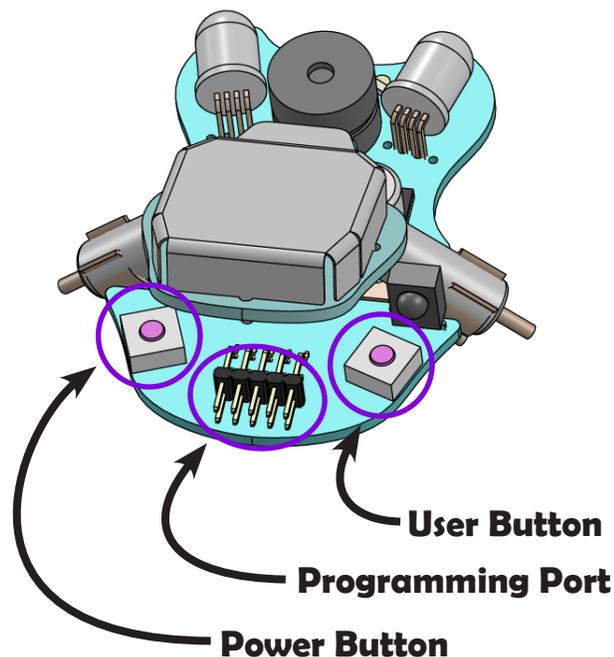




Learning Wink's Basic Operation

Before you get started, please have a read over this quick guide to get started working with your Wink robot.



Power Button

Press the Power Button labeled "PWR" for about 1/2 second to turn on Wink's power.

A power indicator LED just forward of the power button will light and remain lit while the power is on.

To turn the power back off, press the Power Button again for a brief time. The power indicator LED will turn off.

The power must be on to successfully upload code to Wink.

Programming Port

Connect the ribbon cable from the programming adapter to these pins.

BE SURE TO CONNECT TO ALL 10 OF THE PINS.

It is easy to accidentally connect to just the top row of pins.

Uploading new code can be started at any time, even while other code is already running on the robot. The programmer automatically resets the robot before loading new code.

User Button

The user button can be used to index between the pre-loaded behaviors shipped with Wink.

The button can also be read by your code and used to make the robot carry out or change behaviors. Have a look at Learn to Code Lesson 4 to see how to use the button.



Pre-Loaded Demo Examples

Wink is shipped with three demo behaviors pre-loaded so you can start having fun as soon as you open the box. You can index between the demo behaviors by pressing the “BTN” button on the rear right side. When changing to the next behavior, Wink will chirp and his eyes will blink green.

You can download the source code for these demos on our website. You can then expand or customize them further. Have fun!

Demo 1: Wink Dance

The Wink dance is a simple use of the Arduino random number generator. The code works by generating random numbers which are used to randomly set Wink’s eye color and left and right motor speeds. This results in a buggy dance. It turns out that this number generator is “pseudo” random. This means it isn’t really random at all, but follows a predictable sequence. For this reason, if you turn on two or more Winks at exactly the same time, you’ll notice they have the exact same colors and movements - so they dance together. It’s also worth knowing that we set the timing of these movements to match 130 bpm music, which is a common tempo for all kinds of electronic, techno, and dubstep music. Put on some beats in the background and it will look as if Wink is moving to the music.

Demo 2: Obstacle Avoidance

In this example, place Wink on a running surface where he is not directly pointed at any obstacle or barrier. Be sure to move your hand out of the way also. Just before he starts moving, he will sample the reflectivity of the surface as a baseline. He will then begin driving. He will use his IR headlight to sense light reflecting from things in front of him. Once he sees a certain amount of light, he will reverse and attempt to turn away from the obstacle. He sees light colored objects further away than dark colored objects because light colors reflect more light. Set up a maze, or place your hand in front of him. You can edit this code to change how he reacts and how sensitive he is. In bright conditions, he may continually try to back up, and bright light from a window may confuse him. This behavior works well in normal room lighting.

Demo 3: Light Seeking

In this example, Wink will look for light. This is a lot of fun in a dim lit room with a light source like a small flashlight. You’ll notice his eyes get brighter and he goes faster the closer he gets to the light. This way he tends to “pounce” on the light when seen. Remember that his light sensors are most sensitive looking out along his running surface. They are less sensitive from directly above. Try placing the light source on the running surface and shining it toward him for the best response, rather than shining directly down from on top of him.

Getting Started with Wink



Setting up Wink's Software

After you play with the pre-loaded behaviors for a while, please visit www.plumgeek.com/wink and follow the "Getting Started" link. This page is updated if any aspects of the setup or versions of software change over time.

We also provide a Getting Started Video on the setup page. We highly recommend you watch this video as it will guide you through the software and setup process. The setup video also shows how to connect the programming adapter and ribbon cable.

Getting started with the code...

As you begin to write your own code, you should begin by downloading and opening the BaseSketch, which can be downloaded from www.plumgeek.com/wink. This sketch is a template you can use to begin writing your own code.

The BaseSketch includes some additional files which will open inside Arduino as tabs across the top. These extra files include the back-end functions you will be using as you write your own code and move through the Wink Learn to Code lesson series.

Note that you should open these sketches from within Arduino (rather than browsing to them directly with Windows Explorer or other tool). If you do not open the sketches from within Arduino, these extra tabs may not open on some operating systems, resulting in failures to complete the code.

You can also find other interesting software sketches on this page, including the Pre-Loaded Demo, which is a copy of the software that was loaded on Wink when he was shipped. Feel free to edit and customize!

Software File	Description	Demo
Base Sketch	Use the Base Sketch to begin writing your own code. The project automatically includes the supporting tabs that include Wink's background functions.	
Pre-Loaded Demo	This is the code that was loaded on your Wink when it was shipped. The basic behaviors include the Wink Dance, Barrier and Object Avoidance, and Light Seeking. You can index between the three behaviors by pressing the BTN button on the robot.	
Drive with IR Remote	This demo allows you to drive an IR equipped Wink with the optional remote control. Use this sketch as a basis for making your own remote control triggered behaviors.	
ZOMBIE Wink!!	This is a fun behavior we created for Arduino Day 2016. Load this on two or more Wink robots and watch the Zombie Apocalypse happen!	View on YouTube

Web Page: www.plumgeek.com/wink

Getting Started with Wink



Wink Lessons

Wink was designed around a series of free and open source coding lessons we have been writing and continue to write. This getting started guide is actually the first of these lessons - so you're already on your way! Once you get the software set up, we suggest you visit the Getting Smart link from the main Wink page. That will take you to the Wink Learn to Code Lessons page where you can continue the series.

```
void loop(){
  motors(150,150);      //Both motors forward at 150 speed
  delay(1000);         //wait 1 second
  motors(-150,-150);   //Both motors backward at 150 speed
  delay(1000);         //wait 1 second then loop
}
```

Wink_Ch03Motors_Ex01

Note that each lesson includes code snips as examples. Some snips include a label under the lower right corner. These labels correspond to the example being written as a complete functioning sketch in the companion code section. Click the [Download] link on our website next to each lesson to get the companion code for that lesson. The companion sketches are named the same as the snip labels in the written lessons.

on your own. You may view the lessons individually from the table below. The companion code is a set of complete working Arduino sketches for each of the examples in the lessons. We have also published all the lessons and companion code together in single .zip archives below.

- Download the entire archive of lessons [here](#).
- Download the entire archive of Companion Code [here](#).



Lesson	Description	Companion Code
Lesson 1: Basics	Get started and learn the basics of the Arduino code environment.	N/A
Lesson 2: Eyes	Learn how to control Wink's eyes.	[Download]
Lesson 3: Motors	Learn how to make Wink's motors work.	[Download]

Web Page: www.plumgeek.com/learn-to-code

1
SKILL
LEVEL



Troubleshooting

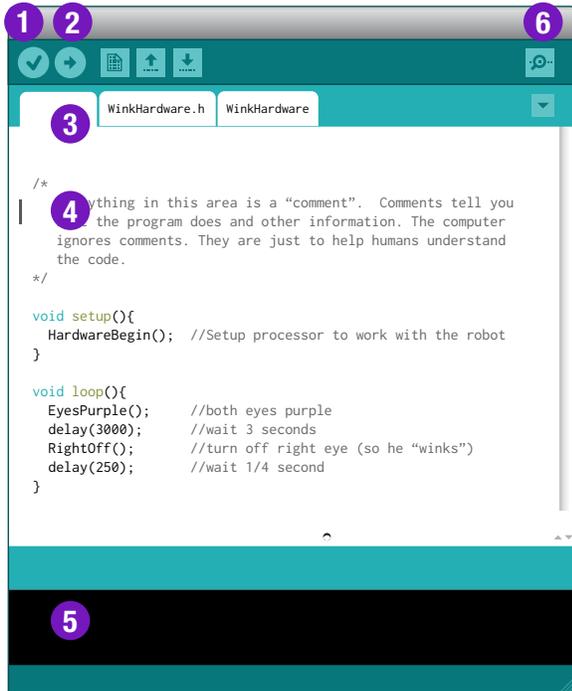
Normally it's a smooth process to get started with Wink. Here are a few items that may be accidentally overlooked. If you have any trouble uploading code to Wink, review these quick points.

1. Be sure you have selected "Arduino Fio" in the Tools>Board menu inside Arduino.
2. Be sure the robot is turned on while uploading. Note there are TWO indicator lights on the robot. The Power indicator is just forward of the PWR button. The Charging indicator is just to the right of the programming port. Be sure the power indicator forward of the PWR button is lit.
3. Be sure the ribbon cable is connected to all 10 pins on the programming port, as well as all 10 pins on the programming adapter. It is easy to connect it to just the top row of pins.
4. A failed upload due to any of the above problems will produce a lot of "skt_500" errors in the Arduino window. If an upload fails for this reason, it takes a while before the Arduino software will reset the connection (we hope Arduino will fix this soon). To immediately reset the connection and quickly begin a new upload attempt, you can disconnect and re-connect the USB cable to the programming adapter. Once this is done you should be able to immediately begin a new upload to the robot.
5. Be sure you have selected the new port that appears under Tools>Port in Arduino when you connect the programming adapter to the USB.
6. If you are still having problems, please visit the Plum Geek web forum. Under the "Wink Basics" topic, there is a sticky topic at the top of the page named "Wink Programmer Issues - Please Read". In this post we have covered the programmer connection in greater detail as well as providing solutions to some of the other issues we occasionally encounter in email support.
7. If all else fails, please contact us through the plumgeek.com website. We haven't encountered a problem yet that we've been unable to resolve.



How the code window is set up...

In the world of Arduino, the window where you write your code is called a “sketch”. You could also call this “source code”. Have a look at how the window is set up.



(1) Press the Check Mark button to “verify” your code. This compiles your code and tells you if you’ve written anything the computer can’t understand.

(2) Press the Arrow button to compile then upload your code to the robot’s brain.

(3) These are tabs. The first one has the name of your sketch. The others are supporting tabs which have background functions inside. More on these in the advanced lessons.

(4) This is your code! It’s the stuff you edit to make the robot do cool stuff. A code example is shown below.

(5) This is the messages area. This area will tell you if you’ve made any mistakes in your code as well as other useful information.

(6) This opens the Serial Monitor, which is a window where the robot can send you messages which are useful when “debugging” your code.

```

/*
  Everything in this area is a “comment”. Comments tell you
  what the program does and other information. The computer
  ignores comments. They are just to help humans understand
  the code.
*/

void setup(){
  hardwareBegin(); //setup processor to work with the robot
}

void loop(){
  eyesPurple(100); //both eyes purple
  delay(3000); //wait 3 seconds
  rightOff(); //turn off right eye (so he “winks”)
  delay(250); //wait 1/4 second
}
    
```

} Use comments to tell yourself or other human readers what the code is doing. In-line comments can be created by using a pair of forward slashes like this: //

} setup() runs one time when the processor first starts. It is useful to configure the initial setup of the processor.

} This is where the fun happens. Everything inside the loop() function is run over and over. You’ll put most of your code inside this function.