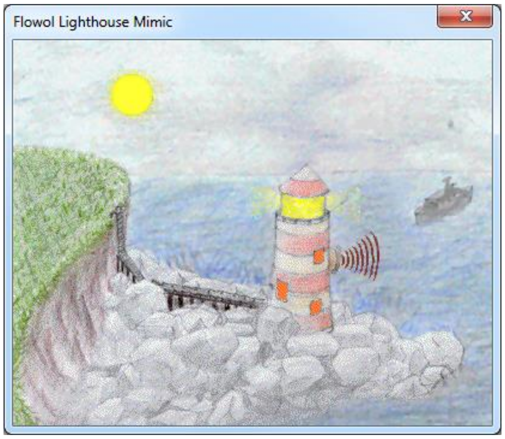
Chapter
6

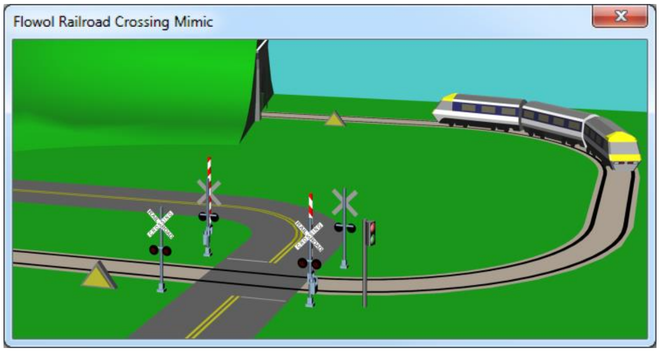
Chapter 6: Mimic Activities

Automatic systems are all around us, keeping us safe, making life comfortable and helping us with difficult and unpleasant tasks.

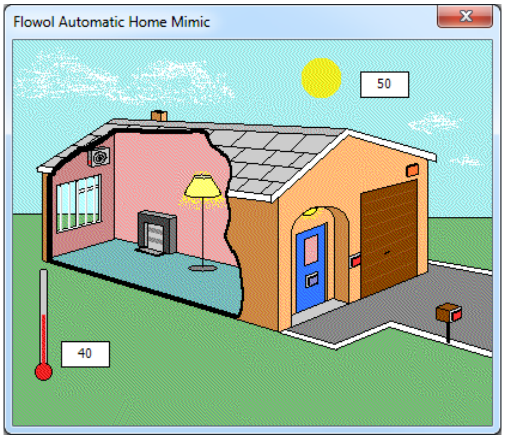


Flowol will allow you to produce your own solutions to many of these situations. We will start by guiding you through the simple tasks to control traffic signals and warning lights.

With your skills, you will soon learn to be able to solve more complex examples such as an automatic railroad crossing/level crossing or the control systems needed to help people in their homes.



What other situations can you think of where automatic control might help?



Use the grey boxes for teacher marks, grades or notes.

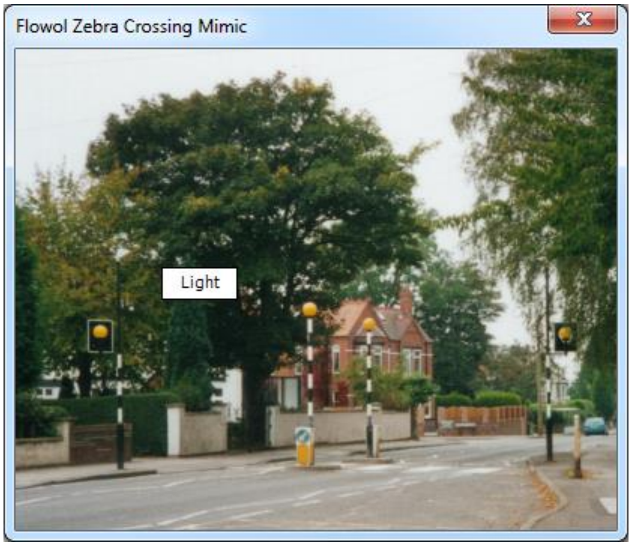
Print copies of these student activities for use in the classroom

NAME ____________________
CLASS ____________________
DATE ____________________

Zebra Crossing

See p. 13 in the tutorial for more help.

Open the Zebra Crossing mimic.



Where is there a crossing like this near to your school?

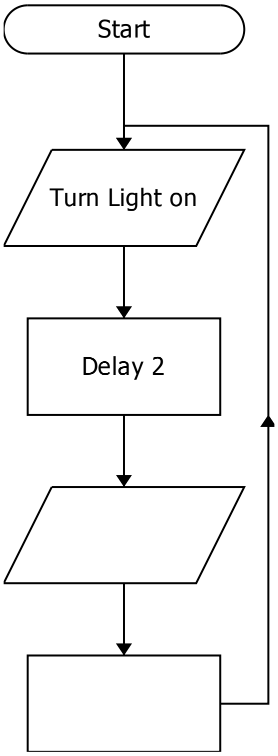
What is special about the Belisha beacons (amber lights) at the crossing? Why are they there?

What must drivers do when they get to this type of pedestrian crossing?

The pictures in Flowol are called Mimics and you can control them. To see what the mimic can do, click on the light in the picture.

Activity 1

Create the instructions (a program) to control the light by building this flowchart.



Click and drag each symbol from the left toolbar and place it on the workspace. Use the prompt box at the bottom of the screen to put the instructions in each symbol. Finally use the line tool to join up the symbols.

Remember to add your own instructions to the blank symbols.

Click on Run ▶ to see if your flowchart works.

The example uses a delay of 2 seconds. Is this a good time? Why would the Belisha beacon be less effective if the delay was too short or too long?

What improvements would you like to see made at pedestrian crossings?

What other types of road crossings already exist?

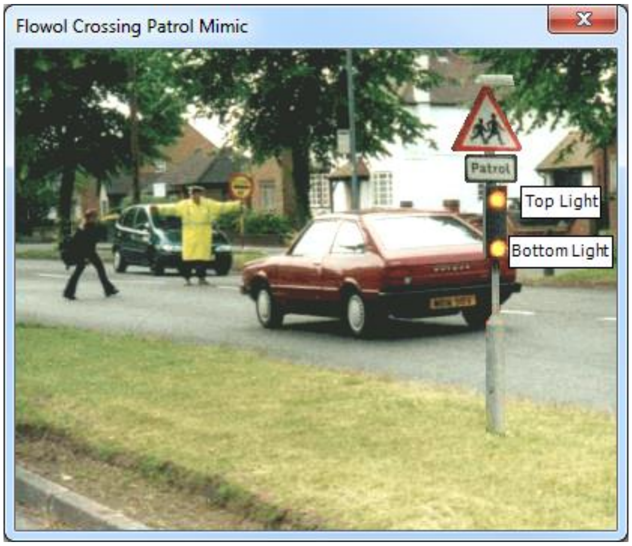
Remember to save your flowchart!

NAME ____________________
CLASS ____________________
DATE ____________________

Crossing Patrol

See p. 16 in the tutorial for more help.

Open the Crossing Patrol mimic.



Where is there a crossing patrol like this near your school?

There are two lights on the signpost.

What are these lights for?

Do the lights come on together, or alternately?

Create a flowchart to control these two lights.

Remember to save your flowchart!

Click on Run to check your flowchart and then make any improvements.

How can crossing a road be made even safer for school children?

Use the Label Tool ***T*** to add a title to your flowchart.

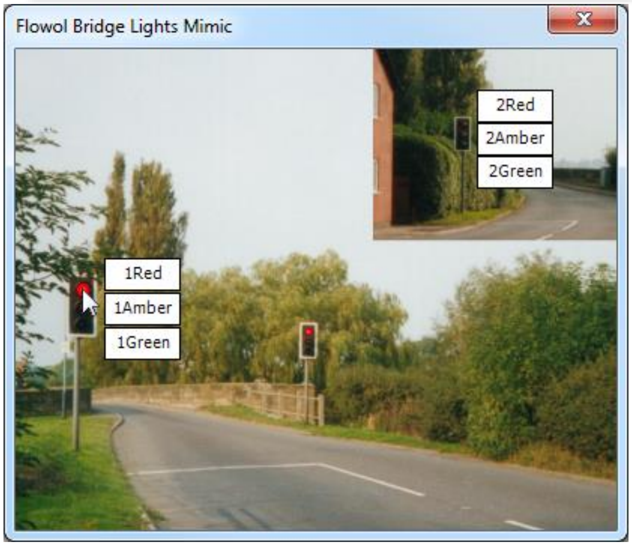
NAME ____________________
CLASS ____________________
DATE ____________________

Bridge Lights

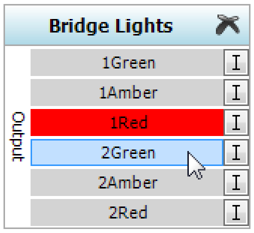
See p. 20 in the tutorial for more help.

Is there a narrow bridge or road near your school that requires traffic lights in order to avoid a collision?

You've seen a single traffic light sequence many times before. But what is the combined sequence when both sets work together?



Explore how the mimic looks when the outputs are turned on by clicking on the outputs in the Status Panel.



Bridge Lights

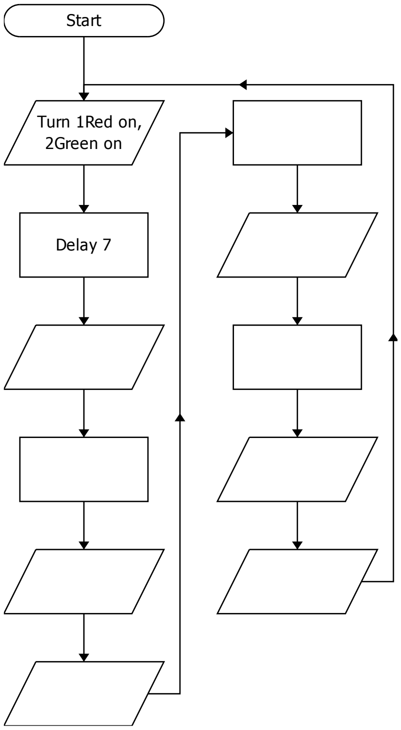
Output
- 1Green
- 1Amber
- 1Red
- 2Green
- 2Amber
- 2Red

Activity 1

First, create a flowchart to control a single set of traffic lights.

Activity 2

Now, modify your flowchart to control both sets of lights together. The flowchart might look like the one below.



Remember to save your flowchart!

Remember to fill in the empty symbols.

Click on Run ▶ to see if your flowchart works. And make any refinements or modifications if necessary.

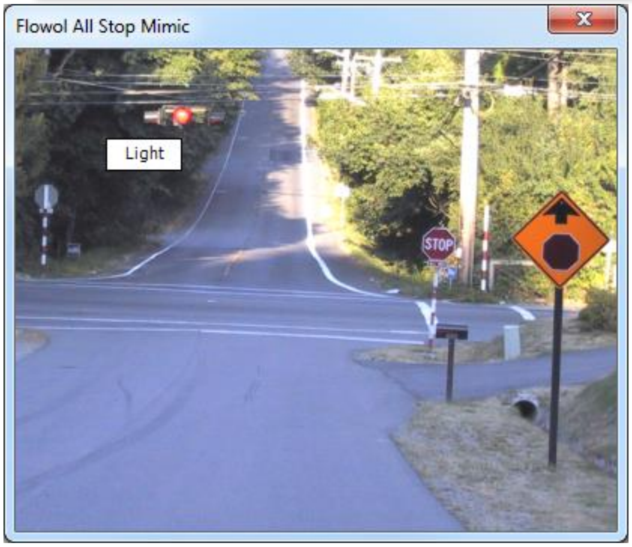
Tooltips

When an output symbol is controlling 3 or more outputs, the text may be truncated if it does not fit in the symbol. When this is the case, and you move the mouse pointer over the symbol, a tooltip will appear showing the full text.

NAME ____________________
CLASS ____________________
DATE ____________________

All-way Stop

See p. 17 in the tutorial for more help.



Open the All Stop mimic. It shows a fairly quiet intersection.

Where is there a street sign like this in your neighborhood?

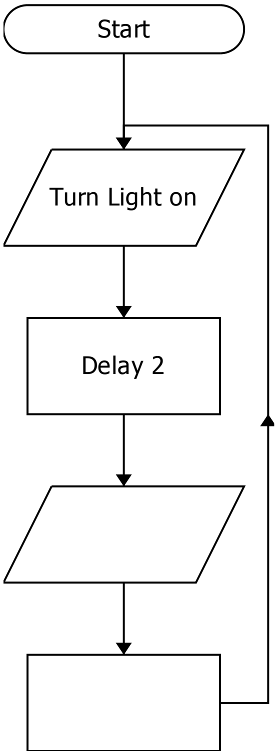
What is special about the light hanging over the intersection? Why is it there?

What must drivers do when they get to this type of intersection?

The pictures in Flowol are called Mimics and you can control them. To see what the mimic can do, click on the light in the picture.

Activity 1

Create the instructions (a program) to control the light by building this flowchart.



Click and drag each symbol from the left toolbar and place it on the workspace. Use the prompt box at the bottom of the screen to put the instructions in each symbol. Finally use the line tool to join up the symbols.

Remember to add your own instructions to the blank symbols.

Click on Run ▶ to see if your flowchart works.

The example uses a delay of 2 seconds. Is this a good time? Why would the stop light be less effective if the delay was too short or too long?

What improvements would you like to see made at road intersections?

What other types of stop light are there?

Remember to save your flowchart!

NAME ____________________
CLASS ____________________
DATE ____________________

Crosswalk

See p. 19 in the tutorial for more help.

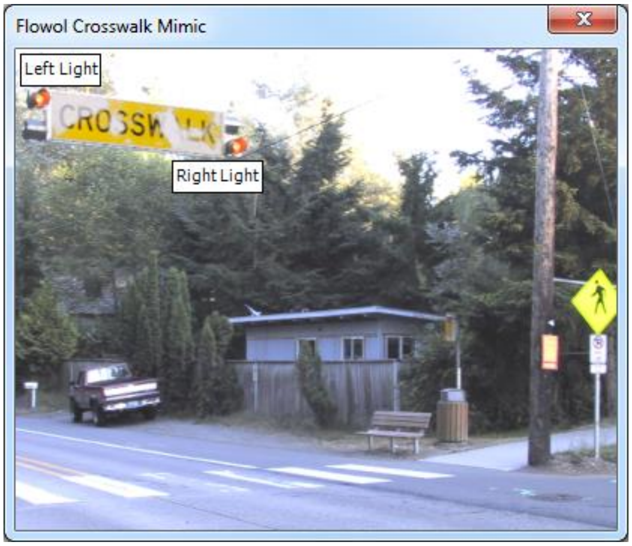
Open the Crosswalk mimic.

Where is there a crossing like this in your neighborhood?

A crosswalk has two lights suspended above it.

What are these lights for?

Do the lights come on together, or alternately?



Create a flowchart to control these two lights.

Remember to save your flowchart!

Click on Run to check your flowchart and then make any improvements.

How can crossing a road be made even safer?

Use the Label Tool ***T*** to add a title to your flowchart.

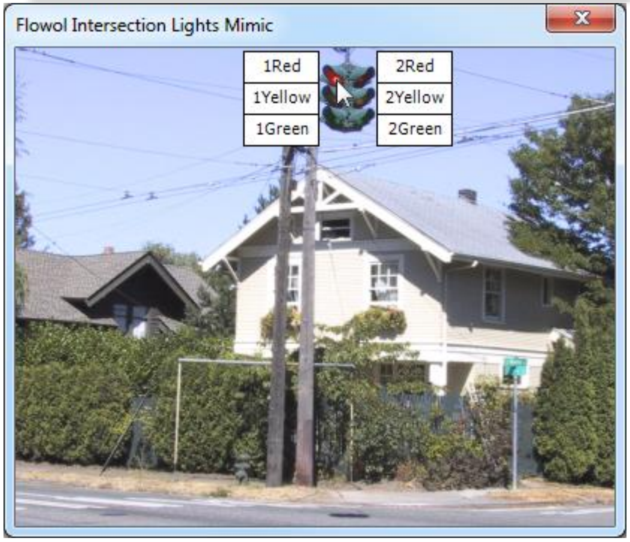
NAME ______________________
CLASS ______________________
DATE ______________________

Intersection Lights

See p. 20 in the tutorial for more help.

Where are the traffic signals in your neighborhood? They may hang from cables in the middle of the street like these, or be fixed to metal supports which reach over the road.

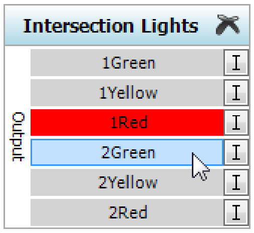
You've seen a single traffic light sequence many times before. But what is the combined sequence when both sets work together?



Explore how the mimic looks when the outputs are turned on by clicking on the outputs in the Status Panel.

Activity 1

First, create a flowchart to control a single set of traffic lights.

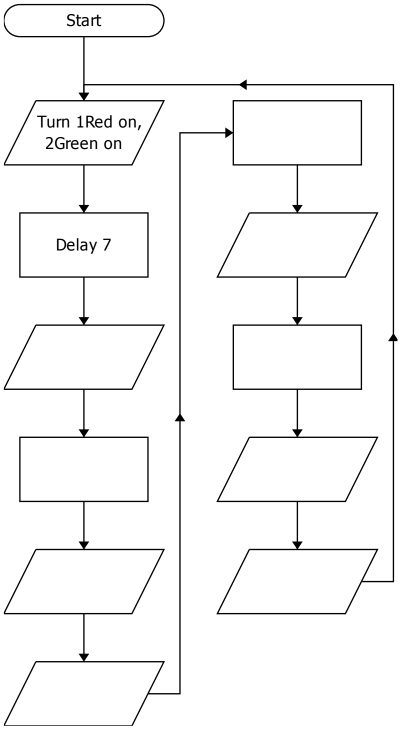


Intersection Lights

Output
- 1Green
- 1Yellow
- 1Red
- 2Green
- 2Yellow
- 2Red

Activity 2

Now, modify your flowchart to control both sets of lights together. The flowchart might look like the one below.



Remember to save your flowchart!

Remember to fill in the empty symbols.

Click on Run ▶ to see if your flowchart works. And make any refinements or modifications if necessary.

Tooltips

When an output symbol is controlling 3 or more outputs, the text may be truncated if it does not fit in the symbol. When this is the case, and you move the mouse pointer over the symbol, a tooltip will appear showing the full text.

NAME ____________
CLASS ____________
DATE ____________

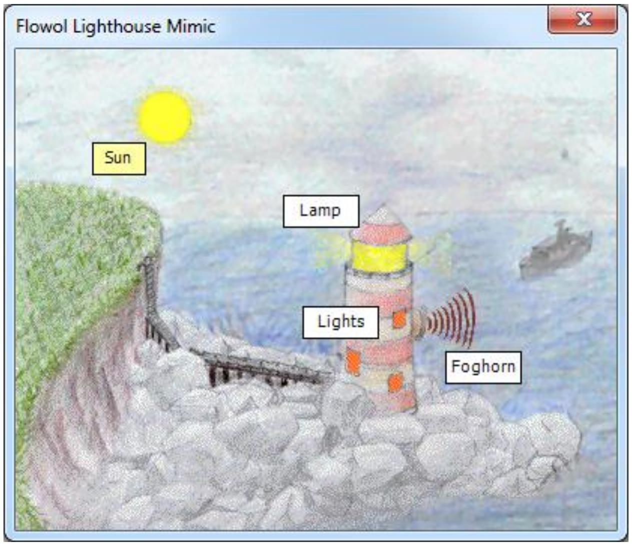
Lighthouse

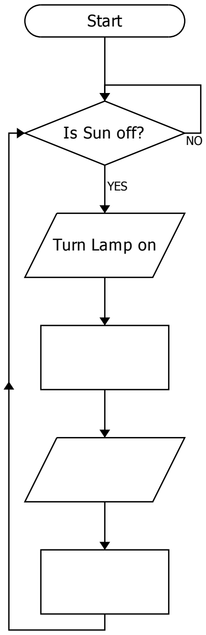
See p. 23 in the tutorial for more help.

So far the systems have been controlled by a set of instructions which are remembered and repeated. In the next mimics, the scenarios may need to respond to an external event such as a button being pressed or the daylight (brightness) changing.

Open the **Lighthouse** mimic and explore by clicking on the three outputs: **Lamp**, **Lights** and **Foghorn**, and the input **Sun**.

The Sun input is representing a digital light sensor which is on when it is daylight. Click on the sun/moon to toggle it.





Activity 1

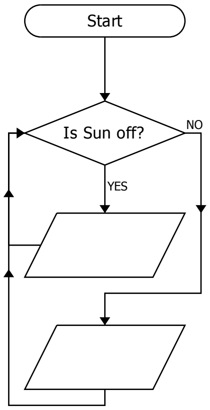
Construct this control flowchart to turn on the flashing Lamp of the lighthouse at nighttime. Add some labels to your flowchart.

Note that you always need both a **YES** and an **NO** line from a decision symbol.

Activity 2

Now create another flowchart on the same workspace to control the inside Lights. The inside lights should stay on when it is dark and go off automatically in the daytime.

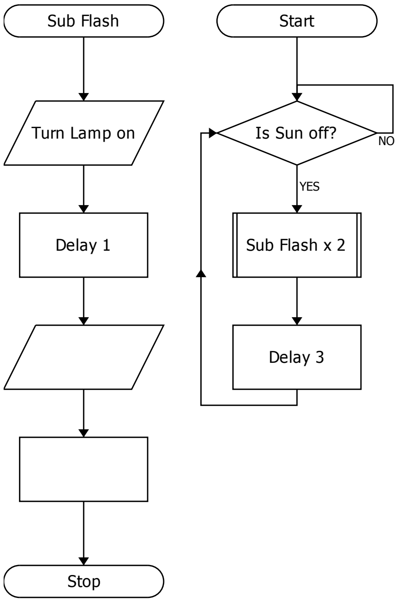
Since both flowcharts have a Start, they will both run in parallel when you click ▶.



Remember to save your flowchart!

Activity 3

Create a more interesting flashing sequence with a subroutine.



A subroutine must first be defined with the ⬭ symbol. Once the subroutine has been defined, the **Call Sub** symbol Sub will appear on the left toolbar. Use it in the main flowchart to call (invoke) the subroutine. In the example to the right, the **Flash** subroutine is run twice (**x 2**).

Now adjust the main flowchart to create your own interesting flashing sequence.

Activity 4

Control the foghorn by constructing another flowchart.

Activity 5

Add an "Mmmmmm" sound to your foghorn. If you have a microphone and suitable recording software, record your own sound.

Remember to save your flowchart!

See page 26 in the tutorial for help playing a sound.

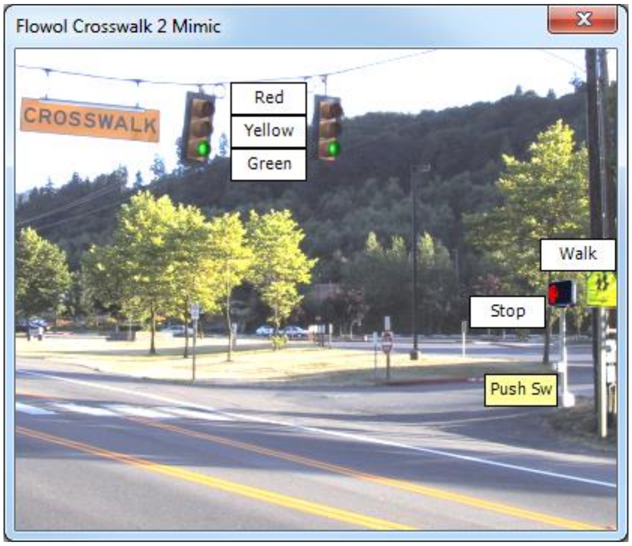
NAME ____________________
CLASS ____________________
DATE ____________________

Crosswalk with Stop Light

Open the **Crosswalk 2** mimic.

Click on the **Push Sw** Input (the small white circle) on the mimic window. This is a push switch with goes off after ½ a second.

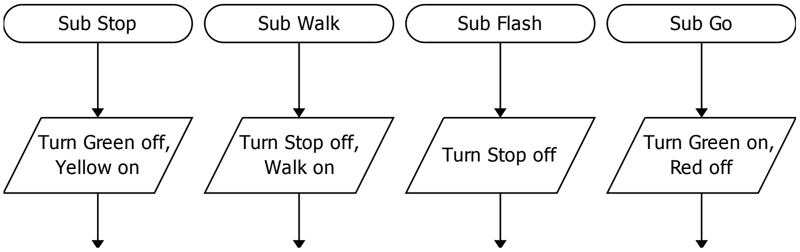
Also explore the Outputs to see what the mimic can do.

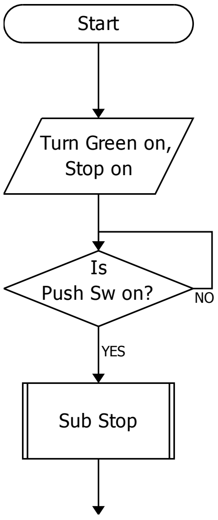


Flowol Crosswalk 2 Mimic

Discuss with a friend to see if you both know how the lights and symbols change when the button is pressed.

Split your solution into four separate statements.





Activity 1

Construct and complete these four subroutines:

- To stop the vehicles.
- To indicate when it is safe to walk.
- To warn the pedestrians to clear the crossing.
- To allow the vehicles to move.

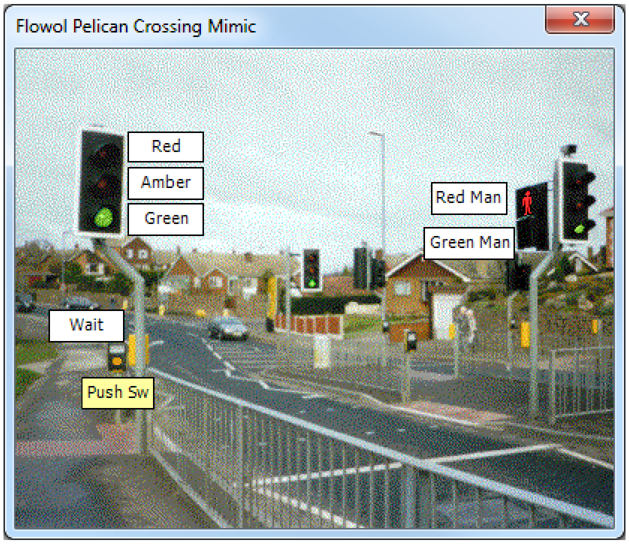
Activity 2

Complete the main flowchart to call (invoke) the subroutines correctly.

Remember to save your flowchart!

NAME ____________________
CLASS ____________________
DATE ____________________

Pelican Crossing



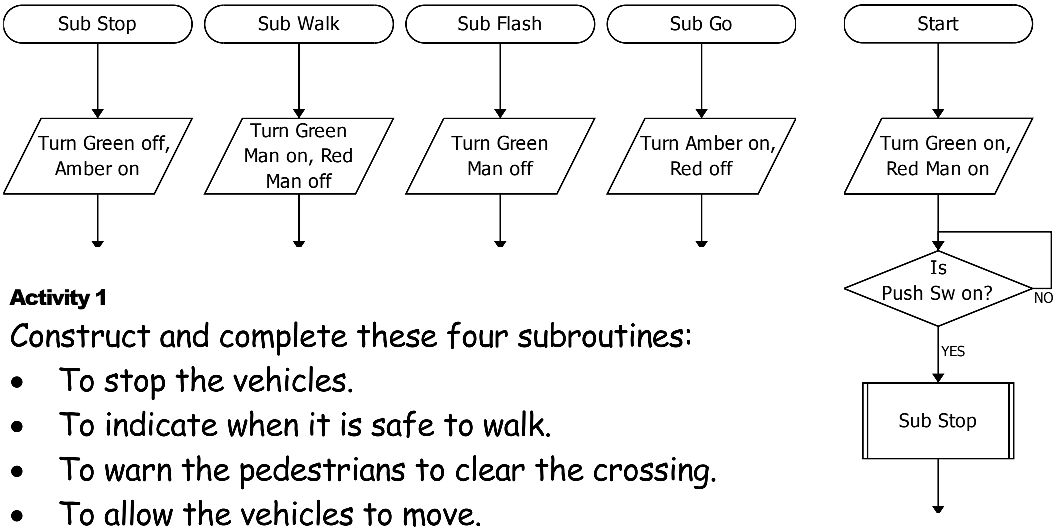
Open the **Pelican Crossing** mimic.

Click on the **Push Sw** Input (the orange circle) on the mimic window. This is a push switch with goes off after ½ a second.

Also explore the Outputs to see what the mimic can do.

Discuss with a friend to see if you both know how the lights and symbols change when the button is pressed.

Split your solution into four separate statements.



Activity 1

Construct and complete these four subroutines:

- To stop the vehicles.
- To indicate when it is safe to walk.
- To warn the pedestrians to clear the crossing.
- To allow the vehicles to move.

Activity 2

Complete the main flowchart to call (invoke) the subroutines correctly.

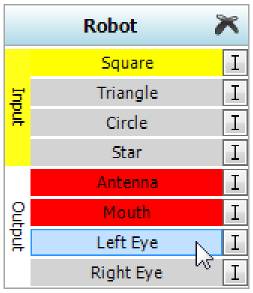
Remember to save your flowchart!

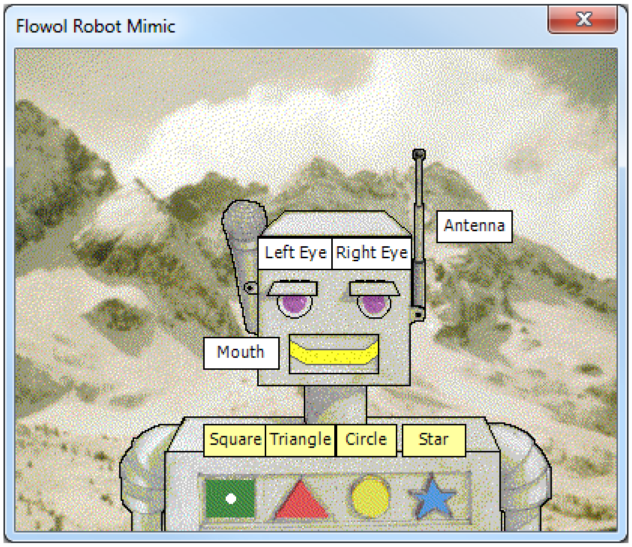
NAME ________________
CLASS ________________
DATE ________________

Robot

Let's have a little fun with the **Robot** mimic.

It has 4 inputs along its chest, and 4 outputs. Explore the mimic by clicking on the Status Panel:





Activity 1

Amuse your friends by constructing three or four separate flowcharts to control different movements of the robot.

Remember to save your flowchart!

Activity 2

Make your robot talk using the Speech feature.

See page 27 in the tutorial for help using speech.

Activity 3

Construct a program to animate the robots mouth while the computer is speaking so it appears as if the robot is speaking.

Activity 4 (Larger project)

Imagine that the robot is a toy for a young child who is just learning their colors and shapes.

Create a program for the robot toy which uses Speech to ask the child to press a certain shape or colored button. Make the robot tell them if they got it correct or whether they need to try again.

Remember to break the program into subroutines.

When the child gets the correct button make the robot react in an excited way opening and closing its eyes etc.

See page 37 in the tutorial to get a larger flowchart workspace.

Activity 5 (Advanced)

Use the Random feature of Flowol to extend the solution above so that the robot randomly asks the child to find different buttons. Ask by color or shape.

See page 75 in the tutorial for help using Random numbers.

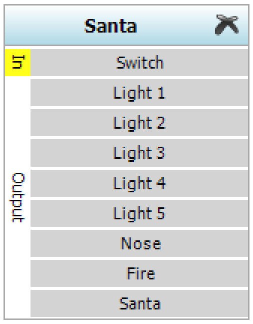
Remember to save your flowchart!

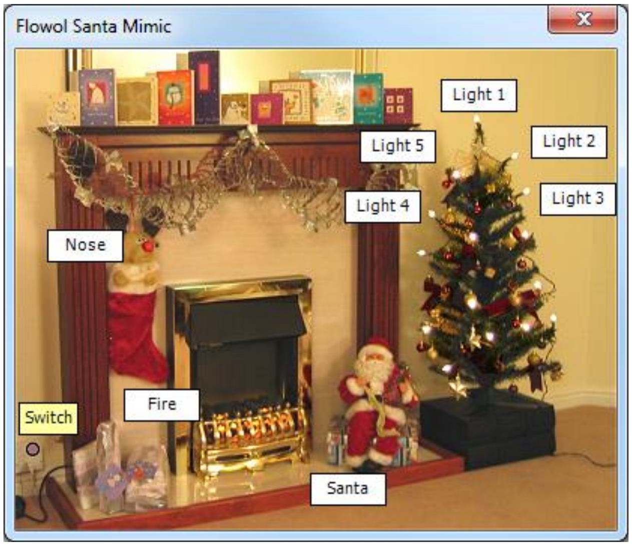
NAME ____________________
CLASS ____________________
DATE ____________________

Santa

Season's Greetings!

Open the **Santa** mimic. Use the Status Panel to see what the mimic can do.





Activity 1

Using the **Switch**, construct a flowchart to turn the electric **Fire** on and off.

Activity 2

Construct another routine to make Rudolph's **Nose** flash when the **Switch** is on.

Activity 3

The Christmas tree lights could be controlled in several ways. You could make them all come on together when the **Switch** is on. You could also make them all flash together, or they could twinkle if you have different groups coming on and off at different times. Create your own program to control the lights.

Activity 4

Santa's movement can be controlled with the **Santa** digital output. Write a subroutine to move Santa once. Then create a main program to call (invoke) the subroutine and make him dance.

Activity 5

Use the Sound feature to play a recorded Christmas greeting and make Santa sing it to you.

See page 26 in the tutorial for help using sound.

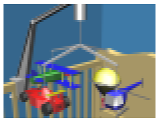
Activity 6 (Advanced)

Use the Random function to have the Christmas tree lights flash in a subtle, but random fashion.

See page 75 in the tutorial for help using Random numbers.

Remember to save your flowchart!

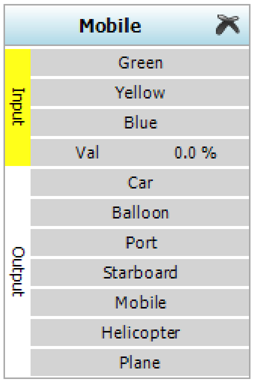
NAME ________________
CLASS ________________
DATE ________________



Mobile

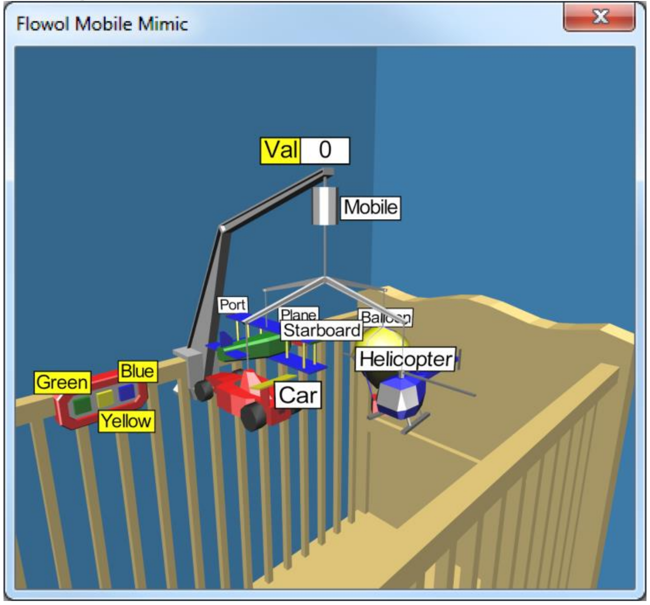
See p. 29 in the tutorial for more help.

A clockwork crib mobile can comfort and entertain a baby, but how could an automatic crib mobile be an advantage?



Mobile	
Input	Green
	Yellow
	Blue
	Val 0.0 %
Output	Car
	Balloon
	Port
	Starboard
	Mobile
	Helicopter
	Plane

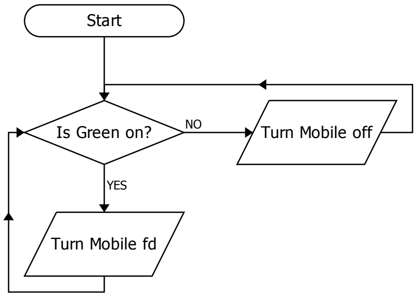
Open the **Mobile** mimic, and show its labels. Explore what it can do by clicking on the outputs and motors in the Status Panel.



Click with the right mouse button on the motors to reverse them.

Activity 1

Controlling the motor to rotate the whole mobile is the most impressive effect. Build this program to use the **Green** digital input to start and stop the **Mobile** motor.



Activity 2

Motors can of course go forwards and reverse. Construct another two flowcharts to give some forward and reverse movement to the **Helicopter** rotors and the **Plane** propeller when the **Yellow** and **Blue** inputs are used.

Run the whole program. This should give some interesting combination movements when the three switches are changed.

Activity 3

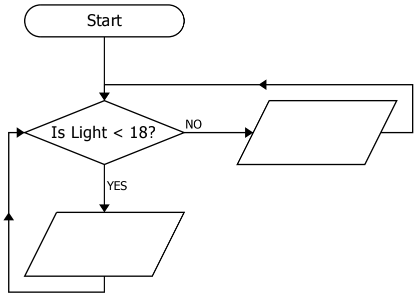
Another important control feature for a motor is to change its speed (or power). Modify your first flowchart to reduce the main rotation speed of the mobile for a while and then speed it up again. [Remember, if you reduce the motor speed percentage, then it must be returned back to 100% for full power].

Remember to save your flowchart!

See page 34 in the tutorial for help using variables to control the speed of the motors.

So far we have used digital inputs which can only be either on or off. Another type of input can be from an analog sensor which detects a range of input values (e.g. analog values could be from different levels of light brightness, different temperatures or different volumes of sound).

The **Mobile** mimic has an analog sensor at the top of the mobile support arm which changes value when you click on the number with the left and right mouse buttons.



Activity 4

Rename the analog value from **Val** to **Light** and treat it as a light level sensor. Construct an automatic light flowchart to turn on the light in the hot air **Balloon** if the light level goes below a value of 18 units (i.e. Light < 18)

Activity 5

The baby should still be occupied before it gets quite this dark, so produce another one or two flowcharts to make the rear light on the **Car** and the **Port** and **Starboard** plane wing tip lights flash if the light level value goes below 60 units.

Activity 6

The cot mobile would be most interactive if it could respond to the baby's sound. So rename the analog sensor to **Sound**. Larger values represent a louder sound.

Rebuild the flowchart to respond to the sound that the baby is making. If the baby makes a quiet noise, some of the lights could turn on or flash for a while. If the noise gets louder, the rotors and propeller could start to move slowly, and if the baby gets very noisy, the whole mobile could become very active.

As the baby settles down, and makes less noise, the mobile should also slow its activity to sooth the baby back to sleep.

Remember to save your flowchart!

A gentle musical sound could also be added to the mobile.

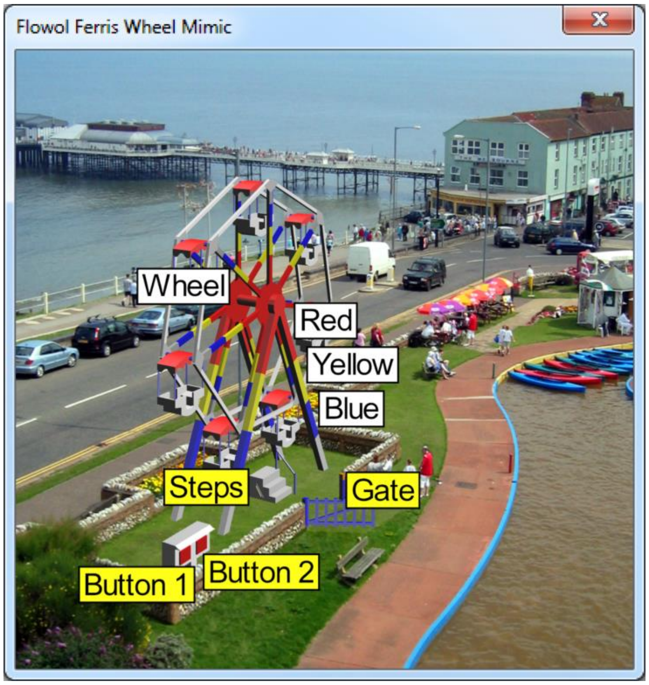
Finally, use a word processor or the label feature in Flowol to write some brief instructions for the parents.

NAME ______________________
CLASS ______________________
DATE ______________________

Ferris Wheel/Big Wheel

See p. 32 in the tutorial for more help.

If you were the operator of a funfair ride, how would you produce an exciting but safe experience?



Open the **Ferris Wheel/Big Wheel** mimic. Explore what the mimic can do by clicking on the inputs, outputs and motors on the Status Panel.

Button 1 and **Button 2** are normal inputs but, when the wheel is rotating, you may notice that the **Steps** input flashes on each time a seat passes over the steps. Also, if you click on the **Gate** with the left mouse button you will find that the **Gate** input comes on when the gate is shut.

The Steps and Gate inputs are called virtual inputs. They cannot be changed by directly clicking on the mimic, but are changed by features within the mimic itself.

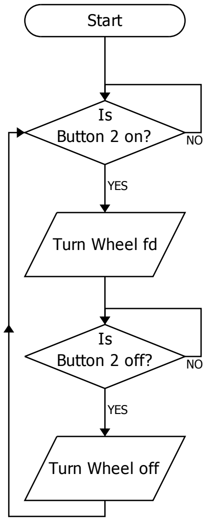
Activity 1

To attract the crowd, use the **Button 1** input to control the lighting effects on the wheel's frame. This might be a simple on/off routine but flashing sequences are more exciting. Use subroutines.

Activity 2

Use the **Button 2** input to control the simple Go/Stop movement of the wheel. You could perhaps make the wheel speedup and slowdown in stages by changing the motor power.

Remember to save your flowchart!



□

Activity 3

Modify activity 2 to include the safety feature of the gate so that the gate must be closed before the wheel will start (i.e. both **Button 2** and **Gate** are on). The wheel should stop if either **Button 2** is turned off, or the **Gate** is opened (i.e. if either **Button 2** or **Gate** are off).

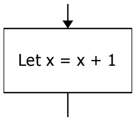
□

Activity 4

To increase the safety even further, use the Speak command to give the passengers an automatic verbal instruction to "Hold tight please" just before the ride begins to move.

See p. 27 in the tutorial for help using speech.

□



Activity 5: Counting how many times the ride is used

Add this symbol to the flowchart created above in Activity 2. This uses variable *x* to count how many times the ride is used.

□

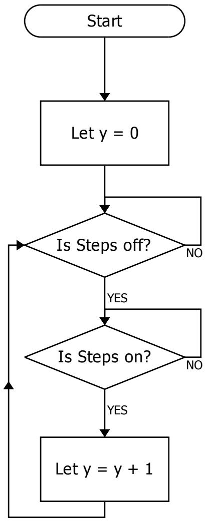
Activity 6: Stopping the Wheel automatically

Construct this counting program to increase the variable *y* each time a seat passes the steps, i.e. each time the virtual input goes off and on.

Since there are 7 seats, each rotation of the wheel should increase the variable *y* by 7.

Now modify your program by introducing a decision symbol, to stop the wheel automatically after it has rotated 3 times.

See page 33 in the tutorial for help using variables.



□

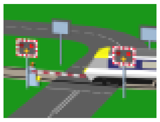
Activity 7

Now that you have learned how to use the **Steps** input, create a subroutine which rotates the wheel and stops briefly at each of the seven seats for passengers to get on or off.

Call (invoke) this subroutine twice; once at the beginning to load the wheel with passengers, and then at the end to unload.

Remember to save your flowchart!

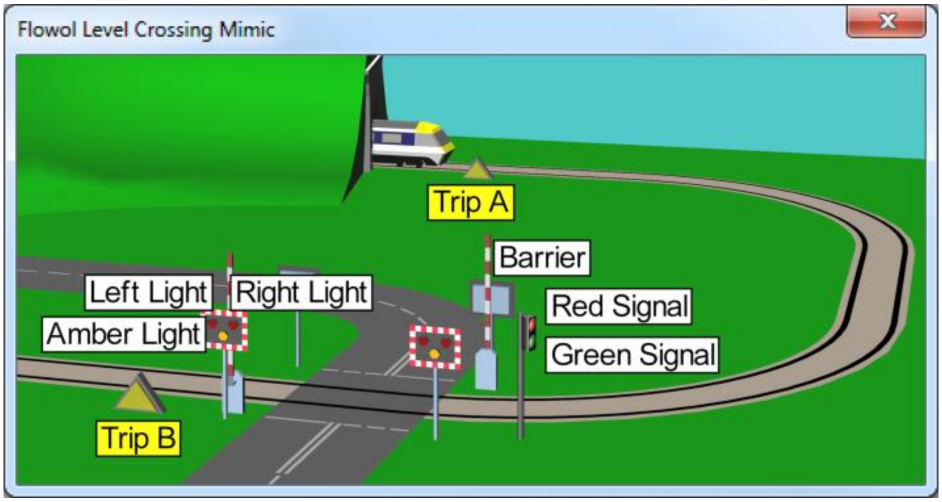
NAME ____________________
CLASS ____________________
DATE ____________________



Level Crossing

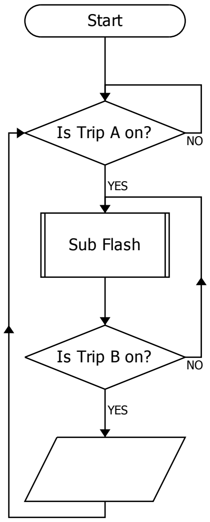
See p. 35 in the tutorial for more help.

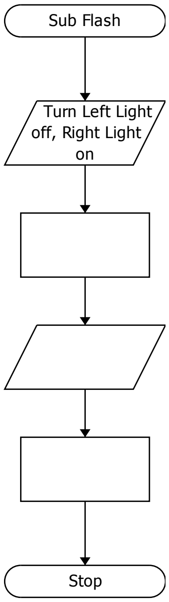
Open the **Level Crossing** mimic.


Flowol Level Crossing Mimic

There are two inputs: **Trip A** and **Trip B** which cannot be clicked on. Instead they are triggered when the train goes past.

The train runs automatically whenever the flowchart is running. Click Run ▶ now, even without any flowchart, to see the train move.





Activity 1

Construct and complete the flowchart to the left to flash the **Left Light** and **Right Light** outputs when the train approaches, and turn them off when the train has passed by.

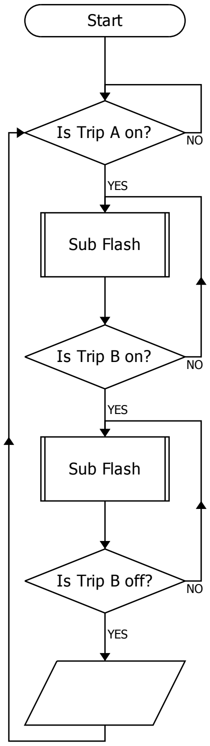
Activity 2

Add symbols to control the **Amber Light**.

Activity 3

The solution above is not quite correct since the lights stop flashing as soon as the front of the train reaches **Trip B**.

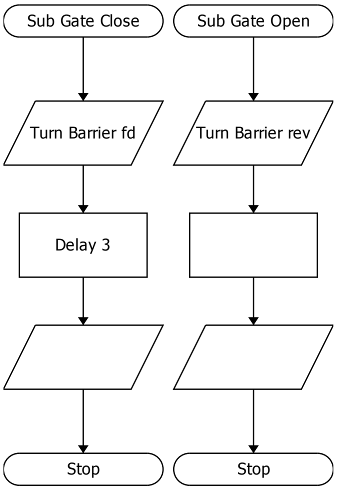
Modify the main flowchart to match the one on the right to correct this.



Remember to save your flowchart!

Activity 4

The **Barrier** motor moves the gate. Turn it forward for a certain time period to close the gate and reverse to open the gate.



Create two subroutines, **Gate Close** to close the gate, and **Gate Open** to open it. Call (invoke) these subroutines from the main flowchart to close and then reopen the gate.

Replace the Delay in the **Gate Close** subroutine to call the Flash subroutine an appropriate number of times to keep the lights flashing while the barriers are closing.

Activity 5

The train can be stopped in an emergency with the signal. To stop the train, turn **Red Signal** on.

Add output symbols in appropriate places to your flowchart so that the Red Signal is shown whenever the barrier is not down.

To verify that the signal works correctly, reduce the speed of the barrier so that it doesn't have time to completely close by the time the train arrives at the signal.

Activity 6 (Advanced)

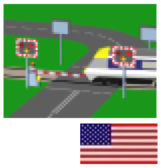
Adjust the mimic options (click the button on the Status Panel) to use feedback switches on the barriers.

Remember to save your flowchart!

Re-run your existing flowchart to see how the barriers vibrate when they reach the feedback switches. Modify your flowchart to use the feedback switches to stop the barrier movement.

Now that the feedback switches are used to control the barriers, it's hard to keep the lights flashing because you don't know exactly how long it will take to lower the barrier. Therefore, create a separate, parallel flowchart which flashes the lights whenever variable **x = 1**. Then set variable **x** to be **1** in the main flowchart to start flashing the lights, and set it back to **0** to stop the lights flashing.

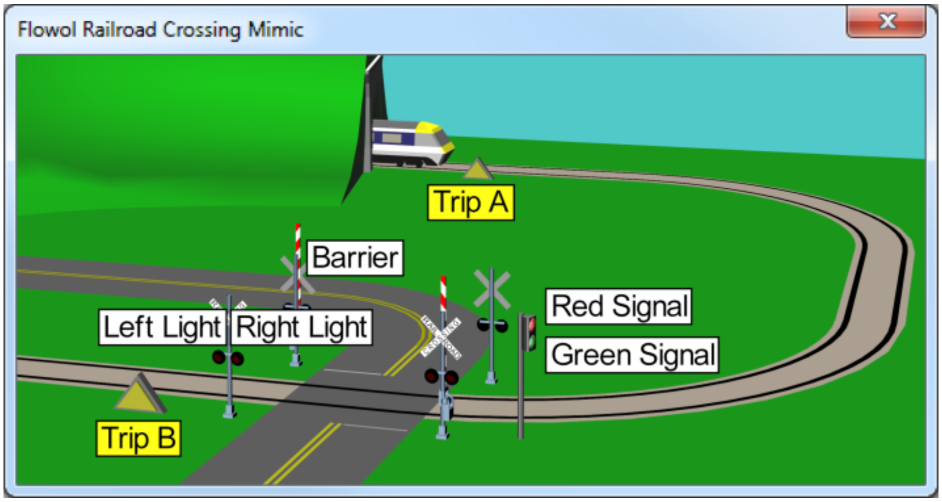
NAME ____________________
CLASS ____________________
DATE ____________________



Railroad Crossing

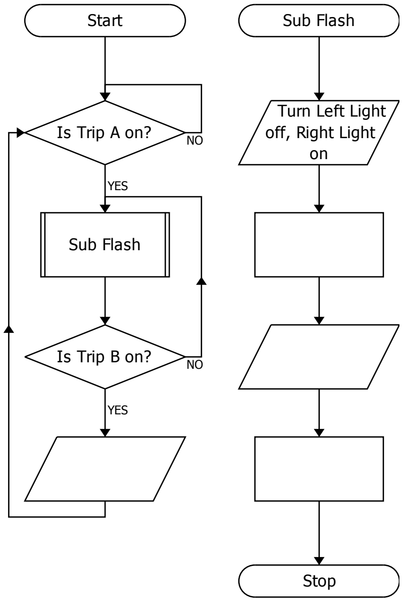
See p. 35 in the tutorial for more help.

Open the **Railroad Crossing** mimic.



There are two inputs: **Trip A** and **Trip B** which cannot be clicked on. Instead they are triggered when the train goes past.

The train runs automatically whenever the flowchart is running. Click Run ▶ now, even without any flowchart, to see the train move.



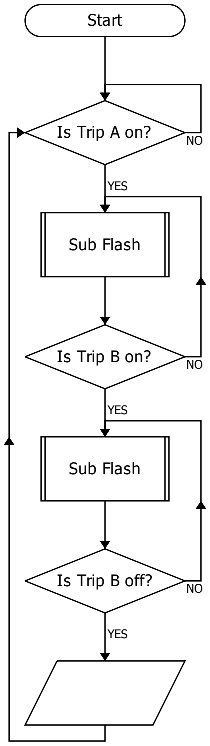
Activity 1

Construct and complete the flowchart to the left to flash the **Left Light** and **Right Light** outputs when the train approaches, and turn them off when the train has passed by.

Activity 2

The solution above is not quite correct since the lights stop flashing as soon as the front of the train reaches **Trip B**.

Modify the main flowchart to match the one on the right to correct this.



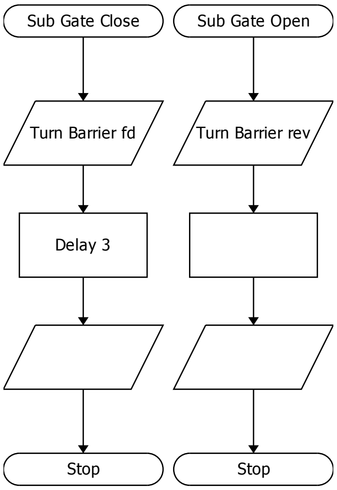
Remember to save your flowchart!

Activity 3

The **Barrier** motor moves the gate. Turn it forward for a certain time period to close the gate and reverse to open the gate.

Create two subroutines, **Gate Close** to close the gate, and **Gate Open** to open it. Call (invoke) these subroutines from the main flowchart to close and then reopen the gate.

Replace the Delay in the **Gate Close** subroutine to call the Flash subroutine an appropriate number of times to keep the lights flashing while the barriers are closing.



Activity 4

The train can be stopped in an emergency with the signal. To stop the train, turn **Red Signal** on.

Add output symbols in appropriate places to your flowchart so that the Red Signal is shown whenever the barrier is not down.

To verify that the signal works correctly, reduce the speed of the barrier so that it doesn't have time to completely close by the time the train arrives at the signal.

Activity 5 (Advanced)

Adjust the mimic options (click the button on the Status Panel) to use feedback switches on the barriers.

Remember to save your flowchart!

Re-run your existing flowchart to see how the barriers vibrate when they reach the feedback switches. Modify your flowchart to use the feedback switches to stop the barrier movement.

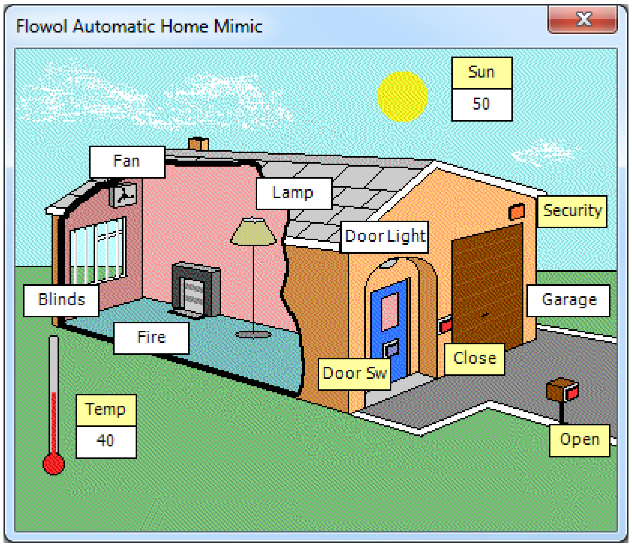
Now that the feedback switches are used to control the barriers, it's hard to keep the lights flashing because you don't know exactly how long it will take to lower the barrier. Therefore, create a separate, parallel flowchart which flashes the lights whenever variable **x = 1**. Then set variable **x** to be **1** in the main flowchart to start flashing the lights, and set it back to **0** to stop the lights flashing.

NAME ____________________
CLASS ____________________
DATE ____________________

Automatic Home

What automatic control features do you have in your home? What control features do you want?

Open the **Automatic Home** mimic and explore.


Flowol Automatic Home Mimic

Activity 1

Construct a program to open the garage door when the **Open** button is pressed, and close it when the **Close** button is pressed.

Activity 2

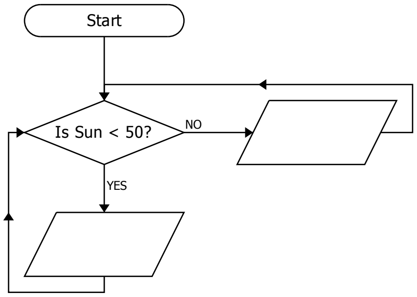
Assume that the **Security** digital input is a movement/infra-red heat sensor which can detect a person on your driveway. Construct a program to turn on the **Door Light** when a person is detected.

Daylight Brightness and Temperature

By clicking on the numbers near the **Sun** and **Temp** thermometer, you can make the sensor readings increase or decrease in steps of 5. The sensors are calibrated by default as a percentage. If you have interface hardware, connect it and calibrate the temperature sensor to °Celsius or °Fahrenheit.

Activity 3

Construct a flowchart, like the one shown, to turn on the inside **Lamp** only when the daylight **Sun** value goes below 50%.



Activity 4

Construct another flowchart to make the electric **Fire** come on when the temperature goes lower than 30 units.

Activity 5

Now control the electric cooling fan above the window. Think about the temperature threshold that you choose.

Remember to save your flowchart!

□

Activity 6

What else can we do? The window **Blinds** can be controlled electrically. Make them automatic.

□

Activity 7

Look at Activity 2 again, the one with the person detector. How can you improve this system?

□

Activity 8

Make the push button (**Door Sw**) on the front door trigger the sound of a bell, or a recording of a dog barking.

Monitoring and Logging

The home can be monitored and data logged using the Flowol Graph feature. Configure the graph window to show the information you want to observe.

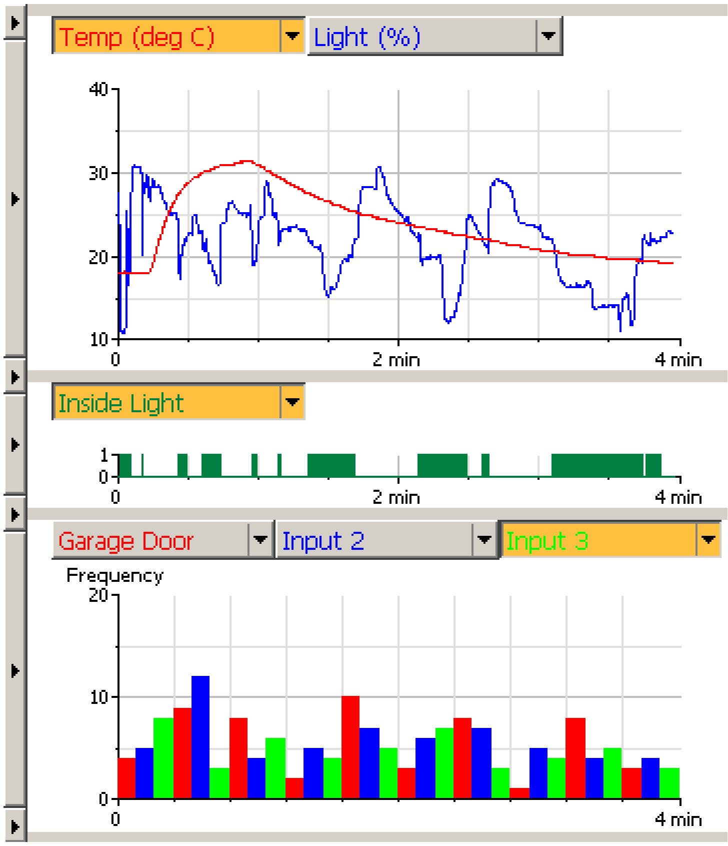
See page 70 in the tutorial for help using graphs.

□

Activity 9

Re-run your program and keep changing the different inputs. The graph shown here is logging temperature, daylight brightness, the inside light and how often the garage door is opened.

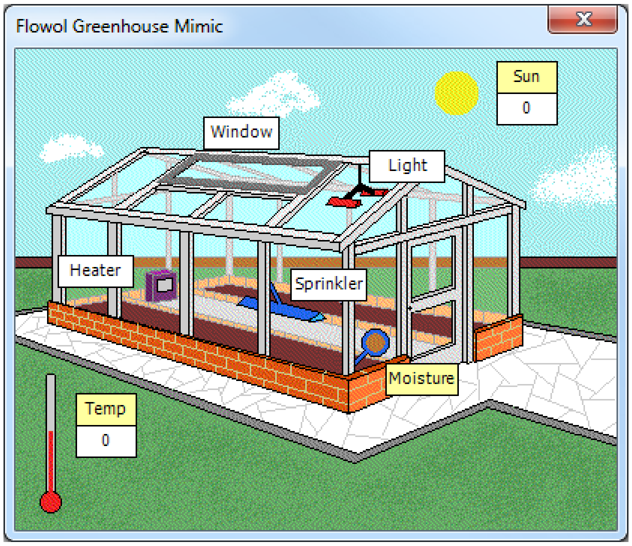
Remember to save your flowchart!



NAME ____________________
CLASS ____________________
DATE ____________________

Greenhouse

Why do we have greenhouses? Do some research and determine the best conditions for growing plants.



Open the **Greenhouse** mimic and explore its functions.

Moisture is a digital input that is off when the soil is dry and on when it is moist. The ambient temperature and light can also be measured with the analog **Temp** and **Sun** sensors.

You can control the **Lights**, **Heater**, **Window** and **Sprinkler**.

Assignment

This is an open assignment. Apply your knowledge and skills to make this greenhouse look after the plants for you.

Remember to save your flowchart!

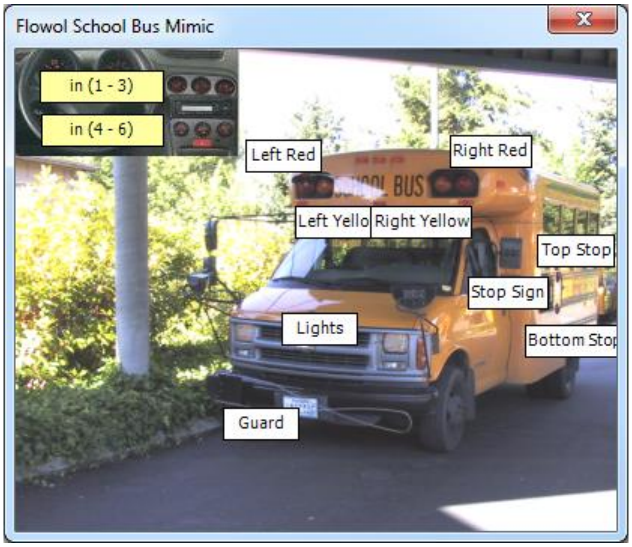
Make sure to label the different flowcharts to show clearly what you are trying to achieve.

NAME ______________________
CLASS ______________________
DATE ______________________

School Bus

How do the special features on a school bus keep the students safe?

Open the School Bus mimic and explore its features.



The driver has six buttons available to control the various lights, **Stop Sign** and **Guard** paddle. These buttons should be used in sequence to operate the safety features in the right order.

Activity 1

Use the first button and build a program to operate the general front **Lights**.

Activity 2

Create another program to control the flashing yellow lights to indicate to passing motorist that the bus is about to stop.

Activity 3

The **Guard** paddle should then be deployed to ensure the children cannot pass near to the front of the bus. Why would crossing just in front of the bus be a hazard?

Activity 4

The red alternating traffic warning lights should then be activated.

Activity 5

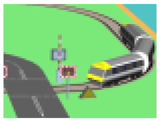
The **Stop Sign** should then be deployed.

Activity 6

Finally the alternating lights on the stop sign should be illuminated.

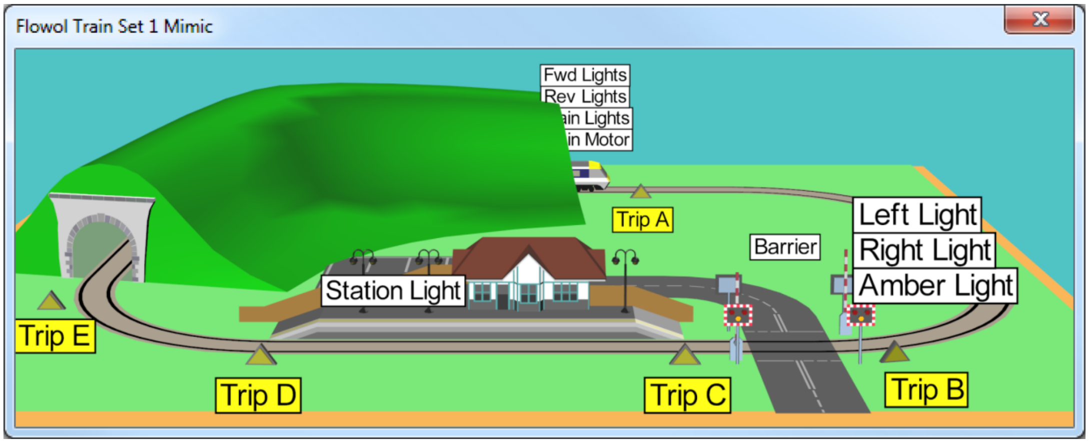
Remember to save your flowchart!

NAME ____________________
CLASS ____________________
DATE ____________________



Train Set 1

Open the Train Set 1 mimic, add the labels and use the Status Panel to explore what the mimic can do.

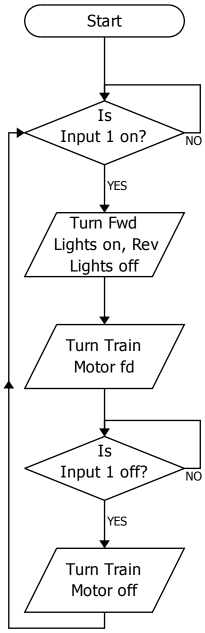


Now imagine that you are the engine driver, the guard, the signal operator and station manager and manually control the train, its lights, the station lights, the crossing lights and of course the barrier gates. How easy did you find that?

You would probably be more successful, and safer, if you are just the engine driver and the other functions are controlled by a system created by you using Flowol 4.

Activity 1

Since you are the engine driver, build this flowchart allowing you to use **Input 1** on the input Status Panel to control the clockwise movement of the train. What do the **Fwd Lights** do?



Activity 2

Now build a similar program to control the reverse movement of the train by using **Input 2**.

Trip Switches

For the next activities we will assume that the train will be moving clockwise around the track.

You may have noticed that the moving train turns several input switches on automatically when it passes over them (the yellow triangles light up).

Activity 3

To be energy efficient, build a program to turn the passenger carriage lights (**Train Lights**) on automatically only while the train is passing through the tunnel.

Note that the carriage lights should go on when the front of the train enters the tunnel and go off again when the back of the train leaves the tunnel.

See page 95 for an example of a similar flowchart.

Activity 4

Now build a similar program to turn the **Station Lights** on only while the train is passing or is stopped at the platform.

Activity 5

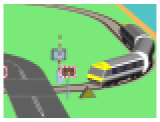
Apply what you learned from the **Level Crossing/Railroad Crossing** mimic to control the barrier and lights for vehicular traffic.

Activity 6

Perhaps you can add an air horn sound to your program when the train passes through the tunnel. You could produce your own sound using a microphone and suitable recording software, or look for a .wav file from the internet.

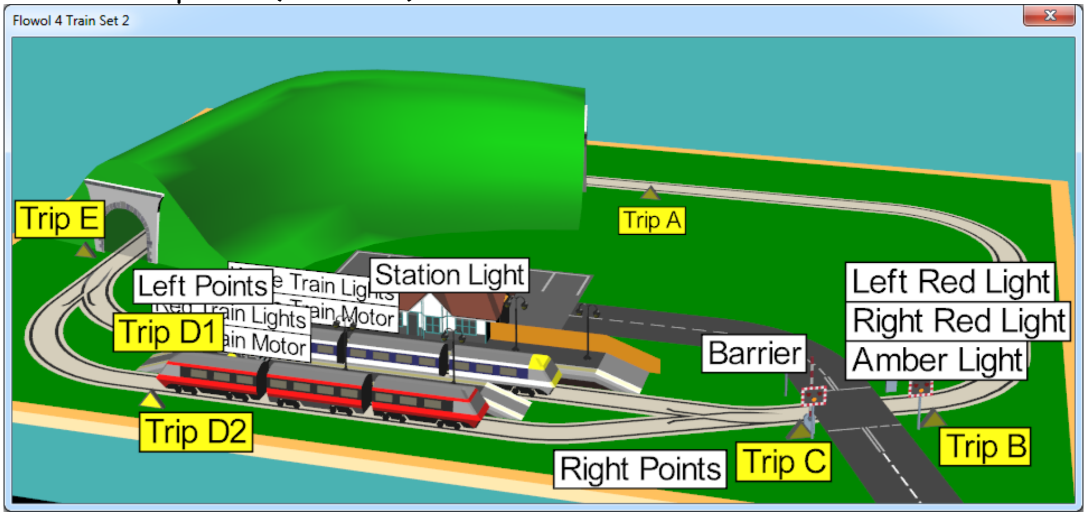
Remember to save your flowchart!

NAME ______________________
CLASS ______________________
DATE ______________________



Train Set 2

The Train Set 2 mimic is more complicated. There are now two trains, and two sets of points (switches).



Each train has its own motor and lights. Click on the motors in the Status Panel to drive each train (right-click the mouse to turn the motor in reverse).

Each set of points (switches) are controlled by the **Left Points** and **Right Points** motors. Turn the motor forward to switch to the outside track and reverse to switch to the inside track.

Activity 1

Create two sub-routines, one for switching to the outside track, and another to switch to the inside track.

Activity 2

Create a master program which drives each train around the track one at a time.

Activity 3

Incorporate all of the features from the TrainSet1.

Activity 4

Have fun!