

Touch and Immerse

Mouse hack for rapid prototyping and simple input.

An introduction to physical computing

The Mouse

The mouse provides real world physical input or information from our actions into the computer.



By modifying the mouse, we can create a new kind of controller, and a new way of expressing through input.

Conceptualize

What role does touch play in our digital world?

How do we touch or not touch objects, space, each other?

What is your relationship to the physical and non physical world?

How is touch perceived in galleries?

How can we lure/seduce/intrigue a participant into interacting with our objects?

Explore the tactile experience of touch in your piece.

Project

In this project we will be looking at notions of touch and immersion by creating a new controller/interface and our own audio visuals.

Using a mouse as input device, you will create a new interfaced object and visuals in Processing or Scratch as an interactive installation.

There are endless possibilities for recreating the mouse.

- Extend and turn the buttons into flat connectors in door way – the door closing makes the connection.
- Extend and embed the buttons into something new – chair? floor?
- Extend the buttons, exchange the buttons for other on/off type of switches.
- Exchange the potentiometer for another variable resistor.
- Encase the mouse in a new material, new shape, with different qualities and expected use.
- Exploiting the physical requirements for the optical controller to function.

What other expressive or poetic uses are there for the mouse?

We can use the mouse as a simple way to turn off or on audio, or video, or execute some sort of code.

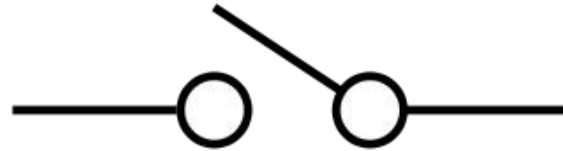
What might you embed a button into?

How might you extend the button?

What other kind of switch might you replace the button with?

Switches

Switches in their basic form are two conductive materials that make an electrical connection.



You can make your own switch easily with two pieces of wire or other conductive material – conductive thread, foil, even lead on paper is conductive.

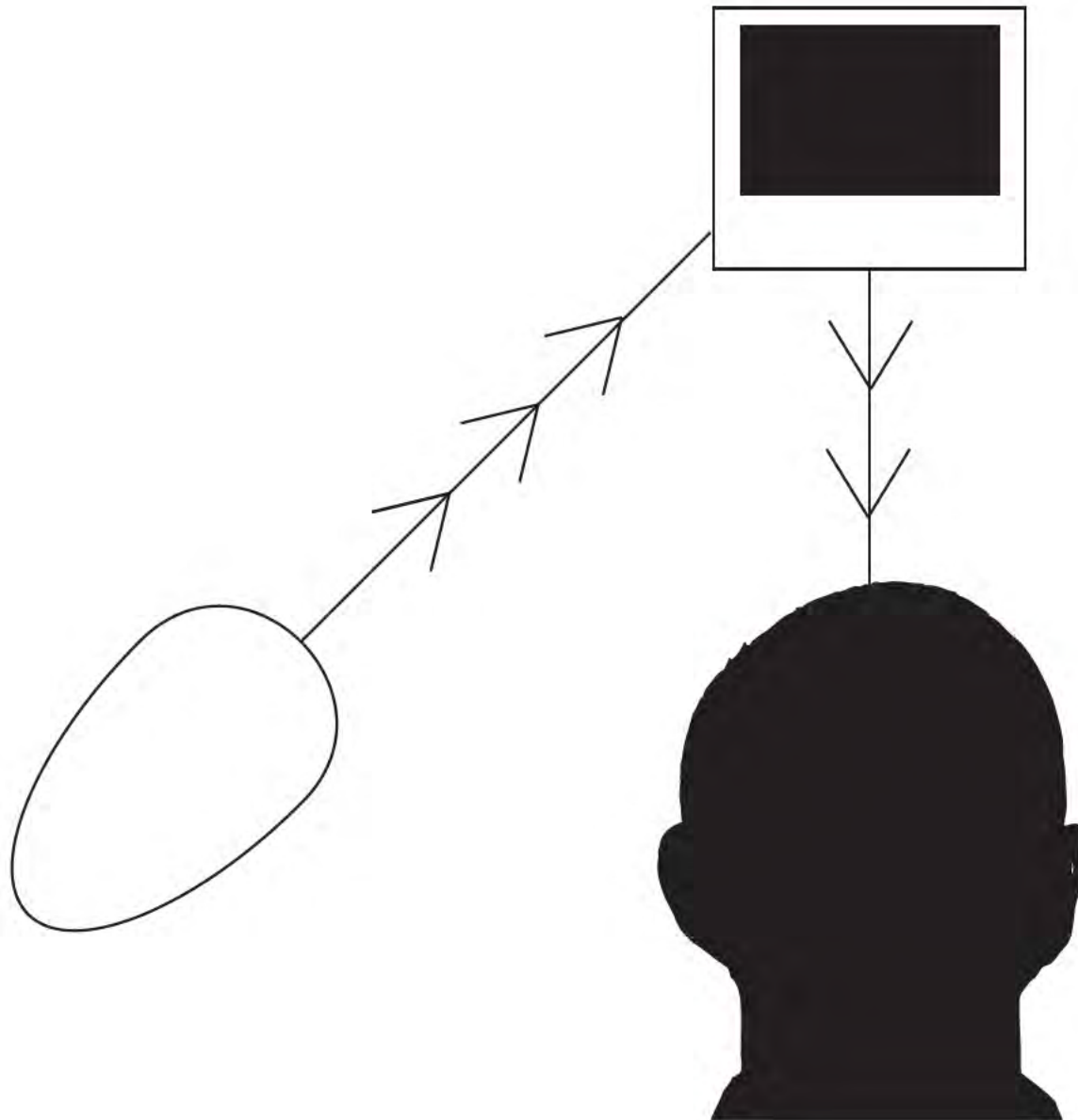
Switches can be automatically open and closed
(eg. by a motor or relay)
or

They can be manually closed by humans –
intentionally or unintentionally

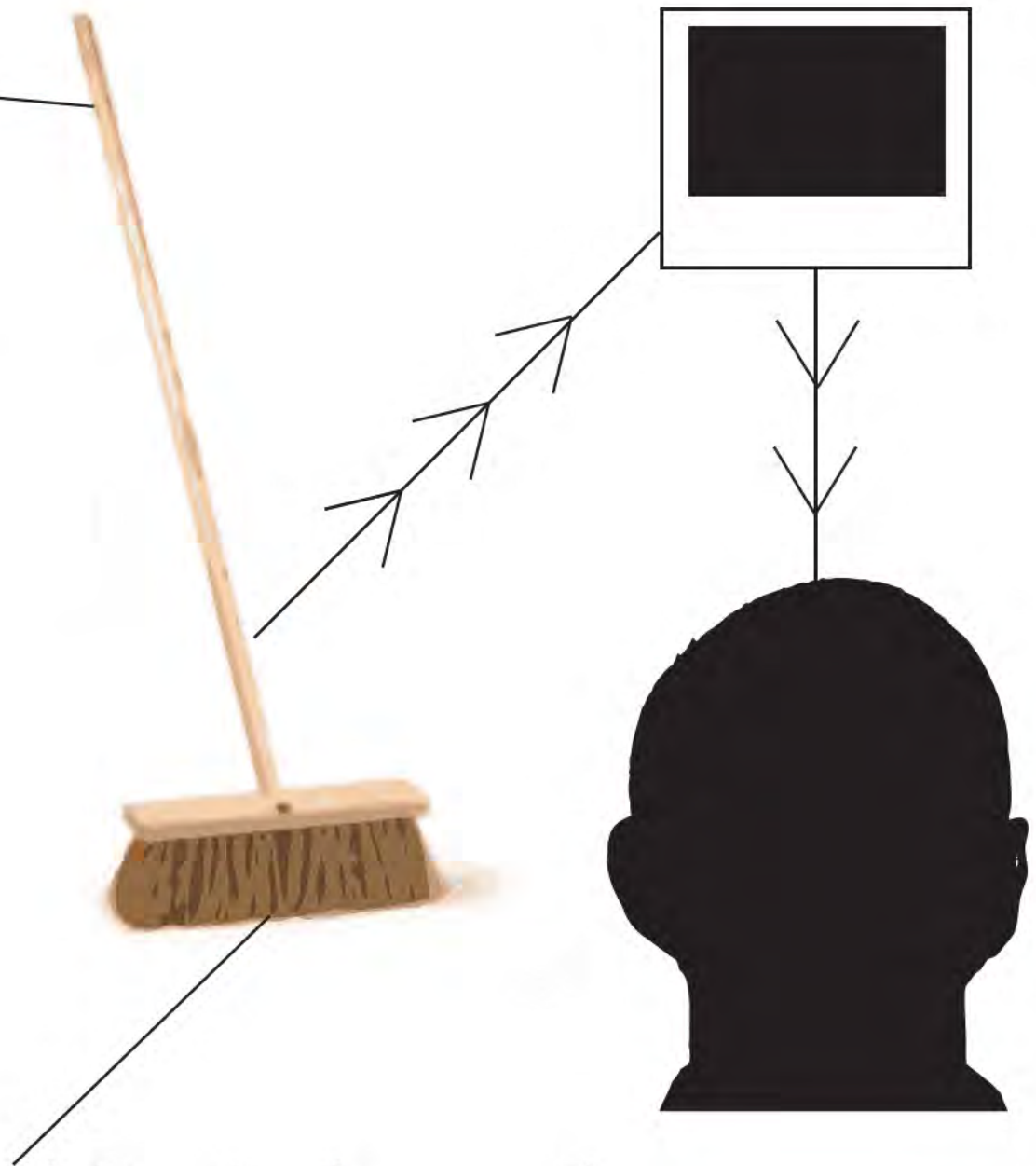


The user controls the mouse. The mouse sends input into the computer through buttons, wheels and an optical lens
The computer receives the information, and executes some commands. The user experiences the output from the computer.

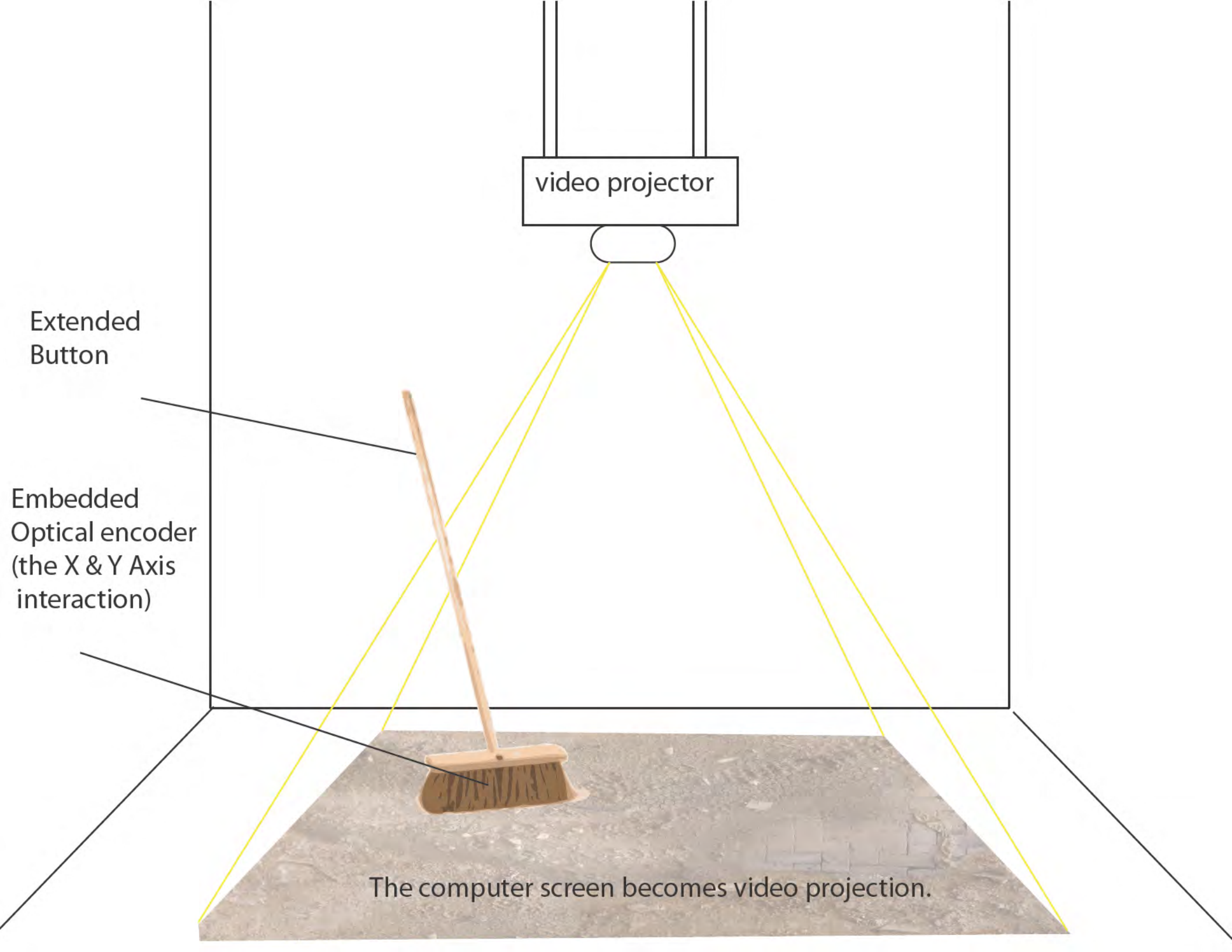
The hacked mouse button can function as a simple trigger for a video, audio, or other event.
We can also re-invent and redesign the mouse interface.



Extended Button



Embedded Optical encoder
(the X & Y Axis interaction)



video projector

Extended Button

Embedded Optical encoder (the X & Y Axis interaction)

The computer screen becomes video projection.

Taking apart the mouse

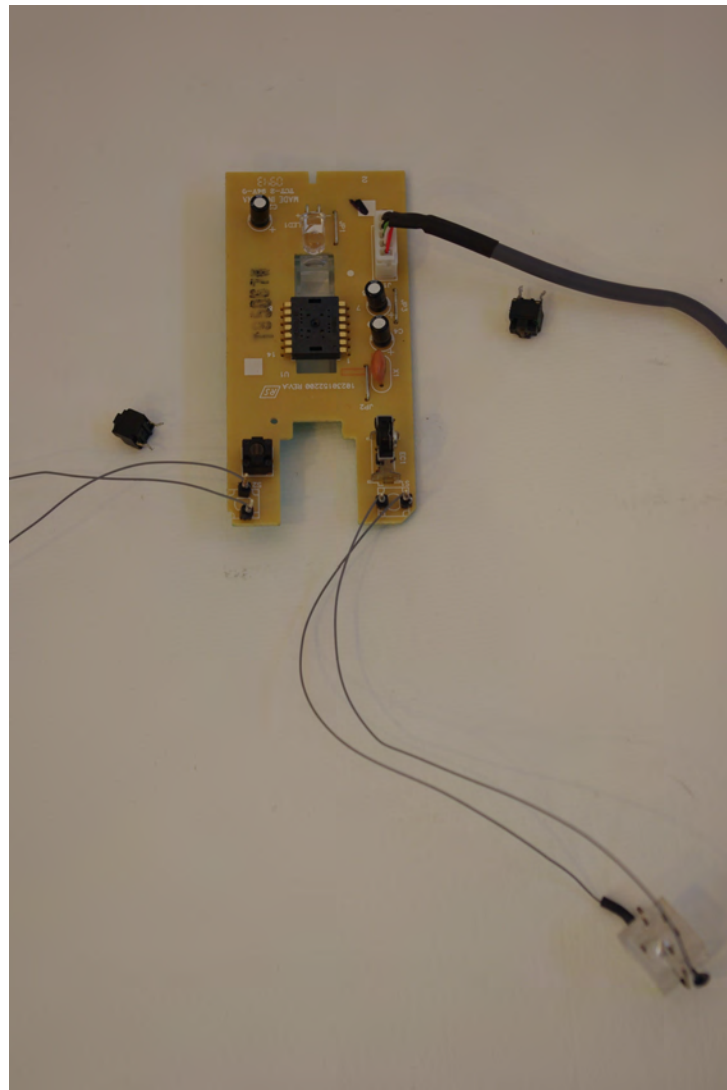
- **Using a screwdriver, unscrew the mouse.**
- **Remove the mouse circuit from the case.**

Next we will look at the circuit.

Note: You may need to remove some rubber covering hiding the screw(s).

Simple button modification

By examining the circuit of the mouse we can find ways to modify the the mouse.
Let's locate the following mouse parts: Optical encoder, lens, buttons,
potentiometer

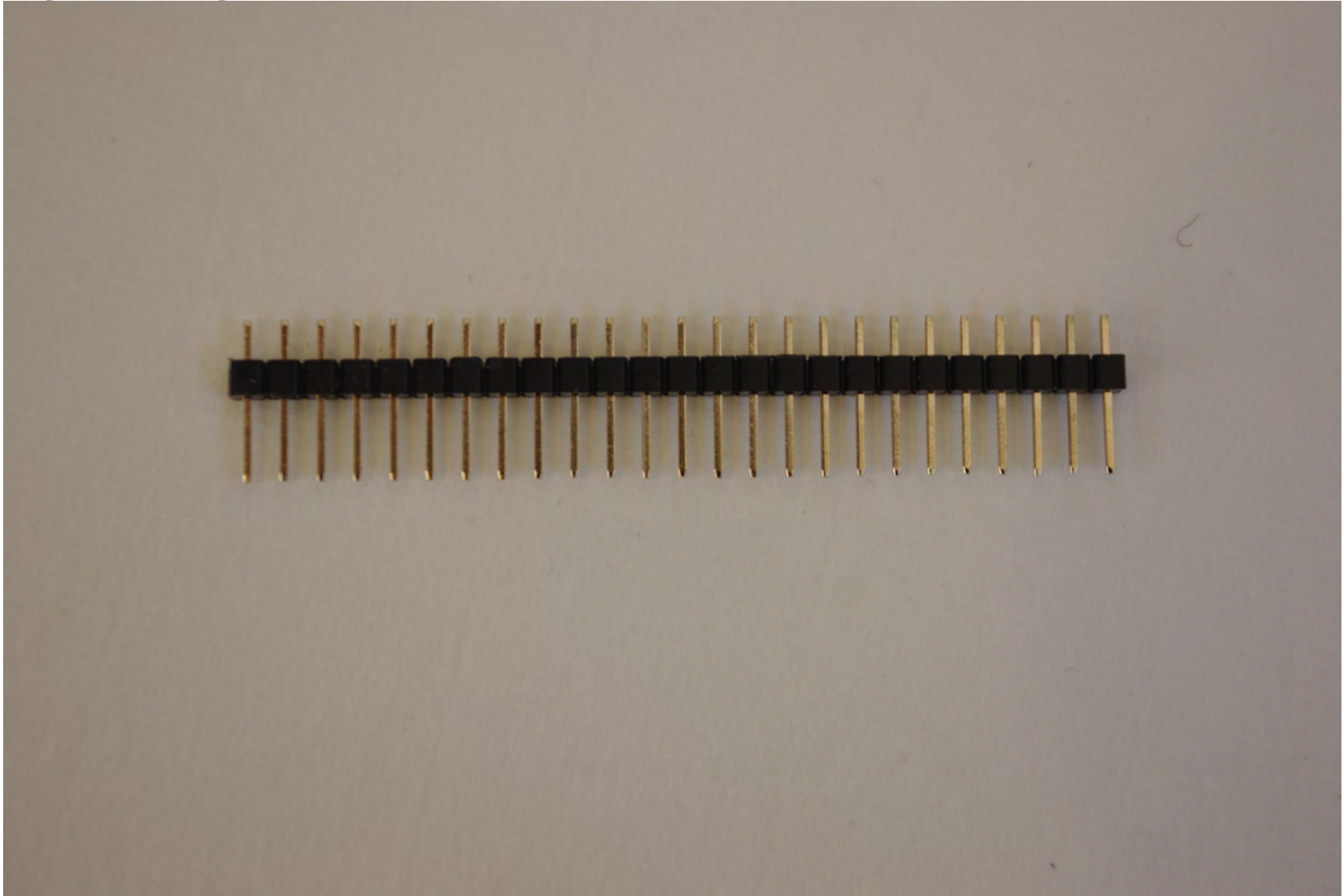


Steps

- Desolder the default button switch with a soldering iron.
- Solder two pin headers to the mouse circuit board.
- Wire up the new switch using wire wrapping wire and a wire wrapping tool
- Solder the wrapped wire.

Note: You can determine the default button by plugging in the open mouse and testing it.

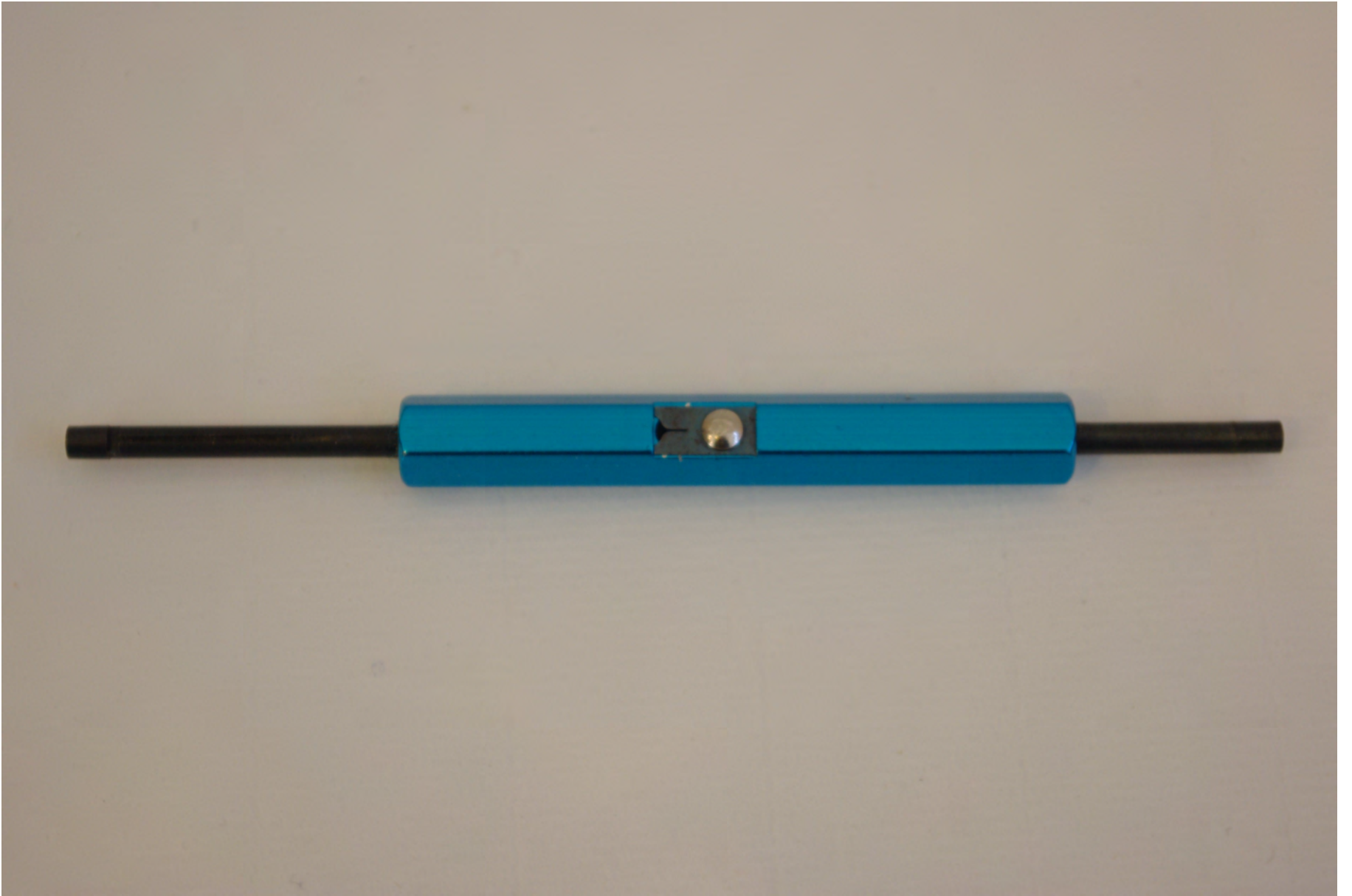
Separate 2 pin headers from the stack.



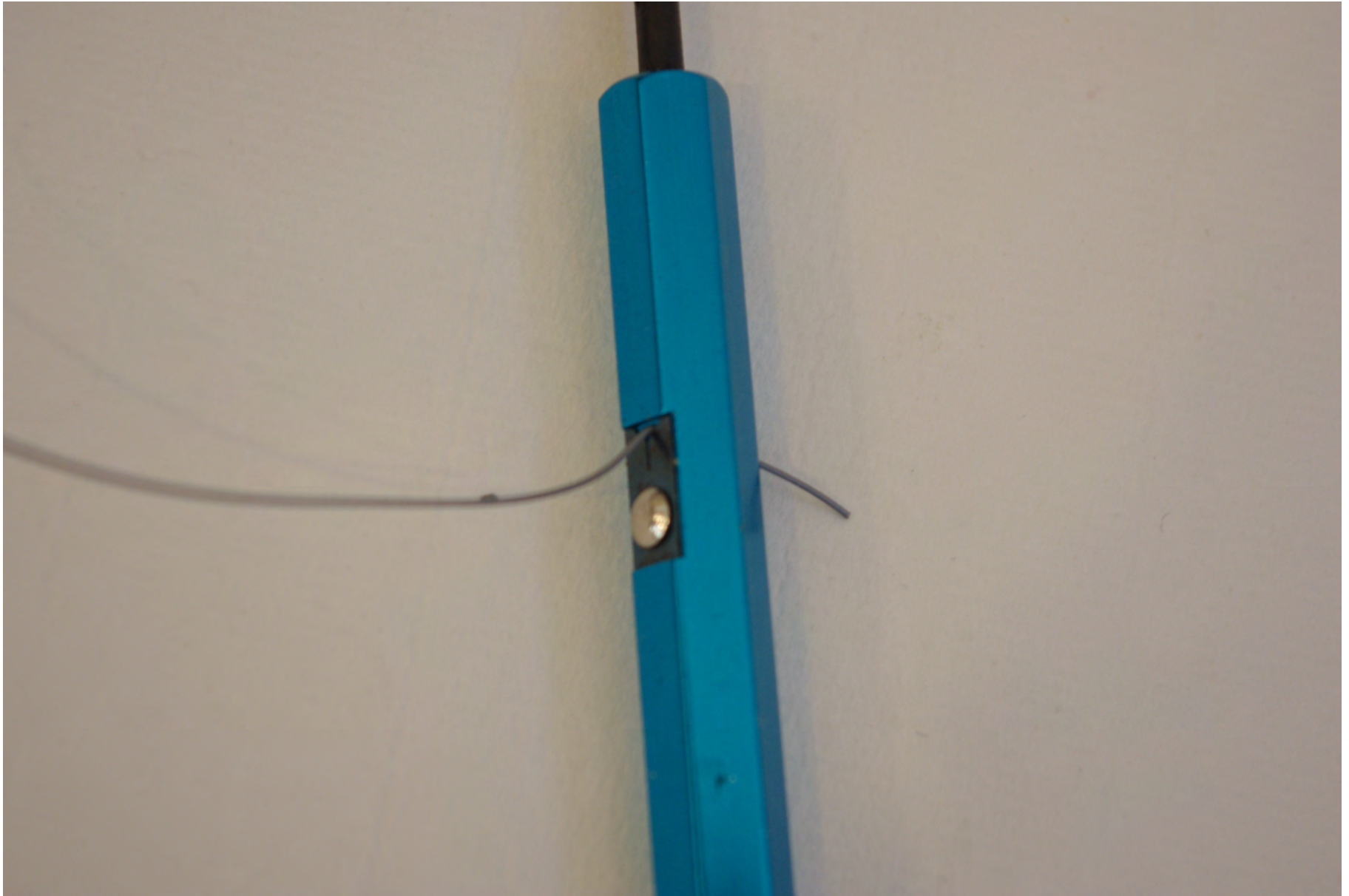
Solder them into the button switch holes on the mouse circuit.



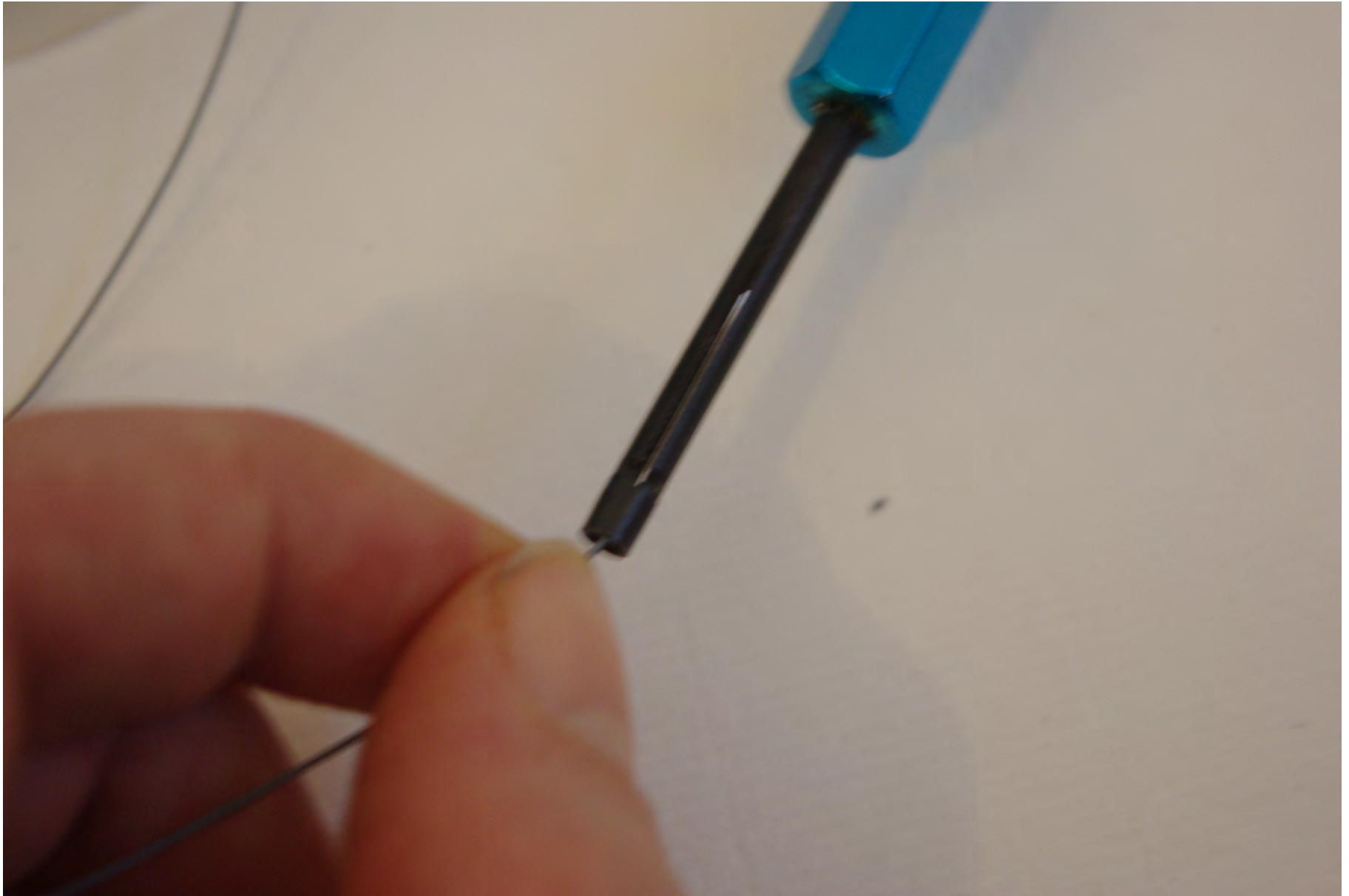
Cut a length of wire.



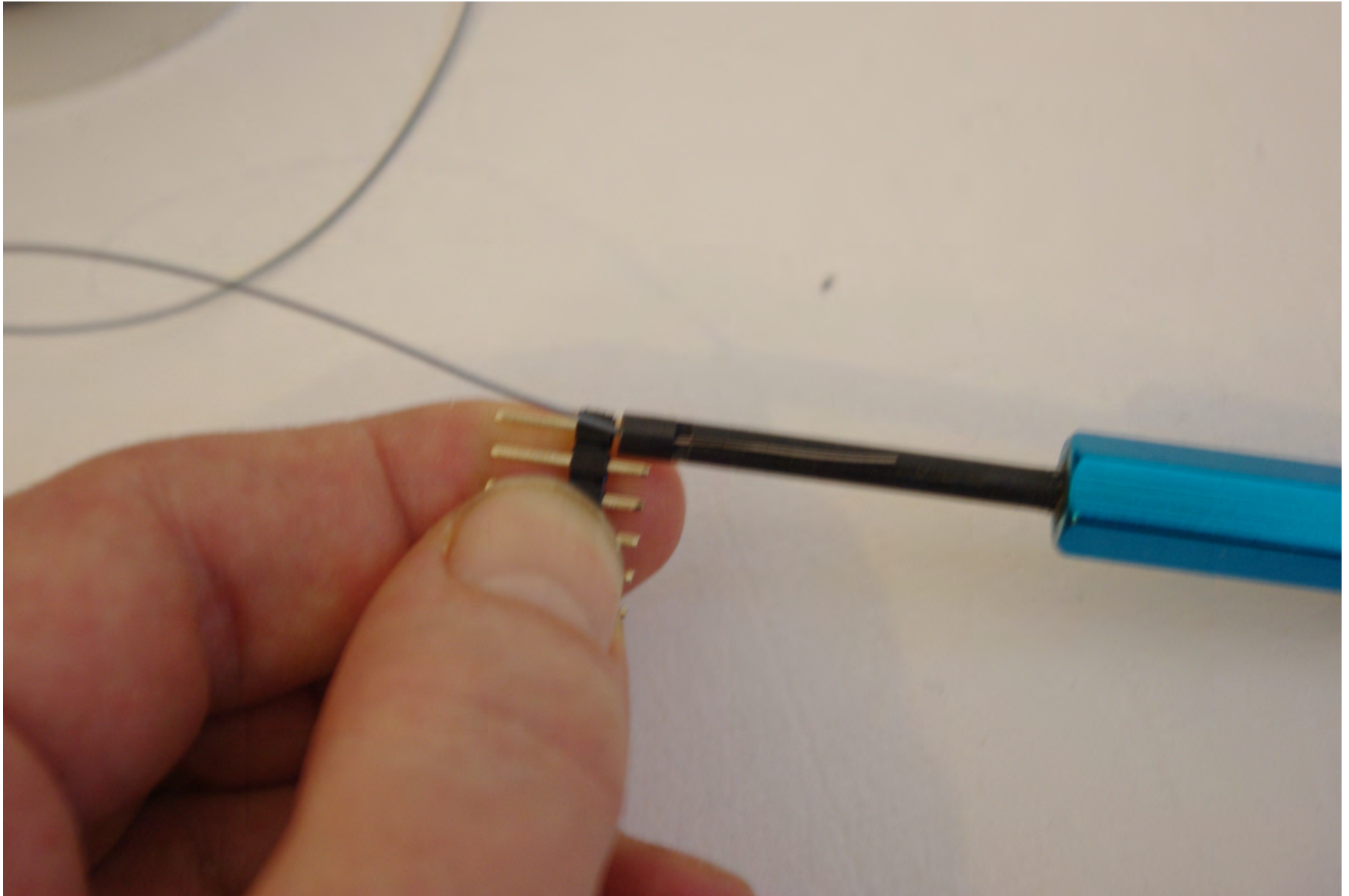
Wire wrapping tool



Strip the wire by inserting the wire into the little slot and pull. Strip about 1 inch to allow for wrapping.

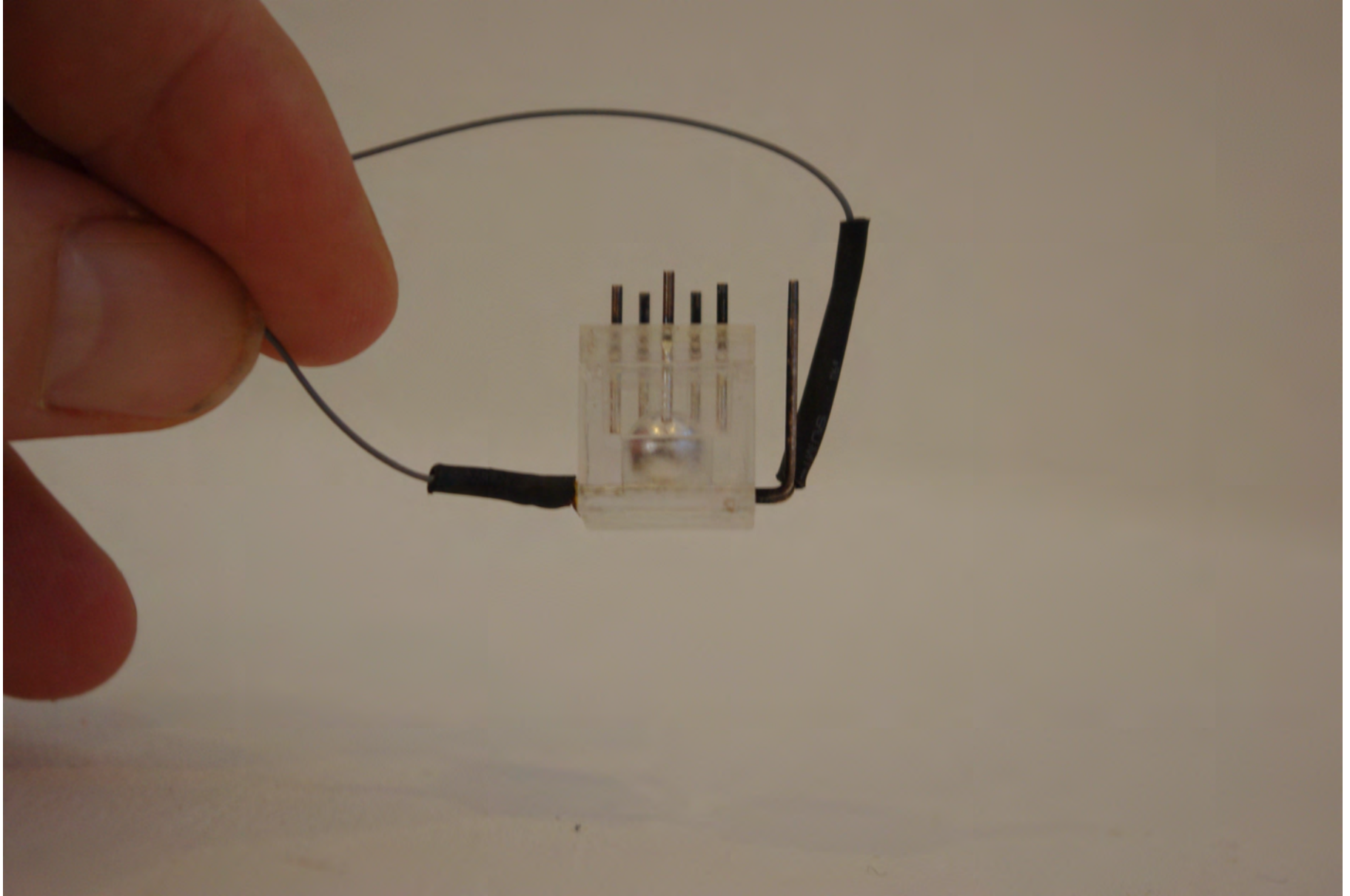


Insert the stripped end into the circle with the slot on the end of the wire wrapping tool.

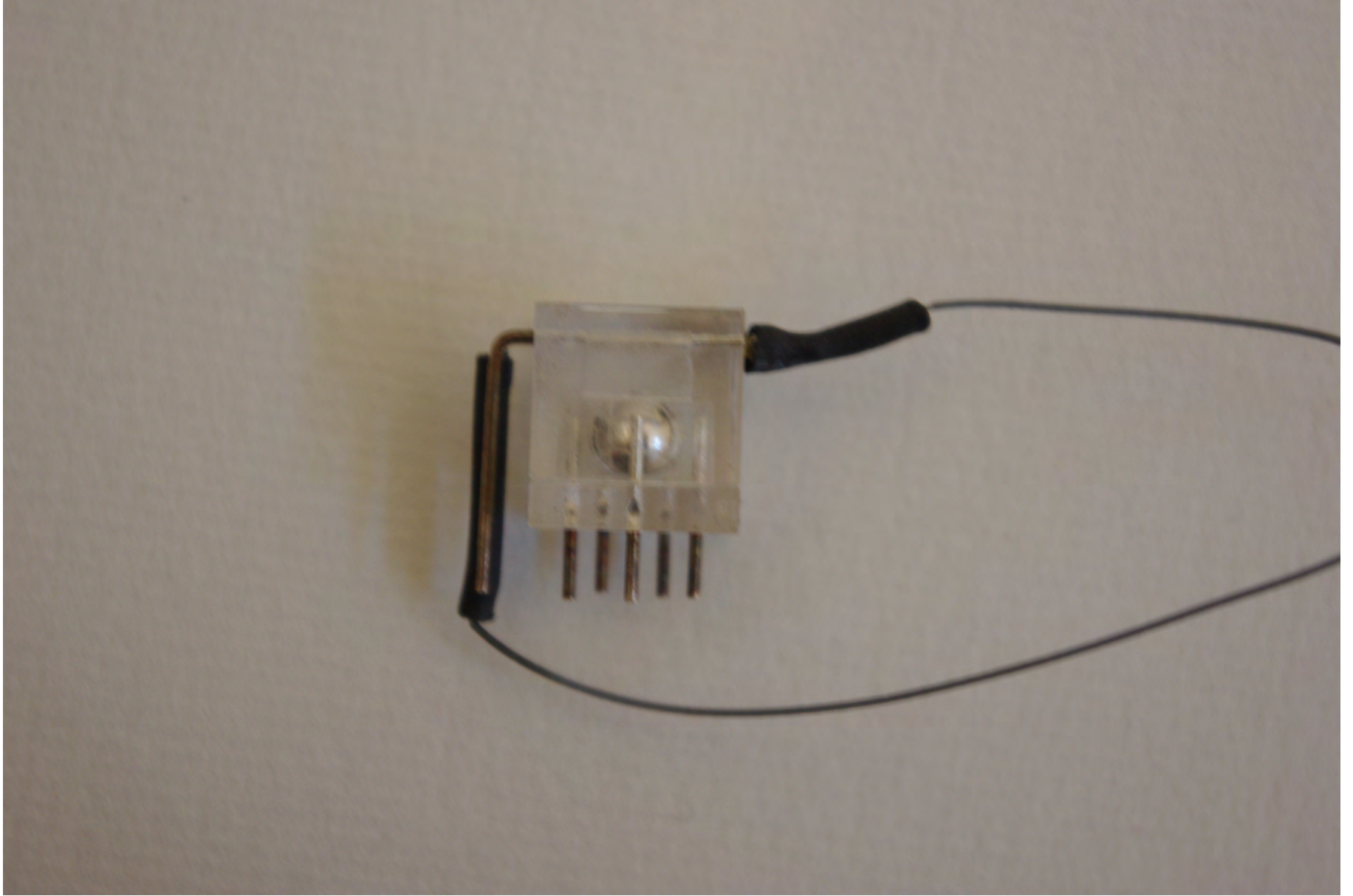


Insert the post of the pin header into the tool and start turning. It should wrap around the end of the pin header.

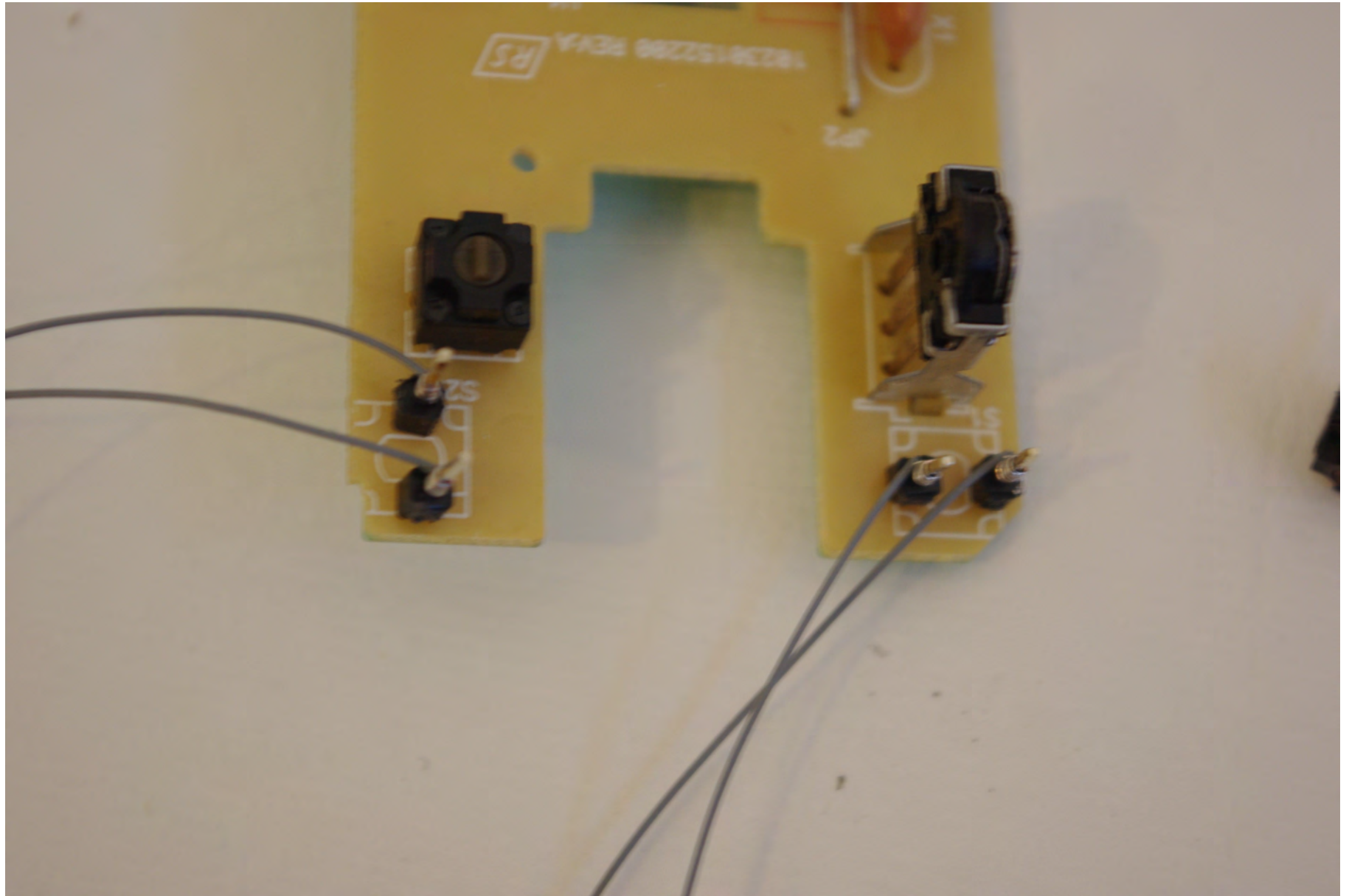
You can use the wire wrapping technique on the legs of the tilt switch.



I have also used heat shrink to make the connections more solid.



The finished circuit



The Code

Scratch

Scratch is a great piece of free software created by MIT Labs for introducing programming.

In scratch, one can create screen-based interaction, simple games rapidly

It's available at scratch.mit.edu

Processing

Processing is an open source and free piece of software currently used by many institutions in new media programs. Processing is available from Processing.org

It is java based coding, and is relatively simple for allowing real world input from an Arduino – which is an open source microcontroller.

For introducing students to real world input for interaction, we will start with a simple mouse hack. The mouse is a built in function in Processing.

**Note: Using an Arduino would be the next logical step for receiving input from sensors, and sending output and are also networkable. FIX THIS wording. Arduino's are open source microcontrollers. For more information on Arduino:
www.arduino.cc**

**Writing some
code in
Processing**

1) Open the processing software

Type in the following highlighted code or load the example from my USB thumb drive.

```
/**  
 * Click.  
 *  
 * Move the mouse to position the shape.  
 * Press the mouse button to invert the colour.  
 */
```

```
void setup() {
```

```
//setup our environment variables. Size of window, background colour
```

```
size(200, 200);
```

```
// this is the size of the window of your project. Try changing it to 640, 480.  
How big do you need to make it in order to fill the entire screen?
```

```
fill(126, 40, 30, 20);
```

```
//fill with the colour assigned the number of 126.The default colour space is  
RGB (Red Geen Blue) with each value ranging from 0 to 255. Try changing  
the numbers and see what happens
```

```
background(255, 204, 0);
```

//make the background this colour. Try changing this colour and see what happens. Use numbers between 0 and 255

```
}
```

```
void draw() {  
  if(mousePressed) {  
    stroke(255);
```

//the outline of the shape will have a black stroke around it

```
  } else {  
    stroke(0);
```

//if the mouse isn't pressed the shape will have a white stroke around it

```
}
```

//if you don't want the cursor showing use this code below

```
if(mousePressed == true) {
```

//this makes the cursor disappear when the mouse is clicked

```
noCursor();
```

```
rect (mouseX, mouseY, 25, 25);
```

//you can replace rect with other shapes. try line, ellipse and play with the secondary numbers as well.

```
} else {
```

```
noCursor();
```

```
ellipse(mouseX, mouseY, 25, 25);
```

//you can replace ellipse with other shapes. try line, quad, rect and play with the secondary numbers. mouseX and mouseY align the drawing with the x and y axis location of the mouse. The next two parameters are width and height. Play with these to experiment with the ellipse

```
}
```

```
}
```

We can also make video interact with the mouse

```
// Learning Processing  
// Daniel Shiffman  
// http://www.learningprocessing.com
```

```
// Example 16-5: Scrubbing forward and backward in movie
```

```
// If mouseX is 0, go to beginning  
// If mouseX is width, go to end  
// And everything else scrub in between
```

```
import processing.video.*;  
//imports the built in video library. You can add other libraries
```

```
Movie movie;
```

```
void setup() {  
  size(640,480);  
  movie = new Movie(this, "face2.mov");  
  //loads the movie that you specify. It must live in  
  //documents/processing/yourprojectfolder/  
}
```

```
void draw() {
```

```
    // Ratio of mouse X over width
```

```
    float ratio = mouseX / (float) width;
```

```
    // The jump() function allows you to jump immediately to a point of time  
    within the video.
```

```
    // duration() returns the total length of the movie in seconds.
```

```
    movie.jump(ratio*movie.duration());
```

```
    // Read frame
```

```
    movie.read();
```

```
    // Display frame
```

```
    image(movie,0,0);
```

```
}
```

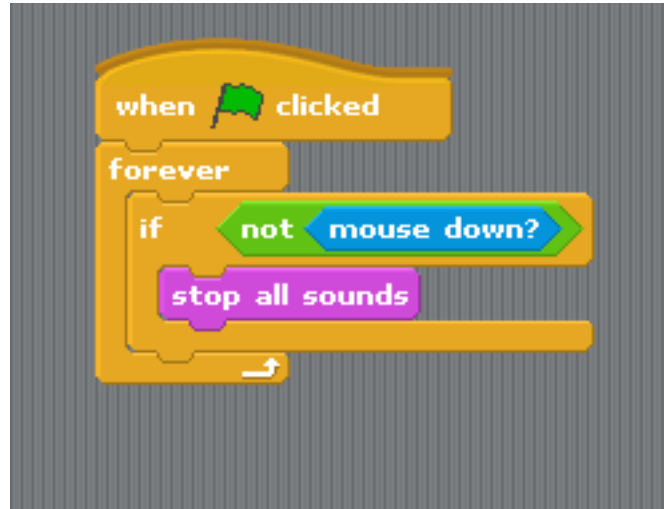
Writing some code in Scratch

Script 1 – Turns audio on with mouse click, makes sprite follow the mouse



Location	Code	Modifier
Control	When green flag clicked	
Control	Forever	
Control	If __ Else	
Sensing	Mouse down?	
Sound	Play note __ for __	60, 0.5
Sound	Play note __ for __	59, 0.5
Sound	Play note __ for __	65, 0.5
Sound	Play note __ for __	
Control	If	
Operator	___ > _____	4
Sensing	distance to __	Mouse pointer
Motion	Point towards	Mouse pointer
Motion	Move __ Steps	10

Script 2 Stops the sound when the mouse click is released.



Location	Code	Modifier
Control	When green flag clicked	
Control	Forever	
Control	If	
Operator	Not	
Sensing	Mouse down	
Sounds	Stop all sounds	

What's the next step?

Once you've mastered the mouse hack, there's keyboard hacking, gamepad hacking for multiple user inputs. After that, pick up an Arduino, and try using different sensor input. By combining Processing and an Arduino, you can control real world items like motors and the screen simultaneously. You can even send email, or other networked data.

