

## Re: bike computer to PC interface

---

*Source:* <http://sci.tech-archive.net/Archive/sci.electronics.basics/2007-03/msg00329.html>

---

- *From:* [shb\\*NO\\*SPAM\\*@comporium.net](mailto:shb*NO*SPAM*@comporium.net) (Si Ballenger)
  - *Date:* Fri, 09 Mar 2007 00:41:03 GMT
- 

On 7 Mar 2007 09:06:32 -0800, pete\_tomson@xxxxxxxxxxxx wrote:

Hi All – I found this newsgroup and was wondering if anyone could help me.

I'd like to connect a magnetic sensor (bicycle computer) directly to my computer through one of the ports without a power supply.

Basically the reed switch will have a current induced in it so I'm thinking I'd ideally like the circuit to be passive with minimal external circuitry.

I have a pic of a basic circuit I found but don't know how it works and I don't know how to post it. Basically it has a connection to pin 4 through a 4.7kohm resistor a direct connection to pin 6 and a ground through pin 20.

If you tell me how to post a pic I can show the diagram.

Can anyone make suggestions about circuit design – can I feed the wires directly into my parallel/serial port?

I'm also trying to write a program in VB6 to read the input, but not having much luck.

I'd really appreciate any help anyone could give me.

Thanks

Pete

I assume you are wanting to make a simple revolution counter for a stationary bike. I made a simple setup by connecting pin 7 (rts) to a resistor (4.7k sounds good) thru a wheel contact setup to pin 2 (rx). When the com port is opened, pin 7 goes high supplying ~+10v. When the wheel contact was made, jibberish is put in the serial port input buffer. I made a simple program using Just Basic (free) that checks the input buffer for input every 150 ms. If there is input to the buffer the wheel has made

## Re: bike computer to PC interface

a revolution. The wheel contact was an index card stuck into the spokes such that I bowed out a little bit. I taped some aluminum foil on the card surface so that it would brush two bare wire ends from the circuit when it went by to make the circuit. This put the jibberish in the buffer. Below is a simple program I made to test it to see if it would work. It assumes that one wheel revolution is ~.00138762 miles.

```
open "Com1:9600,n,8,1,ds0,cs0,rs" for random as #comm
```

```
x = 0
```

```
y = 0
```

```
[repeat]
```

```
timer 150, [delay]
```

```
wait
```

```
[delay]
```

```
dataRead$ = input$(#comm, lof(#comm))
```

```
if dataRead$ = "" goto [repeat]
```

```
y = y+.00138762
```

```
print y
```

```
'print x
```

```
x=x+1
```

```
if x=1000 goto [quit]
```

```
goto [repeat]
```

```
[quit]
```

```
Close #comm
```

```
End
```

.