



Challenge: Pet Car Alarm

TI-84 PLUS CE

TI-INNOVATOR™ STEM PROJECT

STUDENT ACTIVITY

Pet Car Alarm: Challenge #2

Goals:

In the Pet Car Alarm project, you will create a system that detects if the pet is in the car and measure the temperature. The car will sound a warning with flashing lights and roll down the windows if your pet could be in danger

1. Use the SET LED command to turn led lights on and off.
2. Use a For loop to repeat code.

In this challenge, you will use the TI-Innovator and external LED's to send messages. This skill will be used when you design your warning system for the Pet Car Alarm.

Background:

You use pre-programmed lights every day. Does your phone light up when someone texts? Without turn signals how would other motorists know to slow when a car in front of them plans to turn? Some lights such as the one in the refrigerator turn on and off with actions such as opening and closing the door. Other lights, such as traffic lights, cycle through a set pattern. Some lights are programmed with an action/consequence plan meaning they don't change their state until something occurs. Others are pre-set to run in a pattern. Brainstorm with a partner various lights you encounter throughout the day. Do they change based on an action or do they have a pre-defined pattern?

What are three lights you encounter in a day that change based on an action?

What are three lights you encounter in a day that have a preset pattern?

Command	Example	Behavior
CONNECT <type> <number> TO <port>	<code>Send("CONNECT TEMPERATURE 1 TO IN1")</code>	Associates the first TEMPERATURE object with a temperature module plugged into port IN1 on the Hub.
SET <type><number> TO <value>.	<code>Send("SET LED 1 TO ON")</code>	Turns on LED 1. Other parameters may include BLINK <rate> TIME<duration> "SET LED 1 ON BLINK 3 TIME 20"
SET(<type><number> TO <value> [<blink rate>] [<duration in seconds>])	<code>Send("SET LED 1 ON BLINK 3 TIME 20")</code>	Sets LED 1 to a blink rate of 3 times per second for 20 seconds. See Hub Settings menu for ON, OFF, BLINK, etc.
Wait <number>	<code>Wait 3</code>	Pauses the program for a specified number of seconds. In this case 3 seconds.
For(<counter variable>,<start value>,<end value>,<step value>]) <statements> End	<code>For(n,1,10) Output(3,1,n) End</code>	Runs For loop 10 times, starting at 1 and ending at 10. Executes the statement in the block each time, displays the value for the counter variable on row 3.
While <Boolean expression> <statements> End	<code>1→k 220→f While k ≠ 45 Send("SET FREQUENCY eval(f)") getKey → k End</code>	The statements in the While loop are executed until the key 45, the clear key, is pressed. The While loop continues as long as the Boolean expression evaluates to "true". The variable <i>k</i> is set to an initial value of 1 using the store function, →. getKey is a function that returns a number with the value of the last key pressed while a programming is running. In this program the value of getKey is stored to the variable <i>k</i> .
If <Boolean expression> Then <statements 1> Else <statements 2> End	<code>If a=1 Then Output(5,1,"Light On") Else Output(5,1,"Light Off") End</code>	The example decision tree has a Boolean expression with corresponding statements to execute if true. It also has an Else condition that executes corresponding statements when the Boolean expression is false. This Else condition ensures that a set of statements will always be executed. When this decision tree executes, focus proceeds from top-down. If the value of <i>a</i> equals 1 the commands after Then are executed. In this case the output "Light On" is displayed on row 5. If the value of <i>a</i> is not 1 the commands after Else are executed. In this case the output "Light off" is displayed on row 5.



Challenge: Pet Car Alarm

TI-84 PLUS CE

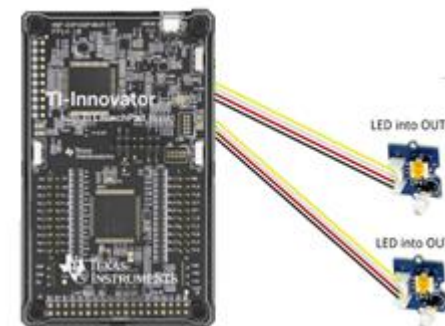
TI-INNOVATOR™ STEM PROJECT

STUDENT ACTIVITY

Challenge:

Challenge 1: Use a For..End to blink two external LED's.

Challenge 2: Try Challenge 2a once more using the “BLINK” and “TIME” options.
This time you can do it without a For Loop.



Extension 1: Write a program that lets the user press 1 to turn on the external LED's and 0 to turn them off. Your IF THEN statements should be inside a while loop that continues to execute until the user presses esc.

Extension 2: Have you ever heard of Morse Code? Morse Code uses a combination of dots and dashes to represent letters and numbers. An S in Morse Code is · · · (dot dot dot) while an O is - - - (dash dash dash). Sending an SOS is the international sign of distress. Write a program that sends out an SOS using three loops, one for each letter.