## Pet Car Alarm

In this TI-Innovator™ project, you will explore the science behind the greenhouse effect and apply your knowledge to design a product to solve a real world problem of pets dying due to owners leaving them in hot cars. You will have to utilize math skills, computer programming and engineering to design and build a smart pet alarm system for a model car. A car equipped with a "smart" pet alarm could prevent harm to a pet left inside a hot car by taking action to cool the interior and notify the owner of impending pet harm.

TEMPERATURE 80° OUTSIDE 123° INSIDE ELAPSED TIME 60 MINUTES
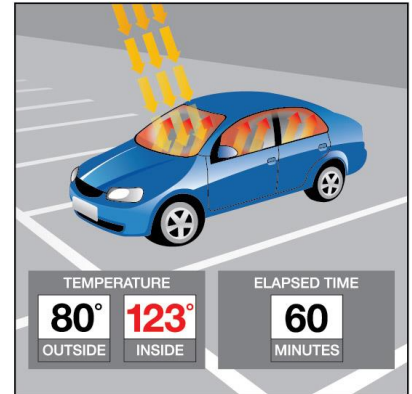
Background:

Pets suffer when left unattended in a car with the windows rolled up on a hot sunny day. The temperatures inside a car may reach greater than $40°$F above the outside ambient temperature within an hour due to a greenhouse effect within the closed car. A car equipped with a pet-smart alarm could prevent harm to a pet left inside a hot car by taking action to cool the interior and notify the owner of impending pet harm.

Your Challenge:

Understand the science behind the greenhouse effect and use math, computer programming and engineering to design and build a pet-smart alarm system using a model car.

Activity Materials:

- TI-Innovator™ Hub (vs.1.3)
- TI-Nspire CX (OS vs. 4.5)
- Hall Effect (magnetic) Sensor
- Temperature Sensor
- 2 White LEDs
- Continuous Servo Motor
- External Battery for TI-Innovator Hub

- ¾" Rare Earth Magnetic disk
- Model Car (other container)
- Plastic Pet
- Plastic (Saran) wrap
- Tape
  Safety Scissors

Project Tasks:

1. Write a program "Hello World" that displays a text message on the calculator using the Disp command.
   - Change task 1 to display a text message multiple times using a For loop.

2. Write a program to display colors on the TI-Innovator Hub.
   - Extension – use a for loop to allow for input from the user to determine color.

3. Write a program to Blink the RGB LED.
   - Extension – use a for loop to allow the RGB LED to blink multiple times.

4. Mini-Project: Create a Traffic Light. Write a program to control the COLOR LED to simulate a traffic light using a single bulb by creating a sequence of statements with proper timing controls.
   - Extension – add displays to show Walk/Don't Walk, add audible signal for blind.

5. Mini-Project: Using what you learned from the example program using the Brightness Sensor, write a program that will use the brightness of the sensor as input to trigger different sounds as output.
   - Extension - Create a function to determine the frequency of the sound that is triggered.

6. Mini-Project: Write a program that connects and reads the temperature of a sensor
   - Write a program that takes inputs from temperature sensors and uses if-then-else decision logic to control an output (sound or color LED) based on the sensor readings.

7. Write a program to connect and read the values of these additional sensors:
   - Connect and use the Hall Effect sensor.
   - Connect an LED and blink it.
   - Connect a Servo motor and make it spin Counter Clockwise

8. Putting it All Together: Use the programming skills you have learned previously, to design and build a Pet Car Alarm using multiple sensors and a toy car, or other material provided.
   - Magnetic Proximity Sensor
   - External digital outputs, LED's.
   - External analog outputs, (sweep) servo motor, with External battery
   - Temperature Sensor
   - Extension – Add a sound component. Write a version using a While loop that runs until an alarm is triggered. Add another temperature sensor to represent "outside" of the car temperature

9. Pitch your product to your peers!
   - What makes your product unique?
   - How does your product work?
   - Why is it better than the next "safety device" for sale?
   - Consider a unique name for your product, customization, etc.
   - Create a slogan, jingle and/or commercial for your product.

Example TI-BASIC Code for TI-Nspire CX:

Code snippet to CONNECT the sensors:

```
Send "CONNECT LED 1 TO  OUT 1 "
Send "CONNECT LED 2 TO  OUT 2 "
Send "CONNECT SERVO  TO  OUT 3 "
Send "CONNECT TEMPERATURE  1 TO  IN 1 "
Send "CONNECT ANALOG.IN 1 TO  IN 3 "
```

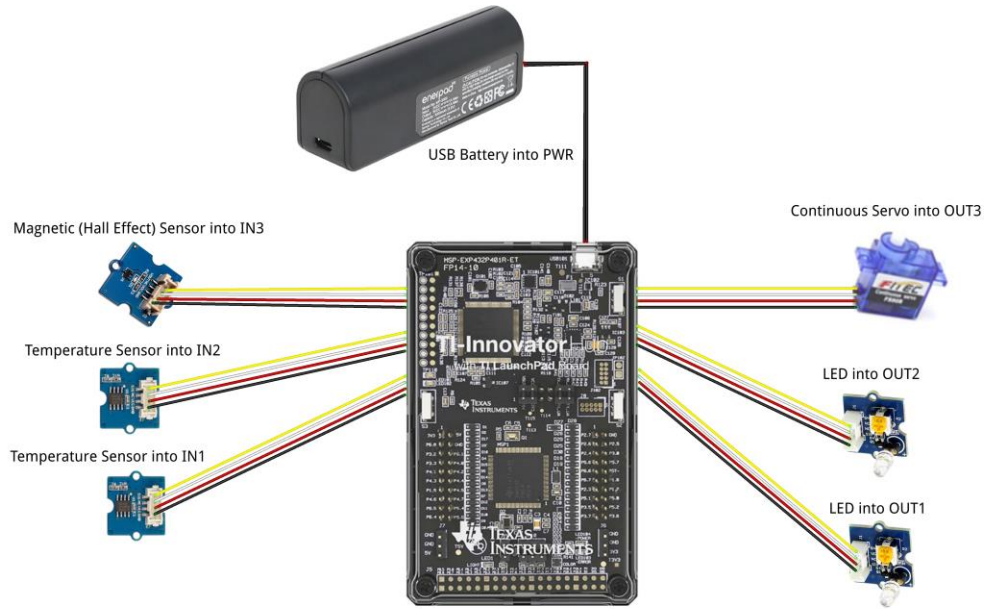Note: ANALOG.IN 1 is the Hall Effect sensor.

Code snippet to READ the sensors:

```
Send "READ TEMPERATURE 1"
Get i
Disp i
Send "READ TEMPERATURE 2"
Get o
Disp o
Send "READ ANALOG.IN 1"
Get a
If a < 100 Then
 Disp "Pet in Car"
EndIf
```

Code snippet to SET the actuators:

```
Send "SET LED 1 ON  BLINK 5"
Send "SET LED 2 ON  BLINK 5"
Send "SET SERVO 1 TO 90"
Wait 2
Send "SET SERVO 1 TO 0"
```

To break an infinite loop, press and hold the ON key until you receive a Quit option

Sensor and actuator Hub connections:

Magnetic (Hall Effect) Sensor into IN3
Temperature Sensor into IN2
Temperature Sensor into IN1
USB Battery into PWR
Continuous Servo into OUT3
LED into OUT2
LED into OUT1

Possible Control Algorithim:

- Connect all sensors and actuators
- For n = 1 to 500
- READ all sensors from Hub
- Get all sensor readings and display values
- T In >30 and Hall <100
  - Yes: SET SERVO to 90 and blink LEDs → Disp "Pet is too Hot!"
  - No: Disp "Pet is OK"
- Wait 1 sec
- Next
- End of Program