Finch Class Methods

Constructors

Method Signature: `Finch()`
Description: Constructor that creates an object corresponding to a Finch. This method assumes that the Finch is the only device connected in the BlueBird Connector.
Example: `bird = Finch()`

Method Signature: `Finch(device)`
Description: Constructor that creates an object corresponding to a Finch. This method requires a string equal to ‘A’, ‘B’, or ‘C’ that specifies the letter of the device in the BlueBird Connector.
Example: `bird = Finch(‘A’)`

Output Methods

Method Signature: `setMove(direction, distance, speed)`
Description: Moves the Finch forward or backward for a specified distance at a specified speed. The method requires a direction (‘F’ for forward or ‘B’ for backward), a distance in centimeters, and a speed from 0-100.
Example: `bird.setMove(‘F’,10,50)`

Method Signature: `setTurn(direction, angle, speed)`
Description: Turns the Finch right or left for a specified angle at a specified speed. The method requires a direction (‘R’ for right or ‘L’ for left), an angle in degrees, and a speed from 0-100.
Example: `bird.setTurn(‘R’,90,50)`

Method Signature: `setMotors(leftSpeed, rightSpeed)`
Description: Sets the Finch wheels to spin at the given speeds. The method requires two speeds between -100 and 100 for the left and right wheels. Setting the speed to 0 turns the motor off.

Example: `bird.setMotors(-50,50)`

Method Signature: `stop()`
Description: Stops the Finch wheels.
Example: `bird.stop()`

Method Signature: `setBeak(redIntensity, greenIntensity, blueIntensity)`
Description: Sets a tri-color LED in the Finch beak to a given color by setting the intensities of the red, green, and blue elements inside it. The method requires three intensity values from 0-100. Setting all three intensity values to 0 turns the beak off.
Example: `bird.setBeak(0,100,0)`

Method Signature: `setTail(port, redIntensity, greenIntensity, blueIntensity)`
Description: Sets a tri-color LED in the Finch tail to a given color by setting the intensities of the red, green, and blue elements inside it. The method requires the port number of the LED (1, 2, 3, 4, or “all”) and three intensity values from 0-100. Setting all three intensity values to 0 turns the LED off.
Example: `bird.setTail(“all”,0,100,0)`

Method Signature: `playNote(note, beats)`
Description: Plays a note using the buzzer on the Finch. The method requires an integer representing the note (32-135) and a number giving the number of beats (0-16). The number of beats can be a decimal number. One beat corresponds to one second.
Example: `bird.playNote(60,0.5)`

Method Signature: `setDisplay(LEDlist)`
Description: Sets the LED array of the micro:bit to display a pattern defined by a list of length 25. Each value in the list must be 0 (off) or 1 (on). The first five values in the array correspond to the five LEDs in the first row, the next five values to the second row, etc.
Example: `bird.setDisplay([1,1,1,1,1,0,0,0,0,0,1,1,1,1,1,0,0,0,0,0,1,1,1,1,1])`

Method Signature: `setPoint(row, column, value)`
Description: Turn on or off a single LED on the micro:bit display. The position of the LED is given by the row and column parameters, which should both be between 1 and 5. The value of the LED must be 0 (off) or 1 (on).
Example: `bird.setPoint(3,3,1)`

Method Signature: `print(message)`
Description: Print a string on the micro:bit LED array. The string must have 15 or fewer characters and should contain only digits and English letters (upper or lower case).
Example: `bird.print("hello")`

Method Signature: `stopAll()`
Description: Stops all outputs. This includes the LED display for the micro:bit and all lights and motors for the Finch.
Example: `bird.stopAll()`

Input Methods

Method Signature: `getDistance()`
Description: Returns the distance in centimeters to the closest object.
Example: `print("Distance to obstacle: ", bird.getDistance())`

Method Signature: `getLight(direction)`
Description: Takes a string corresponding to a direction (‘R’ for right or ‘L’ for left) and returns the intensity measured by the corresponding Finch light sensor. The light measurement is an integer between 0 and 100 (arbitrary units).
Example: `print("Left light sensor: ", bird.getLight(‘L’))`

Method Signature: `getLine(direction)`
Description: Takes a string corresponding to a direction (‘R’ for right or ‘L’ for left) and returns the intensity measured by the corresponding Finch line tracking sensor. The line measurement is an integer between 0 and 100 (arbitrary units).
Example: `print("Right line sensor: ", bird.getLine(‘R’))`

Method Signature: `resetEncoders()`
Description: Sets the value of both Finch encoders to 0.
Example: `bird.resetEncoders()`

Method Signature: `getEncoder(direction)`
Description: Takes a string corresponding to a direction (‘R’ for right or ‘L’ for left) and returns the value measured by the corresponding Finch encoder. The encoder measurement is the number of rotations that the wheel has turned since resetEncoders() was last called.
Example: `print("Right wheel rotations: ", bird.getEncoder(‘R’))`

Method Signature: `getButton(button)`
Description: Takes a string corresponding to a micro:bit button (‘A’,‘B’, or ‘Logo’) and returns a Boolean value that is True if the button is being pressed, and False otherwise. The Logo button is only present on a micro:bit V2, so this option cannot be used with a micro:bit V1.
Example: `print("Button A is pressed: ", bird.getButton('A'))`

Method Signature: `isShaking()`

Description: Returns True if the Finch is shaking and False otherwise.

Example: `print("Shake Status: ", bird.isShaking())`

Method Signature: `getOrientation()`

Description: Returns a string that represents the orientation of the Finch. The possible values are “Beak up”, “Beak down”, “Tilt left”, “Tilt right”, “Level”, “Upside down”, and “In between”.

Example: `print("Orientation: " + bird.getOrientation())`

Method Signature: `getAcceleration()`

Description: Returns a list that contains the Finch acceleration in m/s^2 in the *x*, *y*, and *z* directions.

Example: `print("Acceleration: ", bird.getAcceleration())`

Method Signature: `getCompass()`

Description: Returns the direction of the Finch in degrees from magnetic north (0°-359°). The compass should be calibrated in the BlueBird Connector before using this method.

Example: `print("Compass Heading: ", bird.getCompass())`

Method Signature: `getMagnetometer()`

Description: Returns a list that contains the value of the magnetic field in µT in the *x*, *y*, and *z* directions. The compass should be calibrated in the BlueBird Connector before using this method.

Example: `print("Magnetic Field: ", bird.getMagnetometer())`

Method Signature: `getSound()`

Description: Returns the sound measured by the micro:bit. The sound measurement is an integer between 0 and 100 (arbitrary units). This function can only be used with the micro:bit V2.

Example: `print("Sound: ", bird.getSound())`

Method Signature: `getTemperature()`

Description: Returns the temperature in degrees Celsius measured by the micro:bit. This function can only be used with the micro:bit V2.

Example: `print("Temperature: ", bird.getTemperature())`

Back to Top