



# USB Mouse Control

by Sean King, November 2010



## Abstract

Discover how easy it is to use Flowcode to turn an ECIO module into a custom USB interface compatible with the default Windows HID (human interface device) driver. In this example, a very short Flowcode program can allow the ECIO to function as a PC mouse.

## Requirements

### Software:

- Professional or Educational version of Flowcode V4 for PICmicros

### Hardware:

- ECIO28P or ECIO40P module
- A few discrete potentiometers and switches (Matrix Protostation is used for convenience)

## Introduction

USB is now the only way that most peripheral devices can communicate with a PC. The flexibility and performance of USB requires a complex system that can be difficult to understand and implement.

To make USB applications easier to implement, Flowcode (currently only the PIC version) includes components that hide most of the complexities of the USB protocol, and provide the programmer with a more user-friendly interface. The components can be used with the increasing number of USB enabled microcontrollers to develop intelligent peripheral devices.

The diverse types of devices that communicate over USB have a range of requirements, depending on the characteristics of the data being transferred. The different implementations of USB are usually handled by the driver software supplied with some hardware. Some of the more common drivers have been integrated into Windows and are available for general use.

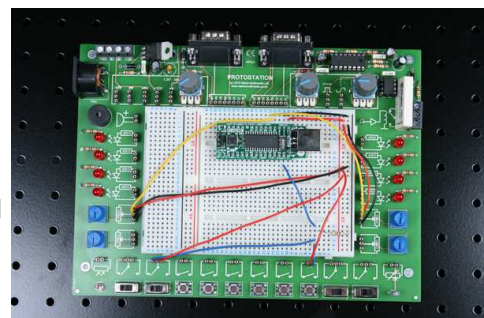
One of the default drivers is referred to as HID (Human Interface Device) which supports basic functions required by a range of input and output devices (mouse, keyboard, game controllers, some display devices, etc.)

The programs linked to this article use the Flowcode USB HID component to allow an ECIO28P to appear to be a mouse when connected to a PC. In both cases the ECIO and additional components are powered by the PC via the USB connection.

## Hardware setup

The picture to the right shows the ECIO mounted on a Matrix Multimedia Protostation and connected to some of the additional components supplied.

When the connection is established, the mouse movement and button information sent to the PC is under the control of any



**ECIO connected to Protostation peripheral components**

Flowcode program running on the ECIO. This opens up the possibility of creating interface devices from new combinations of sensors and signal sources.

Control of the mouse simply involves using the USBHID component SendData macro to send a 3-byte array to the PC.

The first byte contains the button information (bit 0 is the left-button; bit 1 is the right-button; bit 2 is the centre-button, etc.).

The second and third bytes represent the horizontal and vertical movement speed respectively. Values between +8 and -8 give a good speed range on the screen.

To handle the negative numbers, calculations should be carried out using integer variables.

The completed USB control system is shown here on the new Matrix Multimedia Electronic Workstation, allowing further development work to be carried out using the range of test equipment included.



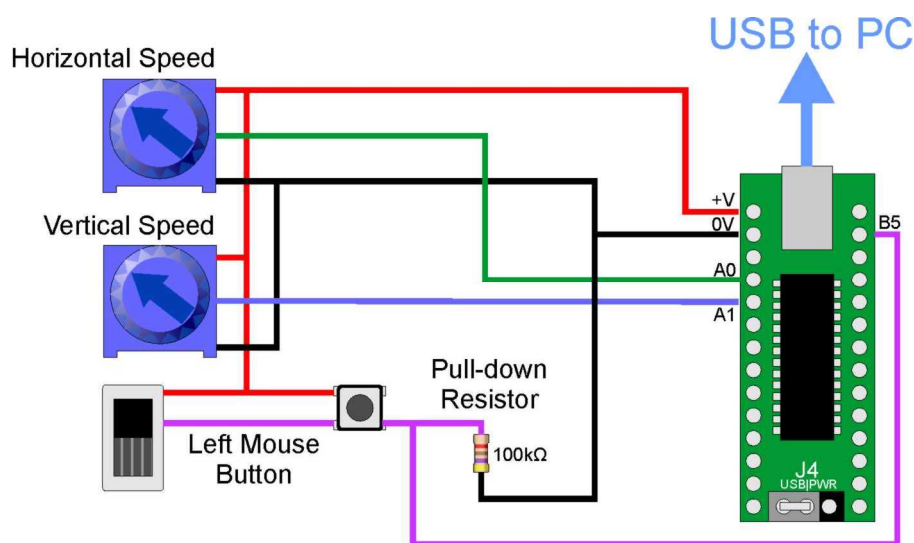
**ECIO, Protostation and Electronic Workstation**

## The programs

There are two example programs associated with this article. The first is the USB\_mouse program which takes over the horizontal movement of the screen cursor, causing it to move under the control of the light level being detected by an LDR (Light Dependent Resistor) connected to one of the ADC input channels.

This program moves the cursor at a constant speed and uses a simple time based calculation to convert the mouse velocity values into screen positions.

The USB\_Sketch program uses two analogue inputs to control the horizontal and vertical movement speed. The input potentiometer values (0 to 255) have 128 subtracted from them to set the centre position.

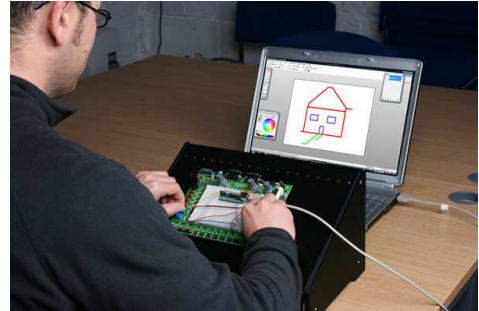


**USB "Etch-a-Sketch" hardware connections**

tion as 0. The results are then divided by 16 to give the required range.

In addition, a toggle switch and a push-button are connected to an input that controls the mouse left-button value. This allows the left-button to be either pressed momentarily, or switched on indefinitely.

When this program is combined with a graphics package (like Paint) it allows new drawing control methods to be developed. Lines can be drawn by switching the left-button control switch on and controlling the cursor using the potentiometers (or other sensor devices).



**The USB "Etch-a-sketch" in action**

**Remember - the cursor speed is being controlled in this program, not its position!**

On the right you can see an example of what can be achieved!

## Taking this further

One obvious improvement would be to allow the potentiometer positions to relate to the mouse position rather than velocity — this would certainly make controlling the PC's cursor a lot easier!

Once basic control of the mouse is mastered, a range of other sensor inputs can be used to control the mouse cursor. This has numerous practical applications, for example allowing alternative PC input mechanisms for computer users who find using the mouse difficult.

HID is used for a whole range of input devices, not just the mouse. The same basic principles used in this article apply to joysticks and keyboards too, which opens up a lot of other interesting and practical possibilities.

## Further reading

Below are some links to other resources and articles on related subjects, and technical documentation relating to the hardware used for this project...

Flowcode: <http://www.matrixmultimedia.com/flowcode.php>  
ECIO: <http://www.matrixmultimedia.com/ecio.php>  
Protostation: <http://www.matrixmultimedia.com/electronicworkstation.php>

Learning Centre: [http://www.matrixmultimedia.com/lc\\_index.php](http://www.matrixmultimedia.com/lc_index.php)  
User Forums: <http://www.matrixmultimedia.com/mmforums>  
Product Support: [http://www.matrixmultimedia.com/sup\\_menu.php](http://www.matrixmultimedia.com/sup_menu.php)

Copyright © Matrix Multimedia Limited 2011

Flowcode, E-blocks, ECIO, MIAC and Locktronics are trademarks of Matrix Multimedia Limited.  
PIC and PICmicro are registered trademarks of Arizona Microchip Inc.  
AVR, ATmega and ATtiny are registered trademarks of the ATMEL corporation.  
ARM is a registered trademark of ARM Ltd.