



Do something when you press Wink's button...

Up to this point, our examples have just run in a quick loop over and over. Sometimes you may want your program to wait for something to happen - like button press. You can cause your program to stop and wait for a button press with the `waitForButton()` function.

```
void loop(){
  waitForButton();      //wait here till' button pressed
  eyesRed(100);        //make eyes red
  delay(1000);         //wait 1 second
  eyesOff();           //turn eyes back off
}
```

Wink_Ch04Button_Ex01

} Wait for button to be pressed. Program continues as soon as button is pressed, turning eyes on for one second, then back off, then program loops again and waits for another button press.

By waiting for the button press, you can trigger an event (like blinking the eyes) whenever you want to. You may notice in this example, that if you hold the button down and keep holding it, the eyes appear to stay on. They are indeed being turned off for a very short amount of time at the end of the loop after the 1 second delay, but then `waitForButton()` is run again quickly following - if you're still holding the button, the light will be turned back on for another second.

Let's make the light turn back off quickly as soon as you release the button. We can do this by making the delay time much shorter. Lets try `delay(25)` this time.

```
void loop(){
  waitForButton();      //wait here till' button pressed
  eyesRed(100);        //make eyes red
  delay(25);           //wait 25 milliseconds
  eyesOff();           //turn eyes back off
}
```

Wink_Ch04Button_Ex02

} Wait for button to be pressed. Program continues as soon as button is pressed, turning eyes on for 25 milliseconds, then back off, then program loops and again waits for a button press.

You'll notice this version responds more quickly when you release the button. The first version is good if you want the eyes to stay on for a while after a quick press, and the second version is good if you want the eyes to only turn on while you're pressing the button. Once you reach the Skill Level 2 lessons, you'll learn how to use "if" and "else" for an even better way to control this.



Adding sound...

Let's learn to control Wink's beep sound. The functions work like this.

```
beep(200);    //beep for 200 milliseconds
beepOn();     //turn on the beep and leave it on
beepOff();    //turn off the beep sound
```

} Use beep functions to control the sound.

You'll quickly notice that Wink's sound beep is fairly loud. It's a good idea to use the `beep()` function in most cases because it turns on the beep, then turns it back off after a certain number of milliseconds. You can also use `beepOn()` to turn on the beep and leave it on. But don't forget to turn it back off in your code by calling `beepOff()`. Note that the `beep()` function limits the beep time from between 10 milliseconds and 1000 milliseconds. If you give it a number less than 10, it will beep for 10 and if you give it a number larger than 1000, it will only go for 1000 milliseconds. This is to prevent you from accidentally entering a really large number and having to wait through a very long beep.

VERY IMPORTANT WARNING!!! The part that makes the beep sound is intended to be either on or off. Attempting to rapidly turn it on and off to create different pitch tones will likely damage the part. The `beep()` function has built in time delays to make sure this doesn't happen, so you should use the `beep()` function when possible. **DO NOT** attempt to control the beep by using the Arduino `tone()` function or similar code as this will likely damage the part. Also avoid calling `beepOn()` and `beepOff()` in tight loops as this can cause the same problem. It is suggested the beep last no shorter than 10 milliseconds, and that there is a delay of at least 50 milliseconds before it is turned back on. The `beep()` function makes sure these limits are not exceeded.

Let's add sound to our button example from above.

```
void loop(){
  waitForButton();    //wait here till' button pressed
  eyesRed(100);      //make eyes red
  beep(25);          //beep for 25 milliseconds
  eyesOff();         //turn eyes back off
}
```

} `beep()` works just like `delay()` except it turns on the sound during the delay.

Wink_Ch04Button_Ex03

Experiment on your own...

Now you know how to control Wink's eyes, his motors, his sound, and do stuff when the button is pressed. Easy, right? This is a good time to have some fun and experiment on your own. In the next lesson we'll put these ideas together to do something fun.