Introducing the iTapStick: Video Game USB Mouse Stick for Wii Nunchuk Controllers

Created by itaparcade

https://learn.adafruit.com/nunchuket-kung-fu-pc-video-game-fighting-stick

Last updated on 2024-06-03 01:41:38 PM EDT

Table of Contents

Overview

Ever wanted to get in on some video game martial arts action and feel like the legendary "Bruce Lee". This 10 minute project will allow you to at least virtally feel like a martial arts expert using Wii Nunchuks (wired and wireless) in combination with a PC game found on Steam known as One Finger Death Punch (OFDP). Using a Pro Trinket and the Nunchucky interface board, you can create a Wii Nunchuk to USB Mouse conversion stick that translates Wii Nunchuk responses into left and right mouse clicks to fight off the agressive Ninjas on the OFDP game. As a bonus, the joystick on the Wii Nunchuk is programmed to be used as a mouse cursor to select different map locations in the game. This allows you to completely play this game with only the Wii Nunchuks. Both wireless and wired Nunchucks can be used with the iTapStick project. You have to build and try this project for yourself to experience the fun you'll have fighting off ninjas using Wii Nunchuks. If you don't believe me, check out the video snippet below that was taken at a Mini Maker Faire using some wired blue third-party Wii Nunchuk controllers. Looks like someone is having fun to me! If you like this project, follow us on Twitter @iTapArcade. In the meantime, let's go ahead and get started with making your own iTapStick!

Prerequisite Guides

I recommend checking out the following tutorials below before starting this project. These tutorials will help you get familiar with the Pro Trinket and help you get setup and configured with the Arduino IDE and libraries.

- Introducing the Pro Trinket (https://adafru.it/nXF)
- Pro Trinket as a USB HID Mouse (https://adafru.it/rd5)

Parts

Components for the iTapStick and tools for building it can be found at Adafruit. These components and tools are featured products on the right sidebar. Search on amazon.com if looking for a wireless Wii nunchuk controller.

- Pro Trinket (http://adafru.it/2000)
- Nunchucky (http://adafru.it/345)
- Wii Controller (wired) (http://adafru.it/342)

iTapArcade is also offering an iTapStick kit for this project with a preprogrammed Pro Trinket. You can purchase this from our Tindie iTapStick Kit Project Site (https://adafru.it/eQ6).

Tools & Supplies

You'll need a soldering iron and some solder to build this project.

- Solder Iron (http://adafru.it/1204) + Solder (http://adafru.it/734)

Learn

Leveraging the Wii Nuncuck Library

Tod E. Kurt (https://adafru.it/eQ7) generated one of the first blog articles that discussed how to send and decode I2C commands to the Wii Nunchuck using an Arduino UNO in February 2008.   He created a nunchuck library as well as produced a wii chuck adapter to easily interface with existing Wii nunchuck controllers (eliminating the need to use extension cable adapters for interfacing).   From his work, numerous makers have extended his library and modified it to deal with both "original" as well as "third-party" Wii nunchuck controllers. Much of his work and others inspired a Wii nunchuck library by Tim Teatro (https://adafru.it/eQ8) discussed in his blog on February 2012.   Other Wii nunchuck libraries might exist, but I plan to use Tim's Wii nunchuck library as a basis for the iTapStick project. To learn more about what has been done with various Wii nunchuck libraries check out the blogs from both Tod and Tim.

Leveraging the Adafruit Pro Trinket Mouse Library

Mike Barela (https://adafru.it/rd6) has a great article on using the Pro Trinket as a USB HID Mouse based on the Adafruit Pro Trinket mouse library. Since the Pro Trinket does not actually have a USB interface chip, he shows how you can leverage the V-USB library to emulate a USB HID Mouse based on this library. Please review the article by Mike for more background information about this library.

The iTapStick Project

A combination of the Wii nunchuck library and the Pro Trinket Mouse library provides the foundation for creating the iTapStick USB Mouse Stick for Wii Nunchuk

Controllers. This project was motivated from playing the OFDP video game found on Steam for the PC. The YouTube video showing the game in action is shown below.

Rather than using the left and right mouse buttons or an XBox controller for playing the game, why not use Wii nunchuck controllers. If we can program the "Z" button to represent a left mouse click for one nunchuck and a right mouse click for the other nunchuck, the fun factor of OFDP dramatically increases. While we are at it, let's use the joystick on the nunchuck to control the mouse position so we can select various areas of the map. With these features combined with two wireless wii nunchucks, the Nunchuket Wii Nunchuck to mouse USB interface stick was born!

Build

Transforming the Adafruit Pro Trinket and nunchucky into a wireless USB mouse stick first begins by soldering the nunchucky to the Pro Trinket. The only soldering required for this entire project involves two steps:

1. Solder the 4 pin header to the Nunchucky
2. Solder the other end of the 4 pin header to the Pro Trinket. Before soldering the other end to the Pro Trinket make sure Pro Trinket pins are connected to the Nunchucky based on Pin A2-->Gnd , Pin A3-->3.3v, Pin A4-->Data, and Pin A5 --> Clk

A picture of the nunchucky soldered to the Pro Trinket is shown below. It is important to make sure the Nunchucky is soldered and oriented correctly with respect to the Pro Trinket.

> You can use a 3.3 volt or 5 volt Pro Trinket for this project, but for the wireless Wii receiver to operate properly only a 5 volt Pro Trinket can be used!
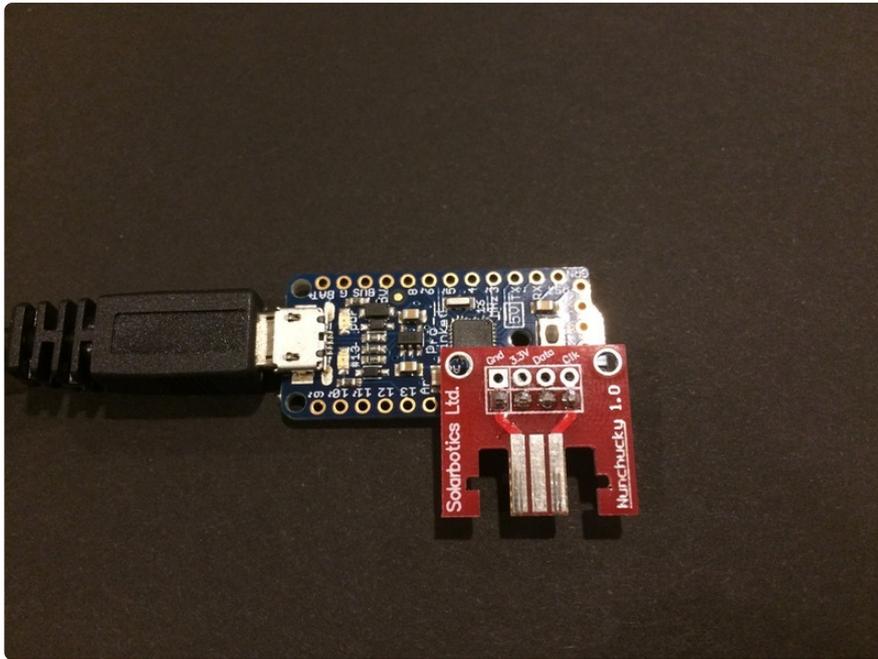
Program

Once adding the Pro Trinket Mouse Library to you Arduino programming environment. Copy the iTapStick project code below and create the iTapStick Sketch in the Arduino IDE.   Since you will need the wiinunchuk.h library, please proceed to download this library and add it to your project as well.

```
/************************************************************************
    iTapStick Project:
    Wii Nunchuk-to-Mouse USB Controller Stick
    Copyright (c) 2015 iTapArcade, LLC

    Date Created: 9 March 2015
    Date Modified: 10 March 2015
    Version: 1.0
    Visit us at iTapArcade.com
    Follow us on Twitter @iTapArcade

    Function: Wired and Wireless mouse emulation (mouse cursor, left click, and
right click)
    from Wii Nunchuk to support Tap and Mouse control based Games for PC, Mac, and
Android based games

    Support our open source projects by getting an iTapStick Kit Available:
    https://www.tindie.com/products/itaparcade/itapstick-wii-nunchuck-to-mouse-usb-
stick/?pt=ac_prod_search

    Microcontroller: Pro Trinket (3 or 5 volt) from Adafruit.com
    Note: Need 5 volt Pro Trinket for Wireless Wii Nunchuks!

    Wii Nunchuck Library Code from:
    Author: Tim Teatro
    Date  : Feb 2012
```

```
      http://www.timteatro.net/2012/02/10/a-library-for-using-the-wii-nunchuk-in-
arduino-sketches/
 ***********************/

#include &lt;Wire.h&gt;
#include &lt;EEPROM.h&gt;
#include "wiinunchuk.h"  // Wii Nunchuk Library
#include &lt;ProTrinketMouse.h&gt;    // Pro Trinket V-USB mouse emulator

int led = 13;    // Use LED on Pro Trinket for Status and Button Press Indicator
int loop_cnt=0;
int mode = 0;      // Defines mode of the iTapStick: left mouse click is C or Z
button

// parameters for reading the joystick:
int range = 40;                  // output range of X or Y movement
int threshold = range/10;      // resting threshold
int center = range/2;         // resting position value

void setup() {
  pinMode(led, OUTPUT);

  nunchuk_setpowerpins();    // set power pins for Wii Nunchuk
  nunchuk_init();            // initilization for the Wii Nunchuk
  delay(250);

  while(!nunchuk_get_data()){ // loop until Wii Nunchuk is connected to Nunchucky
  nunchuk_init();
  delay(250);
  nunchuk_get_data();
  digitalWrite(led, HIGH);
  delay(250);
  digitalWrite(led, LOW);
  delay(500);
  }

  digitalWrite(led, HIGH);  // let user know they can select mode within (5 seconds)
  delay(5000);

 // check if user is holding any of buttons to change mode

  nunchuk_init();
  delay(250);              // Once connected lets get additional data to determine if
C or Z
  nunchuk_get_data();  // button is being held down for programming mode
  delay(10);
  nunchuk_get_data();
  delay(10);
  nunchuk_get_data();

  int leftState = nunchuk_zbutton();
  int rightState = nunchuk_cbutton();

    if(leftState){  // save mode for future time if Z is defined as left mouse click
      mode = 0;
     EEPROM.write(0, 0);
     digitalWrite(led, LOW);
     delay(250);
     digitalWrite(led, HIGH);
     delay(250);
     digitalWrite(led, LOW);
     delay(250);
     digitalWrite(led, HIGH);
     delay(250);
     digitalWrite(led, LOW);
     delay(250);
     digitalWrite(led, HIGH);
     delay(3000);
```

```
    }else if(rightState){ // save mode for future time if C is defined as left
mouse click
      mode = 1;
      EEPROM.write(0, 1);
      digitalWrite(led, LOW);
      delay(100);
      digitalWrite(led, HIGH);
      delay(100);
      digitalWrite(led, LOW);
      delay(100);
      digitalWrite(led, HIGH);
      delay(100);
      digitalWrite(led, LOW);
      delay(100);
      digitalWrite(led, HIGH);
      delay(100);
      digitalWrite(led, LOW);
      delay(100);
      digitalWrite(led, HIGH);
      delay(100);
      digitalWrite(led, LOW);
      delay(100);
      digitalWrite(led, HIGH);
      delay(100);
      digitalWrite(led, LOW);
      delay(100);
      digitalWrite(led, HIGH);
      delay(3000);
    }else
    {
    mode = EEPROM.read(0);   // no mode selection load in saved mode
    }

  digitalWrite(led, LOW);  // get LED indicator ready to show when buttons are
being pressed

 // Start Pro Trinket into Mouse mode
  TrinketMouse.begin();                 // Initialize mouse library
}

void loop() {
  if( loop_cnt &gt; 10 ) { // every 10 msecs get new data
    loop_cnt = 0;


    if(nunchuk_get_data())   // only check for data if data is available from Wii
Nunchuk
    {

    // right and left click control
    int leftState = nunchuk_zbutton();
    int rightState = nunchuk_cbutton();

    if (leftState) // if button is pressed update
    {
      if(mode==0){
      TrinketMouse.move(0,0,0,MOUSEBTN_LEFT_MASK);
      }else
      {
      TrinketMouse.move(0,0,0,MOUSEBTN_RIGHT_MASK);
      }
      digitalWrite(led, HIGH); // show button is pressed by LED
     }else if (rightState) // if button is pressed update
    {
      if(mode==0){
      TrinketMouse.move(0,0,0,MOUSEBTN_RIGHT_MASK);
      }else
      {
      TrinketMouse.move(0,0,0,MOUSEBTN_LEFT_MASK);
```

```
    }
     digitalWrite(led, HIGH);// show button is pressed by LED
    }
    else
    {
    TrinketMouse.move(0,0,0,0); // if no button is pressed update
     digitalWrite(led, LOW); // show no button is pressed by LED
    }

    int xReading = nunchuk_joy_x();    // read the x axis
    xReading = map(xReading, 38, 232, 0, range); // map accordingly
    int xDistance = xReading - center;
    if (abs(xDistance) &lt; threshold) {
      xDistance = 0;
    }


    int yReading = nunchuk_joy_y();   // read the y axis
    yReading = map(yReading, 38, 232, 0, range); // map accordingly
    int yDistance = yReading - center;
    if (abs(yDistance) &lt; threshold) {
      yDistance = 0;
    }

    if ((xDistance != 0) || (yDistance != 0)) { // move the mouse based on x and y
    TrinketMouse.move(xDistance, -yDistance, 0, 0);
    }
    }
  }
  loop_cnt++;
  delay(1);
}
```

We have built in two modes of operation for using the iTapStick. One mode allows the right click and left click to correspond to the "Z" and "C" respectively. The other mode allows the right click and left click to correspond to the "C" and "Z" respectively. You can set the mode when initially connecting the Pro Trinket to the computer. Once setting the mode, it will remember the next time so you don't have to set it every time.

Play

Onced program, the iTapStick can be used with both wired and wireless Wii nunchuk controllers. An OFDP setup on a PC tablet with two iTapSticks is shown in the video below. We have two wireless Kama Wii Nunchuks interfaced to our iTapSticks. You can always use the wired wii nunchuk controllers found at adafruit (http://adafru.it/342)as well if you are not able to find wireless ones. Let the OFDP fun begin!