Using USB Mouse

When you select "USB Keyboard/Mouse" from the **Tools -> USB Type** menu, the Teensy becomes a USB keyboard and mouse while running your program.

Mouse interaction is obviously limited because the Teensy can not view the screen. Nonetheless, you can send mouse actions, which might be useful for some types of projects.

Positioning The Mouse (Teensy LC & 3.x Only)

Teensy 3.x supports positioning. First, use Mouse.screenSize(width, height) to configure Teensy for your screen. Then you can use Mouse.moveTo(x, y) to precisely position the mouse at any pixel on your screen.

```
#include <Bounce.h>

Bounce button4 = Bounce(4, 10);
Bounce button5 = Bounce(5, 10);

void setup() {
  Mouse.screenSize(1920, 1080);  // configure screen size
  pinMode(4, INPUT_PULLUP);
  pinMode(5, INPUT_PULLUP);
}

void loop() {
  button4.update();
  button5.update();
  if (button4.fallingEdge()) {
    Mouse.moveTo(25, 100);       // point to pixel at 25, 100
  }
  if (button5.fallingEdge()) {
    Mouse.moveTo(1580, 14);      // point to pixel at 1580, 14
  }
}
```

For Macintosh computers, use Mouse.screenSize(width, height, true). This extra parameter tells Teensy to use an alternate algorithm for correct positioning on Apple's OS-X operating system.

X11 on Linux has a known limitation where Mouse.moveTo() does not work. This ugly hack can allow Mouse.moveTo() to work on Linux, by removing Mouse.move(). It also breaks Mouse support on Windows.

Moving The Mouse

To move the mouse, use Mouse.move(X, Y), where X and Y range from -127 to +127. Positive X moves to the right. Positive Y moves downwards. For natural looking motion, many small moves performed slowly are needed.

Here is a simple example that moves the mouse in a triangle.

```
void setup() { } // no setup needed
void loop() {
  int i;
  for (i=0; i<40; i++) {
    Mouse.move(2, -1);
    delay(25);
  }
  for (i=0; i<40; i++) {
    Mouse.move(2, 2);
    delay(25);
  }
  for (i=0; i<40; i++) {
    Mouse.move(-4, -1);
    delay(25);
  }
}
```

Video by Théo Reichel

Clicking

For a simple mouse click, just use Mouse.click(). Use with caution!

```
Mouse.click();
```

For more control over the 3 mouse buttons, you can use Mouse.set_buttons(LEFT, MIDDLE, RIGHT). For each input, 1 means the button is pressed, 0 means not pressed.

```
Mouse.set_buttons(0, 0, 1);
Mouse.set_buttons(0, 0, 0);
```

Click and Drag

You can combine Mouse.move() with Mouse.set_buttons() to perform drag and drop operations.

```
Mouse.set_buttons(1, 0, 0);  // hold left button
Mouse.move(-30, 10);         // move the mouse while holding
delay(500);
Mouse.set_buttons(0, 0, 0);  // release button after move
```

Of course, how to position the mouse and where to move are tricky. You may also need to add substantial delay before the Mouse.set_buttons() call, because many GUI-based programs feature delayed respose, anticipating human-like timing.

Scroll Wheel

You can also control the scroll wheel. Positive numbers scroll upward; negative numbers scroll downward.

```
Mouse.scroll(-3);
```

You can also scroll horizontal and vertical.

```
Mouse.scroll(vertical, horizontal);
```

To scroll only horizontal, just set the vertical change to zero.