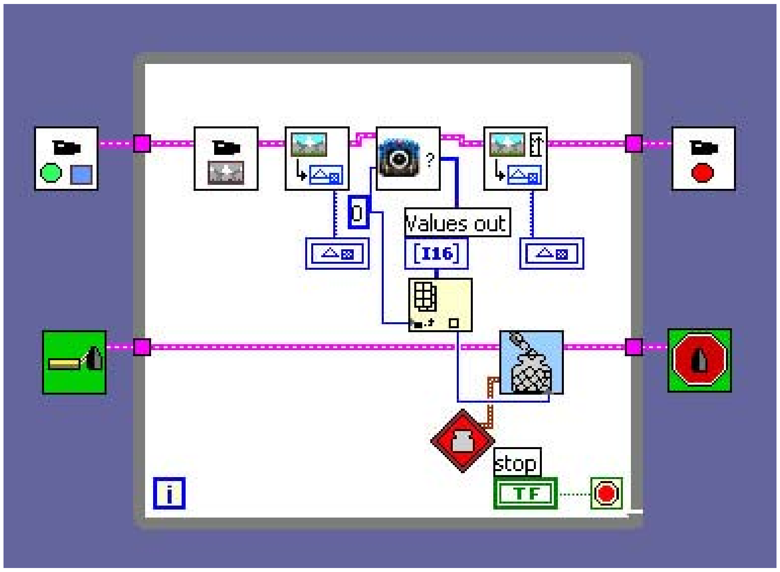


Result

educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Example 2

This program shows how to download the information received from the vision sensor to a container on the RCX. This example uses Direct Mode to continually update the Red Container on the RCX. It also refreshes the image once per loop and applies the vision sensor to the new image:



See Vision Center and Direct Mode sections for more information on creating vision sensors and communicating with the RCX.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

3.2 Containers in ROBOLAB

There are 47 available containers (variables) on board the RCX. The following table outlines how they are generally allocated in ROBOLAB. It is recommended that most users only utilize 0 – 22 for programs. Addressing containers above 22 can result in conflicts and execution problems. Only advanced users should work directly with variables above 22.

0	Red Container
1	Blue Container
2	Yellow Container
3 -22	Generic User Containers
23 – 25	Data Pad (Red, Blue, Yellow) Values (last value logged)
26 – 28	Data Pad (Red, Blue, Yellow) Counts (number of points acquired)
47	Wait for brightness or darkness
47	Play System Sound
46/47	Wait for Angle
33 - 47	For Loop iterations (Accessible through the Loop Counter Container Modifier

- Variables above 22 cannot currently be directly displayed on the RCX’s LCD panel

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

3.3 User Data in ROBOLAB

ROBOLAB user data and examples are stored in a set of folders that allows the user to organize their information and the ROBOLAB software access customized settings and programs.

User Data

Program Vault (All Versions of ROBOLAB)

The Program Vault contains programs created in the Programmer section of ROBOLAB in Pilot and Inventor. Programs are stored in the *My Programs* folder of the section and level they were created (i.e. A level 3 Inventor Program would be stored in *Program Vault->Inventor->My Programs->Level 3->myprogram.vi*). Examples that correspond to different LEGO sets are also stored in the Program Vault under the corresponding theme names.

Investigator Themes (ROBOLAB 2.0 and higher)

Investigator Themes contains the programs and data created in the Investigator section of ROBOLAB. The folder contains two sub folders – *Examples* and *My Projects*. *Examples* contains the locked sample programs that ROBOLAB ships with. My Projects is where user programs are stored.

HTML (ROBOLAB 2.0 and higher)

Within Investigator, users can publish their project as a set of web pages with pictures of their pages and a downloadable version of their project. Published projects are stored in folders in the *HTML* project (folders are named with the original project and the date the web pages were created).

Songs (ROBOLAB 2.5 and higher)

The songs and sounds shipped with ROBOLAB as well as user created songs (generated from Piano Player) are stored with in the Song folder. The subfolders Classical, SoundFX, and Misc contain the music and sounds shipped with ROBOLAB. Users should save the songs they create to the subfolder My Songs. Songs saved in the My Songs folder appear in the Pilot pull down menu of song choices.

LEGO educational division

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Vision (ROBOLAB 2.5 and higher)

The Vision folder contains two subfolders Pictures and Video_Sensors. The Pictures subfolder contains any images acquired by using the snap or grab pictures buttons or commands (in Pilot, Vision Center or the corresponding Inventor icon). The Video_Sensors subfolder contains the default Video Sensors that ROBOLAB ships with that are used in Vision Center. Any user created Video Sensors should also be saved in this folder so that they can be used in Vision Center.

Path to User Data

As ROBOLAB has evolved the method and structure of storing data has changed. To see where your user data is stored check the paths for your version.

REMINDER: Always Back UP DATA before upgrading or installing new versions of ROBOLAB

ROBOLAB 1.0-2.0

User Data is stored directly in the ROBOLAB folder in the Program Vault, Investigator Themes, and HTML folders. All user data is saved to these folders unless otherwise specified.

ROBOLAB 2.5.0-2.5.2

The User Data is stored within the ROBOLAB folder in a folder entitled My Data (which contains the folders Program Vault, Investigator Themes, HTML, Vision, Songs). All user data is saved to these folders unless otherwise specified.

ROBOLAB 2.5.3 and higher

ROBOLAB 2.5.3 was designed to accommodate multiple users and security settings. User data is stored in a folder entitled ROBOLAB Data. The folder is placed into the location designated by the system for user data to be saved (on the PC this is most often My Documents and on the MAC it is typically in the user's documents folder). If the logon to a computer has been customized to use a network location for saving user data, ROBOLAB will use that location for the ROBOLAB Data folder.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

4. Customizing ROBOLAB

ROBOLAB has been designed to support the widest audience of users possible. While the majority of the features are configured to make it easy for the majority of users to work with, many elements of the software can be customized or altered by advanced users by editing text files or writing some code. These alterations range from the very simple, like changing ROBOLAB to use another LEGO interface by default, to the creation of custom sensors and palettes.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

4.1 Editing the Default.prf file

The Default.prf file is a text file that holds preferences for the copy of ROBOLAB. It contains the information on which port is being used for communication, the path for saving program data, and many other items. The file is created the first time ROBOLAB is run after a new install (if you upgrade an old installation your Default.prf file will be preserved) and is updated every time ROBOLAB is run.

The Default.prf file is located in different placed depending on your version of ROBOLAB.

Version 1.0 – ROBOLAB->Vi.lib->Splash->Default.prf
Version 1.5 – 2.0 – ROBOLAB->Default.prf
Version 2.5 – ROBOLAB->Engine->Default.prf
Version 2.5.3 -> The default.prf is located in the ROBOLAB Data folder (see section 3.3 for location explanation)

The Default.prf file can be edited in any basic text editor (Notepad, Simpletext, Wordpad). However, the file should only be edited by advanced users or those trying to fix a communication problem. Most of the settings can be altered from within ROBOLAB. ROBOLAB should NOT be running while you are editing the Default.prf file. If while editing the file, you manage to corrupt the file or are unable to correctly configure a setting, the file should be deleted. A new file will be created the next time ROBOLAB is run with the default factory settings.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

```
Preference File: Last modified at 15:34 on 04/12/2001
Path to Program Vault:=My Computer:Robolab:My Data:Program Vault
Locked:=TRUE(hi)
Port:=2
Admin:=TRUE(hi)
Power:=FALSE(lo)
Powerdown Time:=15
Number of Retries:=10
Serial Wait Time (msec):=10
Path to Investigator Vault:=LGM I:Desktop Folder:Robolab94:My
Data:Investigator Themes
Network Address:=somewhere.outthere.com
Network Port:=3237
Init each time:=TRUE(hi)
Bytes/Upload:=10
Level 3 Steps:=3
Investigator State:=1
ID:=2
(0=Error, 1=Text, 2=RCX,  3=Scout, 4=Control Lab)
Camera:=KritterUSB
```

Sample Default.prf File

Summary of Tags

Preference File:
Indicates the last time ROBOLAB modified the file. It should not be edited outside of ROBOLAB.

Path to Program Vault:
Indicates where User Data is stored on the computer. This tag is changed from within ROBOLAB in Administrator in the ROBOLAB settings section.

Locked:
Indicates whether or not Programs 1 & 2 on the RCX are locked (new programs are unable to be downloaded to these slots). This tag should be changed from within ROBOLAB in Administrator in the RCX Settings section.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Admin:
Specifies whether the Administrator button on the front panel is visible or not. This tag can be changed in the main section of Administrator.

TRUE(hi) – means the Administrator Button is visible
FALSE(low) –means the Administrator Button is NOT visible

If the button is hidden, hit F5 when in the main ROBOLAB splash page.

Power:
Specifies whether the RCX is transmitting IR signals in high or low power mode. This setting can be changed from within ROBOLAB in Administrator in the RCX Settings Section
FALSE(low) – RCX is transmitting in low power mode
TRUE (hi) - RCX is transmitting in high power mode

Powerdown Time:
Specifies the amount of time (in minutes) that the RCX will remain on before shutting down. This setting can be changed from within ROBOLAB in Administrator in the RCX Settings Section.

Number of Retries:
Specifies the number of times ROBOLAB will attempt to talk with the RCX via the tower. If ROBOLAB completes the number of retries without successfully communicating an error will be reported. This number can be increased if you are having difficulties communicating with the tower (it can also be decreased for those wanting the RCX communication to time out faster, for instance when the RCX is moving in and out of tower range).

Serial Wait Time (msec):
Specifies the time between retries. This number should be increased if you are working with a slower computer that is having difficulties with the serial communication.

Path to Investigator Vault:
Specifies the path to the folder where Investigator Projects are stored. This tag is changed from within ROBOLAB in Administrator in the ROBOLAB Settings section.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Network Address:
Specifies the default address used to communicate when Internet mode is selected from Select COM Port in the Project Menu or from the Administrator menu. This should be changed in Select COM port.

Network Port:
Specifies the TCP/IP port over which Internet commands are sent and received. Don't change this number or you will not be able to communicate via the Internet with ROBOLAB software.

Init each time:
Allows you to configure the serial port to every time ROBOLAB – RCX communication is attempted. Mainly a debugging feature for older PC computers and is not used in the latest versions of ROBOLAB.

Bytes/Upload:
The computer sends the information down to the RCX in batches. This tag specifies the amount of information (in bytes) in these batches. This number can be decreased if you are having difficulties communicating with the RCX or increased to make the download speeds faster. The larger the batch size the faster the download if the tower and RCX are communicating well. If batches have to be resent, large batch sizes will slow the communication down.

Level 3 Steps:
Specifies the number of steps in Investigator Program Level 3. This number can be increased if you wish to have additional steps in this Pilot-like interface.

Investigator State:
Specifies the component (Programmer or Investigator) of ROBOLAB that is loaded into memory when ROBOLAB first starts up.

ID:
Specifies which mode ROBOLAB is running in.
0=Error ~ this mode creates a text log that can be used to troubleshoot problems
1=Text ~ this mode outputs lists commands in text form instead of downloading them to an interface

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

2=RCX ~ this mode configures all communication and settings for the RCX
3=Scout ~ this mode configures all communication and settings for the Scout
4=Control Lab ~ this mode configures all communication and settings for the Control Lab Interface

If the user sets this tag to be 4 (Control Lab mode), for example, then all of the Pilots, Investigator pages, and Inventor programs will work directly with the Control Lab Interface instead of the RCX.

Camera:
Specifies which camera is being used by ROBOLAB. This can be selected from within ROBOLAB by choosing the 'Select Camera' option from the project menu.

Note: If there are any problems with any of these commands simply delete the line and ROBOLAB will replace the line with the default factory setting

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

4.2 Creating your own Sensor Definition Files

Sensor definition files allow ROBOLAB to be correctly configured for a variety of different sensors. These files provide all the information needed to collect and display data – from sensor name and the units measured to calibration equations and tables.

What makes up a sensor definition file?

```
Name:=ProTemp LogIT
Number:=200
Units:=Celsius
Resolution:=1
Make:=DCP
Types: 0:none 1:switch 2:Temp 3: Reflect 4: Angle 5: ID0 6: ID1
7:ID2
Modes: 0:raw 1:Boolean 2:Transition 3:Period 4:% 5: Celsius
6:Fahrenheit 7:Rotations
Type:=3
Mode:=0
Date of Calibration:=12    22    1999
Reset Each Time:=FALSE
Calib Type:=table
BeginTable
0     1.500000e+2
1     1.500000e+2
2     1.500000e+2
3     1.500000e+2
4     1.500000e+2
5     1.500000e+2
6     1.500000e+2
7     1.500000e+2
8     1.500000e+2
1023  -5.500000e+1
EndTable
Date of Calibration:=1  28    1999
Reset Each Time:=FALSE
Notes:= This is for a DCP/LogIT ProTemp Temperature sensor
```

Name:
Specifies the name of the sensor.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Number:
Gives each sensor a unique number. This number is used in upper level programming to specify the sensor in structure commands (wait fors, loops etc…)

Resolution:
Increments the sensor measures in.

Make:
Manufacturer of the SENSOR (currently ROBOLAB officially supports sensors made by LEGO and DCP Micro. However, the RCX and ROBOLAB are capable of supporting any sensor that requires 5v power).

Types:
This item simply lists the different kinds of sensors that can be specified in the Type: field.

0:none
1:switch – Un-powered measurement.
2:Temp
3:Reflect – Powered (5v) measurement.
4:Angle
5:ID0
6:ID1
7:ID2

Modes:
This simply lists the different modes in which a sensor may operate. This value is specified in the Mode: field.

0:raw – Sensor operates on the full scale of the 10-bit A/D, and will return values from 0-1023 (for custom sensors)
1:Boolean – Sensor returns only 1 or 0 (LEGO touch sensor)
2:Transition – Sensor returns 1 or 0 based on whether a transition between high and low has occurred (LEGO touch and release sensor)
3:Period
4:% - Sensor return value is a percentage of the total range of the 10-bit A/D (0-100)

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

5: Celsius – Sensor return value is the total range converted to a Celsius measurement
6:Fahrenheit - Sensor return value is the total range converted to a Celsius measurement
7:Rotations – Sensor value is the total range converted into a Rotations measurement (16 per revolution)

Type: Define the sensor type (e.g., Type:=1)

Mode: Define the sensor mode (e.g., Mode:=0)

Calib Type:
This field specifies the type of calibration used to convert the measured data into the required units. The two options are Curve (applies a linearly interpolating equation to the measured data) and Table (applies a lookup table you specify to the measured data).

Calibration: This allows you to specify the line equation if **Calib Type** is Curve or the lookup table if **Calib Type** is Table.
If you are using the Curve type, specify the **Calibration** field as follows:

Calibration (a0+a1X+….):=number1   number2

Where number1 is the y-intercept (or b from mx+b) and number2 is the slope (or m from mx+b). These two numbers should be separated by a tab.

If you are using the Table type, specify the **Calibration** field as follows:

```
BeginTable
0    value0
1    value1
2    value2
…    …
1022 value1022
1023 value1023
EndTable
```

Where 0,1,2,…,1022,1023 are the measured data and value0, value1, …, value1022, value1023 are the values of the

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

lookup table. Each number-value pair should be separated by a tab, and each line should be separated by two tabs (see 200Temp.sa in your ROBOLAB→Engine→Sensors folder for an example). The effect of this table is to assign a specific value for each of the possible measured values from the sensor. If value0…value1023 were all set to 4, for example, the sensor would always return the value 4, no matter what value it measured.

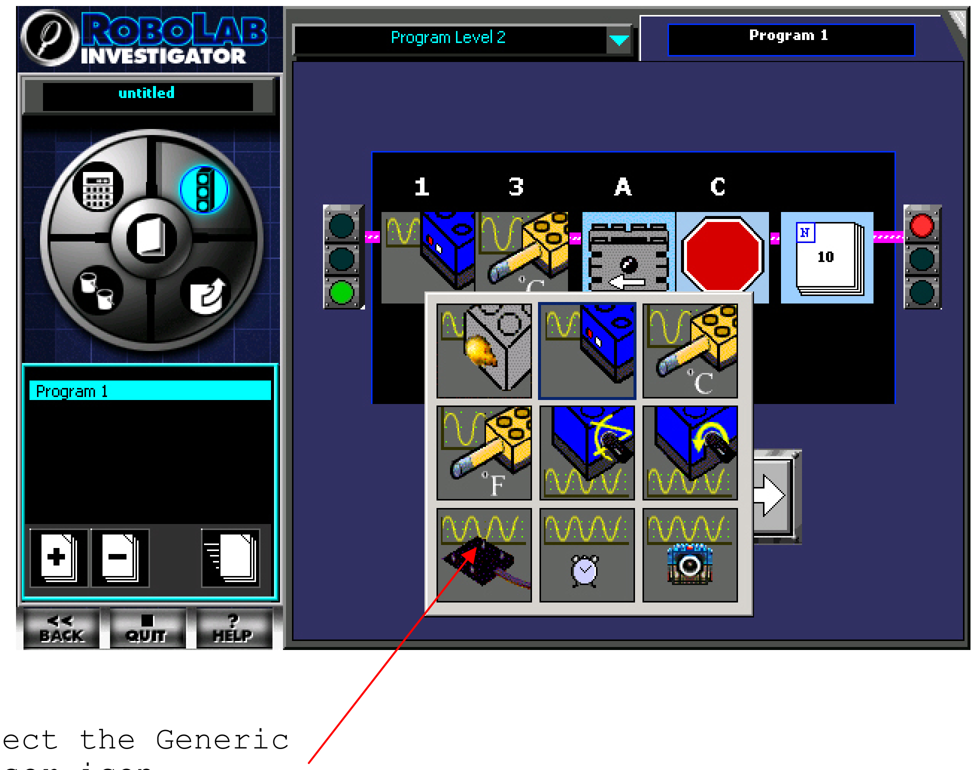
Date of Calibration:
Specifies the data the calibration was last changed (updated manually) in mm dd yyyy format

Reset Each Time:
This field specifies whether the sensor is reset

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Using your new sensor definition file in Investigator

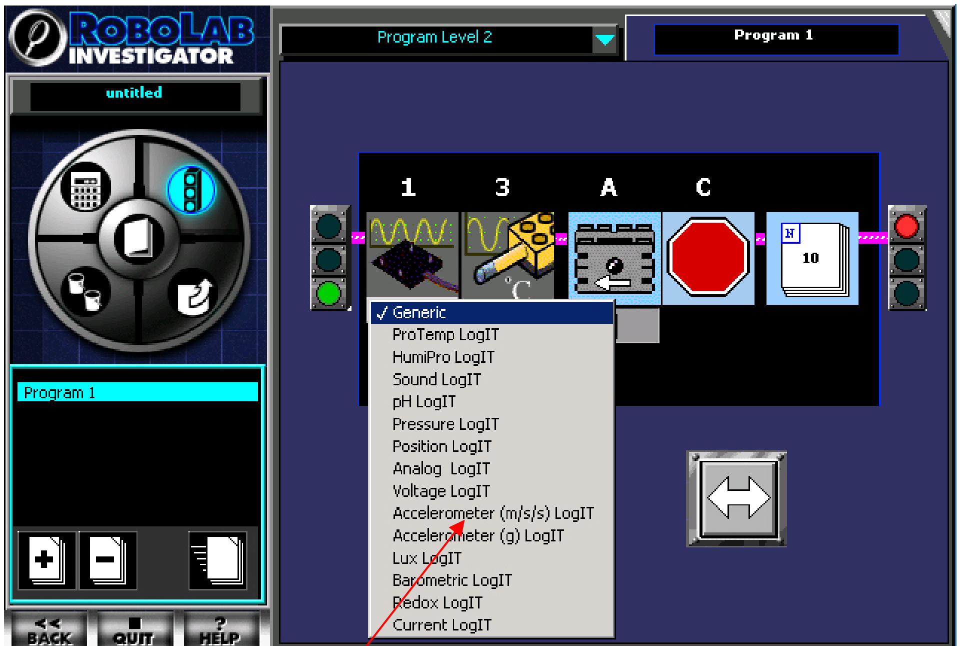
You can datalog a custom sensor in Program Levels 1, 2 and 3. You do this by first selecting the generic sensor icon:



Select the Generic Sensor icon

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Then you can select your custom sensor adapter from the list of available sensor definition files:



Select your sensor adapter from the list

For information on how to include your custom sensor adapter in Inventor programs, please visit:

www.lego.com/eng/education/mindstorms or
www.ceeo.tufts.edu/robolabatceeo

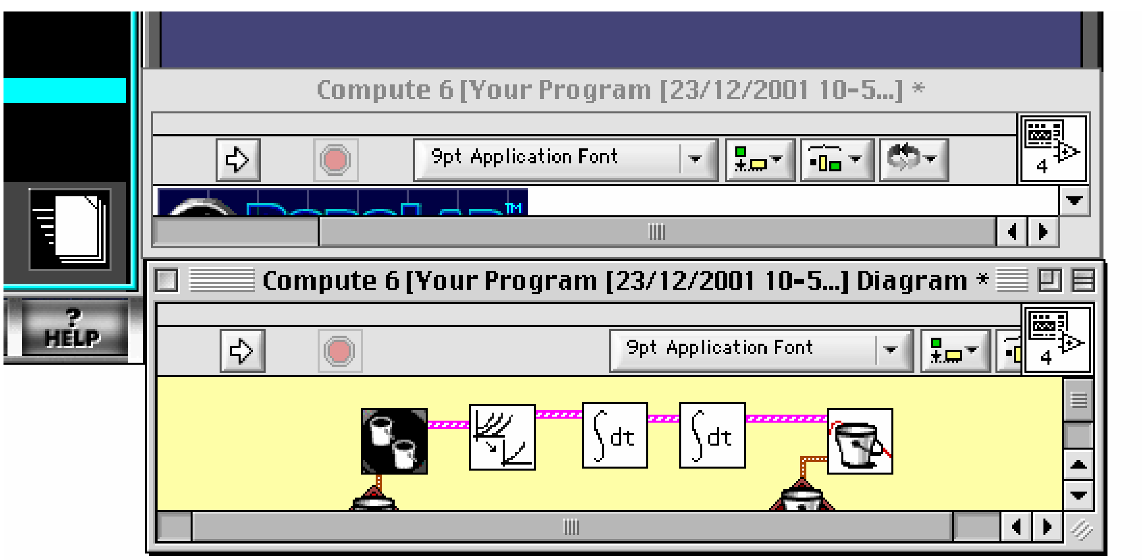
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

4.3 Customizing Investigator

There are two levels of creating custom features in the Investigator component. The first (and easier) method does not require any LabVIEW programming skills. The second method requires working knowledge of LabVIEW.

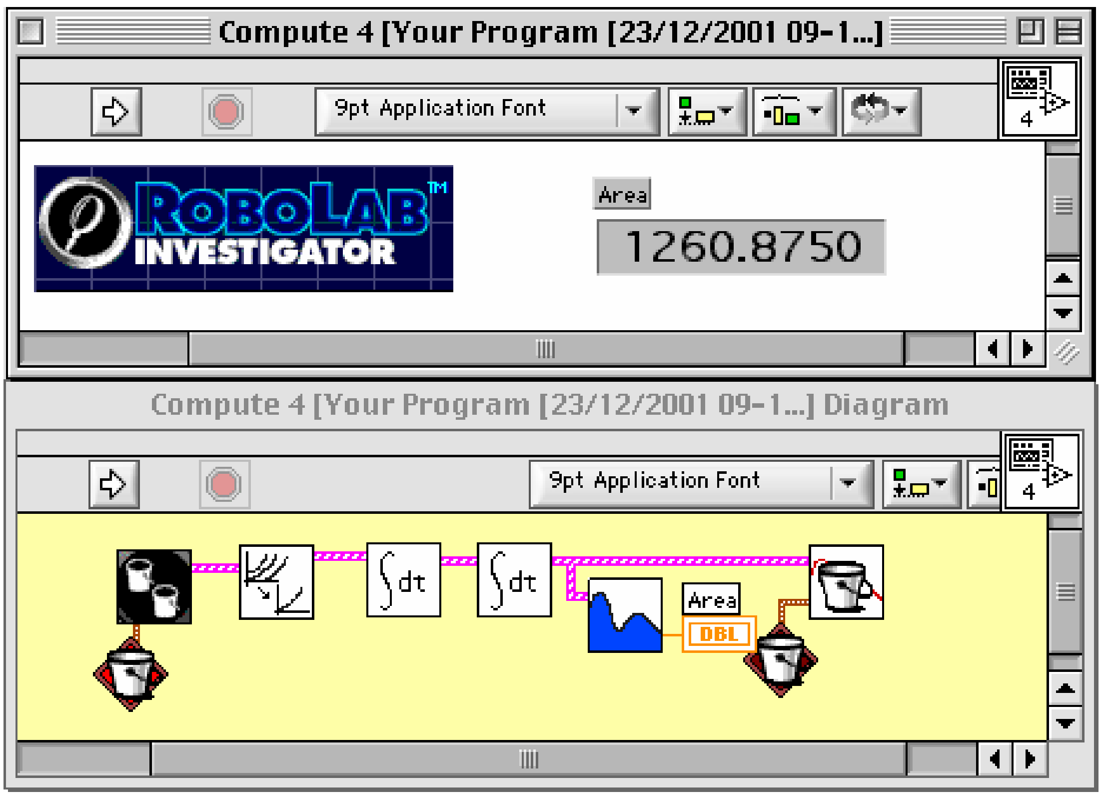
The Simple Way

If you simply want to have some customized output from a compute page, you can start using the front panel on a compute level 4 or 5 palette. Every vi (or virtual instrument) has two parts: a panel and a diagram. Inventor level coding is all based only on the diagram and you never use the panel. Pilot levels are all just panels with some rather complicated Inventor-style diagrams. If you start up a compute 4 page by clicking on: and move the diagram down by moving the title bar. You should have something



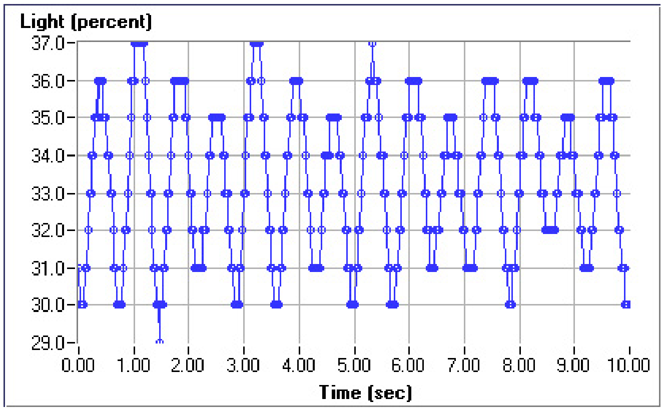
educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

like: The upper long and thin box is the panel. Drag it up and make it bigger so that you can see both a panel and a diagram. Next add on the icon that finds the average area and you can add an indicator on the front panel to show that area. The easiest way to do this is to right click (or click while holding the apple key down) on the lower right corner of the icon and select "Create an Indicator". The help menu (Ctrl H) would tell you that the lower left corner of the icon gives the calculated area. You can move your indicator around on the panel and getting the paint tool (hit the tab key a few times to get it) will allow you to change the color of the panel as well. The completed customized palette should look like:



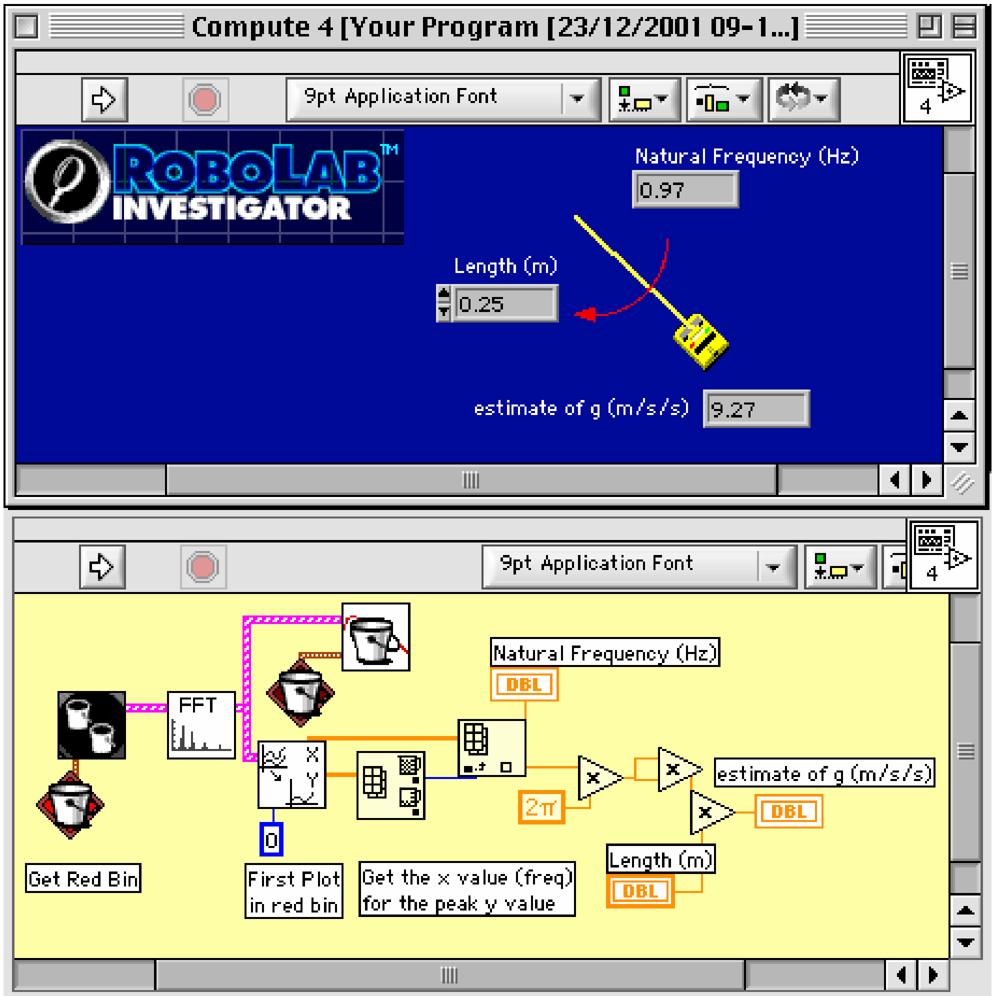
LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

For instance, if you had measured the oscillations of a pendulum by putting a light sensor on the RCX and then swinging the whole thing you would get data that looks something like:

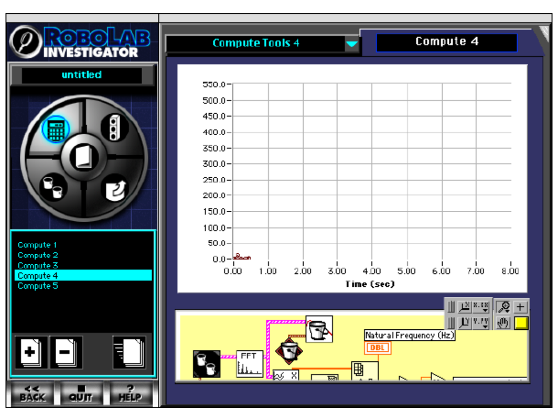


You can then use a spectral analysis to determine the natural frequency, find the peak, and then use a little math to calculate and display g: the gravitational acceleration. Your final panel and diagram might look like the figure below. You can cut and paste pictures onto the panel.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.



The only real disadvantage of this method is that when the user leaves the page and returns to it, you get the picture



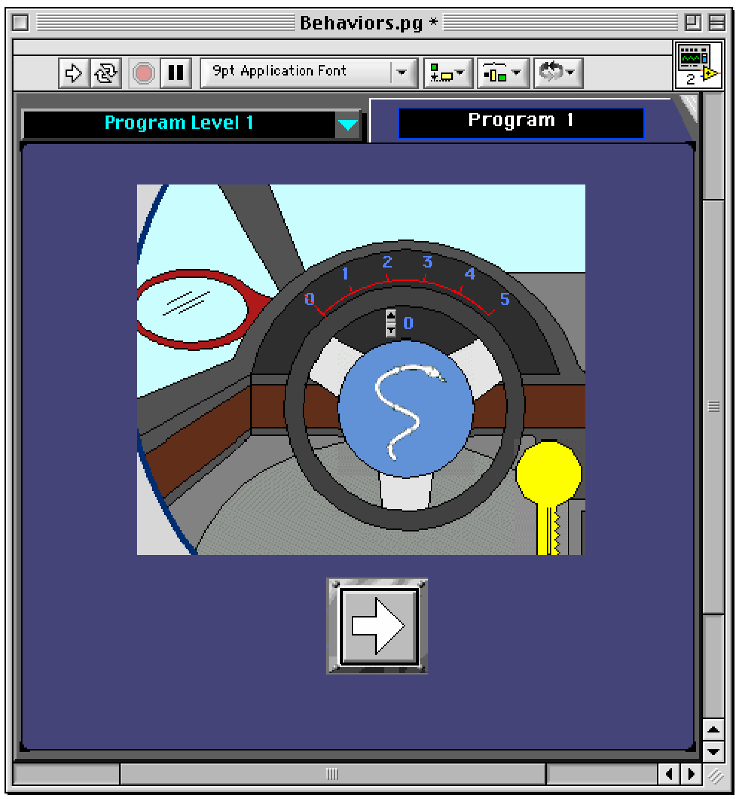
below and the user has to click on the button to get the

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

panel back. Therefore you might want to make your own template; described in the next section.

The Hard Way

To make templates that look and act just like the templates that came with ROBOLAB entails programming in LabVIEW. However it is possible to create templates (like the ones below) that are embedded in Investigator and can be completely customized to the needs of specific application or need. The template below allows for a customized programming interface to be added to the Program area.



Developers interested in creating customized templates should visit www.lego.com/eng/education/mindstorms or www.ceeo.tufts.edu/robolabatceeo to download the latest documentation and code needed for creating customized templates. At the website are instructions on how to make a title page in the journal area, a behavior-based programming page in the programming area, an upload page

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

that uploads microphone measurements, and a View area page that returns the natural frequency of a data set. To learn about programming in LabVIEW refer to either the LabVIEW Student Edition or LabVIEW for Everyone, or a host of other books on programming in LabVIEW (See www.ni.com for more information).

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

5. Help & Idea Sources

The ROBOLAB developers and the general ROBOLAB community are constantly coming up with new ways of learning how to use the software as well as cool new ideas for classroom, home, or after school projects. The best way to find the latest news, downloads, and links is to visit http://www.lego.com/eng/education/mindstorms (under Support) or http://www.ceeo.tufts.edu/robolabatceeo.

Included with Your Installation of ROBOLAB

- **Setup** describes the steps for preparing the hardware: the LEGO RCX programmable brick and IR Transmitter used for communication between the computer and the RCX.
- **Training Missions** provide step-by-step instructions so that you and your students get started with Pilot and Inventor styles of programming.
- **ROBOLAB Help** is available as you are running ROBOLAB. Help is activated by clicking Help in the menu or pressing CTRL-H on your keyboard. Help provides information about the object or icon on which your mouse is placed.

Software Guide - *Using ROBOLAB*

For a more detailed introduction to using the software, step by step directions, and more examples and challenges, the ***Using ROBOLAB*** guide is very useful. This book is available through your LEGO Education distributor or in electronic (pdf) format at the website. If you would like to find a distributor, visit http://www.lego.com/education (under Where to Buy).

Topics in the User's Manual Include

- The Pilot and Inventor levels of programming in the PROGRAMMER component of ROBOLAB. This is all one needs to program a LEGO robot.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

- The INVESTIGATOR component that is used for data logging. This section adds scientific investigation to the robot's capabilities, allowing the RCX to become a smart and mobile data logging tool. The Investigator component also includes a Journal Area for documenting a project in text and photographs or illustrations and a Publish feature for sharing results.

- The enhanced Media features: the Piano Player and Camera available in ROBOLAB 2.5.1. It also shows how to use some of the higher level capabilities including Vision Center and subroutines.

Additional Help Resources, Developer Support, & Add Ins

Visit http://www.ceeo.tufts.edu/robolabatceeo for the latest information and help on working with the ROBOLAB environment.

Technical Support

For help in troubleshooting problems, www.lego.com/eng/education/mindstorms (under Support) or contact your Lego Education distributor.

Activities

There are many activities and building sets created for LEGO MINDSTORMS FOR SCHOOLS. For information on these materials, go to www.lego.com/eng/education/mindstorms and then click on Classroom Solutions.

Activity ideas are also available online using the Invent & Investigate database in the ROBOLAB website. You can sign in, search and download student activity pages and teacher notes for over two dozen activities. Go to
go to www.lego.com/eng/education/mindstorms and then click on Classroom Solutions.

Partner Resources

ROBOLAB software was developed as a cooperative effort by LEGO Education, Tufts University and National Instruments.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

The main website for LEGO Education is www.lego.com/education.

Tufts University uses the RCX and ROBOLAB in its own engineering curricula. The University also extends its work to K-12 educators through the Center for Engineering Educational Outreach (CEEO). For curriculum, teacher workshops, student camps, and just for inspiration -- go to www.ceeo.tufts.edu.

ROBOLAB software is powered by the National Instruments LabVIEW programming environment. National Instruments created LabVIEW for industry engineering applications. The general National Instruments website is found at www.natinst.com. If you would like to learn more about LabVIEW programming, National Instruments has online and instructor-led courses available. For details, see www.ni.com/custed/.

Robotics Ideas

For inspiration on building different kinds of LEGO robots, see the LEGO MINDSTORMS website, www.lego.com/robotics.

For robotics enthusiasts and information on programming in other environments, try the online community www.lugnet.com. To get to the ROBOLAB area on this site, go to www.lugnet.com/robotics/rcx/robolab.
Note: This group is not affiliated with the LEGO Group.

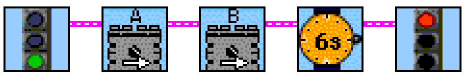
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Appendix

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

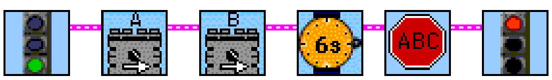
Troubleshooting Programming Problems

"My motors never stop running"



Sample Problem Code

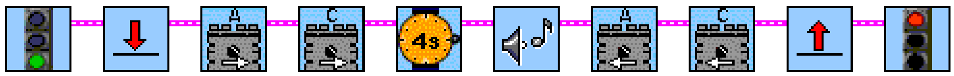
A common mistake in Inventor level programming is to write programs similar to the one above. The program above will turn motors A and B on, wait for 6 seconds, and then the program will stop running. However, the motors will continue to run and run. Inventor level programming requires that you explicitly stop the motors as in the program below.



Solution

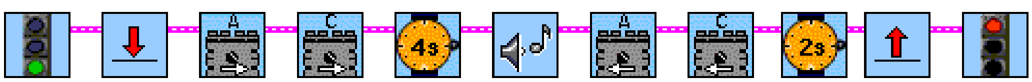
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

"I programmed my car to do X but it seems like it skips over that part"



Sample Problem Code

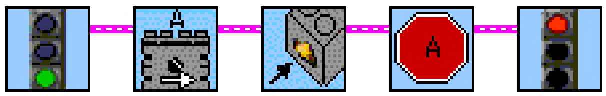
A simple programming challenge would be to program a car to drive forward for 4 seconds, beep, and then back-up. The above program is an attempt at solving that challenge (assuming you have a car with motors on ports A & C). However, there is a problem – the car will never back-up. Why? Immediately after the motors are set to run in reverse the program loops to the beginning of the program and sets the motors to run forward. The process happens so quickly (within 100ths of a second that the car will never back-up (you may hear the motors click). The solution is to add a Wait For Statement after you have set the motors in reverse.



Solution

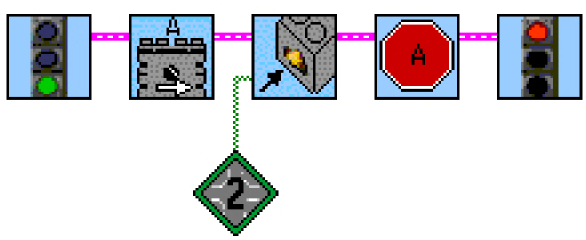
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

"I programmed my motors to stop running after the touch sensor is pressed but the motors keep running even after I press the touch sensor"



Sample Problem Code

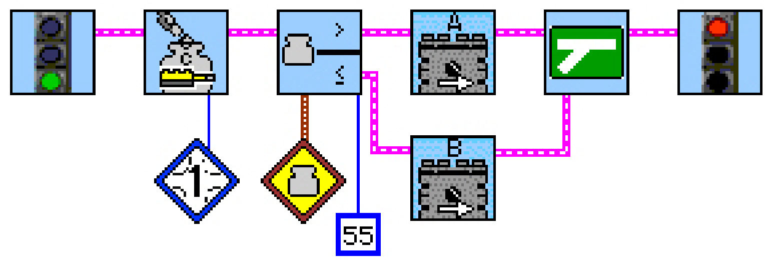
Often times programming problems are the result of how you have built your robotic creation. In the above program there is no modifier wired into the Wait For Touch icon, therefore it defaults to waiting for a touch sensor to be pressed on Port 1. However, if in your creation you have the touch sensor plugged into Port 2 - no matter how much you press it your motors will never be triggered to stop running. The programming solution to this problem is to specify the port which your touch sensor is plugged into (if your model allowed you could also physically switch which port your touch sensor was plugged into).



Solution

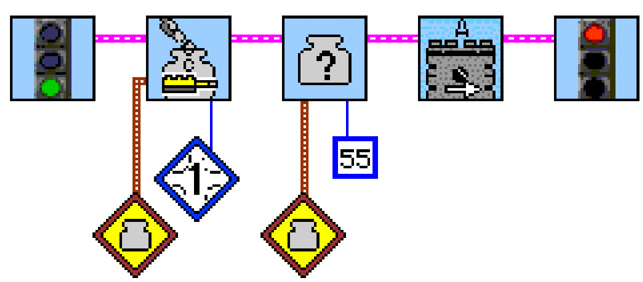
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

"Motor A should turn on when the yellow container (being filled by the temperature sensor) has a value of 55 or greater, but even when the RCX says the temperature sensor is 60, nothing happens."



Sample Problem Code

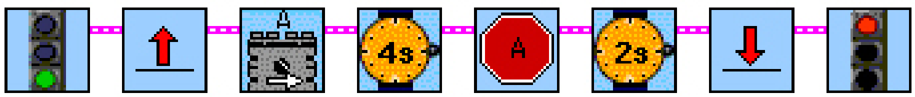
This program does not work because the temperature sensor data is being put into the red container, but the program is waiting for the Yellow Container to equal 55 or greater. Container commands default to the Red Container if no other color container is denoted. Changing the temperature container to yellow or using the Red Container for the Wait for Container will allow the program to run.



Solution

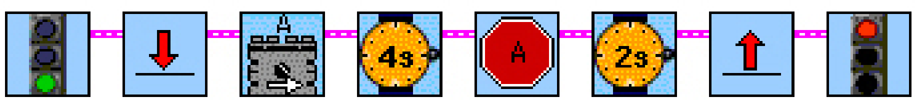
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

"I programmed my car to drive for 4 seconds, stop for 2 seconds, and then repeat indefinitely, but nothing happens."



Sample Problem Code

The jump and land icons are in the wrong positions. The land icon should be first. The first thing that happens in this program is that the Jump icon tells the program to go directly to the Land icon. This means that the rest of the program is bypassed and thus, the car never moves.

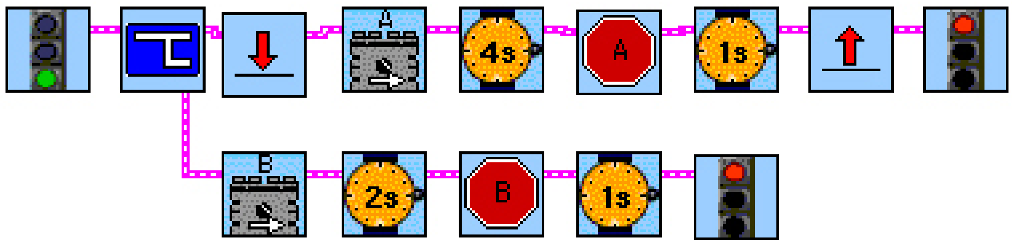


Solution

"

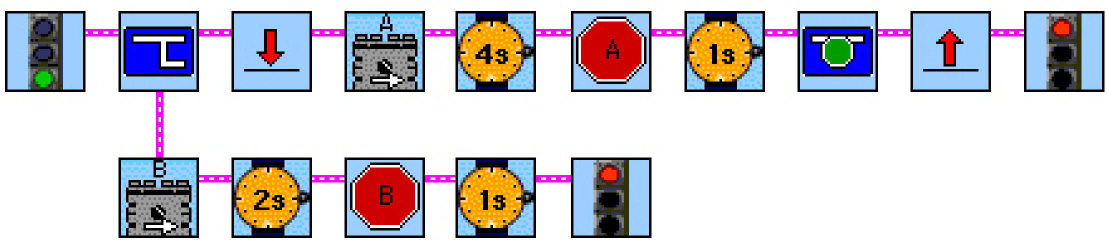
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

My program should run two motors, but Motor B never turns on."



Sample Problem Code

Motor B is programmed to run as a Subroutine. It will not turn on until you tell it to. Insert the Subroutine icon into the top string when you want the motor to turn on.



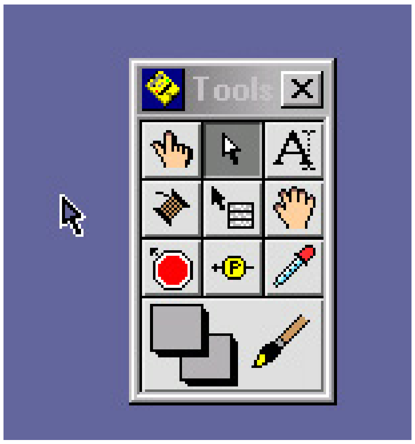
Solution

LEGO educational division

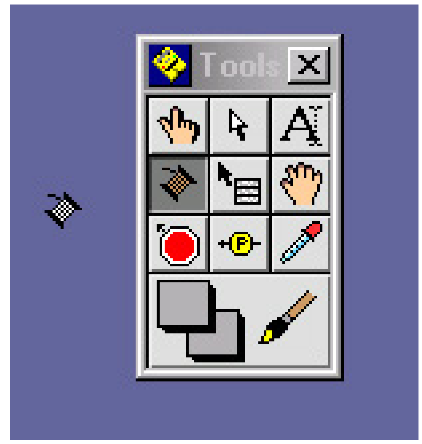
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use. All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Tips and Tricks

Using the Space Bar to Switch between Tools

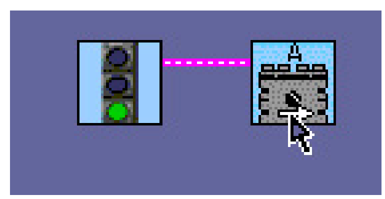


To quickly change between the Position/Size/Select tool and the Connect Wire tool press the Space Bar on the keyboard. You can even do this when the Tools Palette is not visible.



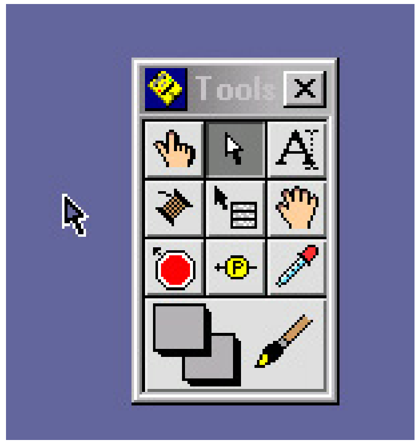
Auto Wiring and the Space Bar

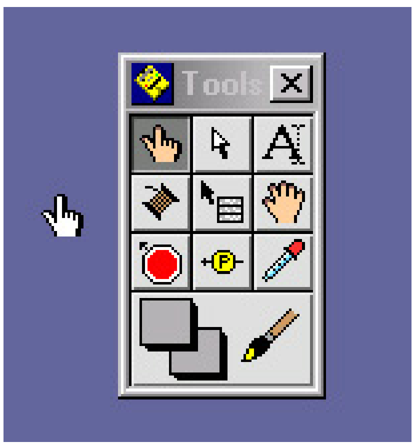
Once you have placed an item in your program you can reactivate auto wiring by selecting the item and pressing the space bar



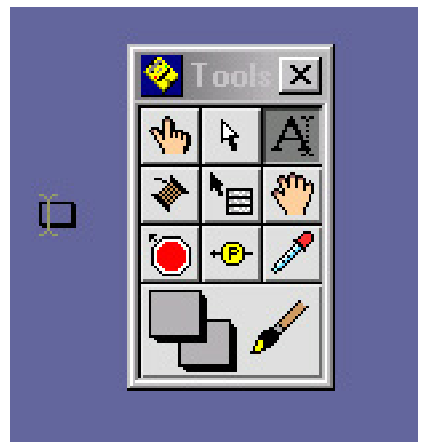
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

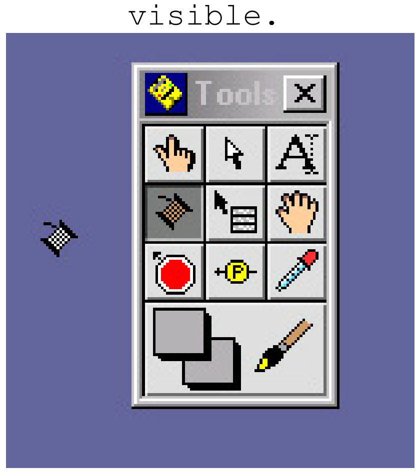
Using the Tab Key to Switch between Tools





Use the Tab Key to switch between the four most commonly used tools – Operate Value, Position/Size/Select, Edit Text, and Connect Wire. This can also be done while the Tools Palette is not visible.

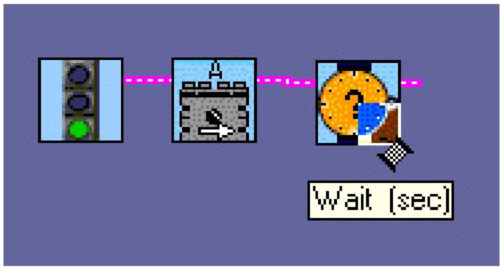


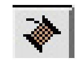


©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

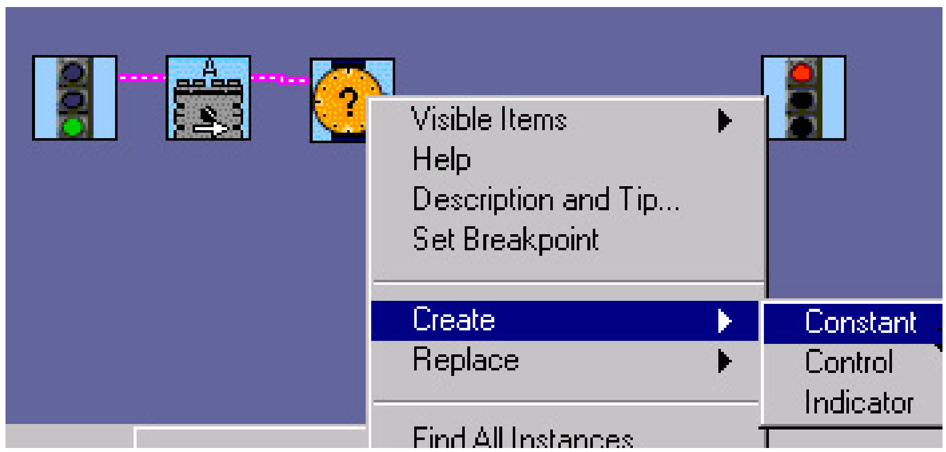
Creating a Constant

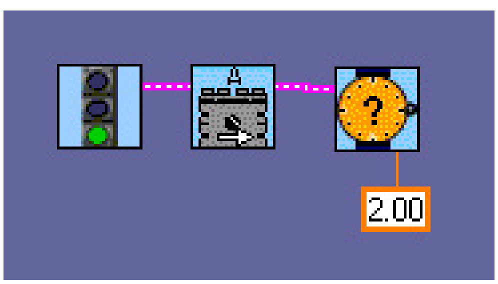
Rather than selecting a modifier from Functions Palette you can create modifiers for certain commands.

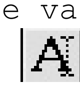
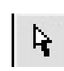


Using the Connect Wire tool , Right Click (or Apple Click) on the connection you want to create a constant for

From the menu that appears select 'Create' and the 'Constant'





An appropriate constant will be created. Immediately after it is create you can type in the value you wish (or modify the value later with the Edit Text tool ). Reposition the constant with the Position/Size/Select Tool .

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Related Web Sites and Resources

Aside from the official ROBOLAB (www.lego.com/education/robolab) and LEGO resources (www.lego.com/robotics) there are lots of great sites create by ambitious and talented RCX users.

ROBOLAB @ CEEO
(www.ceeo.tufts.edu/robolabatceeo)
This is Tufts University's Center For Engineering Educational Outreach's site for all things ROBOLAB including patches, beta features, activities, and other resources

Boulette's Robotics in Luxemburg
(www.convict.lu/Jeunes/RoboticsIntro.htm)
A fabulous site that features high end robotics and ROBOLAB projects with great descriptions and the code used. Projects address top

LUGNET (Lego Users Group Network)
(www.lugnet.com)
LUGNET has fabulous LEGO resources ranging from pieces in sets to a wide array of discussion groups on robotics, education, ROBOLAB and more

Mindstorms RCX Sensor Input Page
(www.plazaearth.com/usr/gasperi/lego.htm)
Michael Gaperi, one of the authors of ***Extreme Mindstorms: An Advanced Guide to Lego Mindstorms***, has a great web site on sensors and the RCX.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Acknowledgements

The ROBOLAB Reference Guide was written, compiled, and made possible by Martha Cyr, Scott McNamara, Elissa Milto, Merredith Portsmore, and Chris Rogers.

ROBOLAB for LabVIEW was developed by Tufts University with the support of the National Science Foundation and the cooperation of Lego Education and National Instruments

ROBOLAB for LabVIEW adds the ability to interface with numerous LEGO products -- including the RCX, LEGO Camera, and Control Lab Interface -- to LabVIEW Student Edition. It also provides an interaction tutorial that provides instruction on learning basic LabVIEW programming as well as data acquisition and analysis techniques.

For more information on using ROBOLAB for Labview, supporting curriculum, and additional materials visit: http://wwww.ceeo.tufts.edu/robolabatceeo/college.

ROBOLAB for LabVIEW is based upon work supported by the National Science Foundation under Grant No. 9950741. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation (NSF).

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

Index

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

educational division

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and
ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

LEGO educational division

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

LEGO educational division
©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.

educational division

©2002 LEGO, LEGO Educational Division, LEGO Mindstorms for Schools, and ROBOLAB are trademarks of the LEGO Group. This page may be photocopied for non-commercial educational use.
All other rights reserved.
Visit www.lego.com/eng/education/mindstorms & www.ceeo.tufts.edu/robolabatceeo.