WIRELESS OPTICAL USB MOUSE

Reza Afshari
Final Project Description
Etec 471, Professor Todd Morton
December 8, 2004

*Western Washington University*
*Electronics Engineering Technology*
FUNCTIONAL DESCRIPTION

Introduction

Almost everybody who uses a computer uses a mouse. The traditional, corded, mechanical mouse is the most widely used. However, in some situations more freedom is required than a corded mouse can offer. Situations such as a presentation using a projector or just an over-cluttered workspace are appropriate for a more versatile mouse. I propose to build a wireless optical mouse with a receiver that plugs right into the Universal Serial Bus (USB) port of a computer. This design will be an improvement over the traditional mouse by giving the user the freedom to use their mouse unbound to their computer. In addition, optical sensing will result in more precise control of the cursor on the computer screen and the added versatility of being able to use the mouse on many different surfaces.

Hardware Description

![Block Diagram]

Figure 1 - Block Diagram
Figure 1 shows the hardware block diagram of the wireless mouse system. The system will consist of a wireless optical mouse and a USB receiver. The user will move the mouse in the direction they desire the cursor on the computer screen to move. The mouse movement information, along with the user input from the right and left buttons, is converted into a radio signal and sent over free air to the receiver. The receiver decodes the information and sends it to the computer through the USB port.

The wireless mouse product sketch is shown in Figure 2. The maximum dimensions of the mouse enclosure are 4.75 in x 2.75 in x 1.6 in. The maximum dimensions of the receiver are 5 in x 4 in x 2 in.

![Figure 2 - Maximum Dimensions](image)

**Detailed Functional Description**

The functional hardware block diagram of the wireless mouse transmitter is shown in Figure 3. The mouse will be controlled by the 9S12C32 microcontroller. The mouse will also consist of Radio Frequency (RF) transmission circuitry, an optical motion sensor, and
two push buttons, which will serve as the left and right mouse input buttons. The mouse will be powered by two AAA, 1.2-1.5 volt batteries.

The optical sensor detects the movement of the mouse by taking thousands of tiny images per second of the surface, which is illuminated by a red LED. The Digital Signal Processor (DSP) on board the IC compares image after image from the sensor and determines magnitude and direction of the mouse’s movement. The data of the mouse’s movement is output on the sensor’s serial port.

The microcontroller will process and encode the data produced by the optical sensor and the activity of the two push buttons, so that it can be transmitted. The microcontroller will also perform power management tasks for the sensor. The XTAL circuitry, which runs off of a 16MHz crystal, will control the timing of the system.

The RF circuitry will create a 916.48 MHz ISM (Industry, Scientific, Medical) RF signal from the encoded data using Frequency shift keying (FSK) modulation. The signal is amplified and transmitted over free air to the receiver.
The functional hardware block diagram of the wireless mouse receiver is shown in figure 4. The receiver will consist of the 9S12DP256B (Star 12) microcontroller, RF reception circuitry, and a USB interface adapter.

The RF receiver will be implemented by an integrated circuit. The receiver will receive the signal sent from the mouse transmitter and demodulate it. The demodulated data is amplified and sent to the microcontroller.

The 9S12DP256B microcontroller will decode the data. The decoded data is converted into mouse data format using the Microsoft Mouse Protocol, which is compatible with all major operating systems. The data packets, each three bytes long, are sent to the host computer via the Universal Serial Bus (USB) port. The USB adapter is used as the interface between the 9S12DP256B microcontroller and the computer’s USB port. The microcontroller will be powered by the 5 volts provided by the USB port. The XTAL circuitry, which runs off of a 16MHz crystal, will control the timing of the system.

![Figure 4 - Mouse Receiver Hardware Block Diagram](image-url)
Software Requirements

The software for the wireless mouse system will be written in the C programming language. I chose to write the code in C because it is easier to produce organized code that is easy to read and portable.

In the transmitter, the MicroC/OS-II preemptive kernel will be used to handle all the tasks and scheduling on board the 9S12C32 microcontroller. The software will need to get the mouse displacement as well as the buttons status, encode this information for transmission, and perform power management to extend battery life.

In the receiver, the MicroC/OS-II preemptive kernel will also be used to handle tasks and scheduling on the star12. The software on the star12 will need to get the information being received from the mouse and convert it to Microsoft mouse protocol before it is sent to the computer.

The following modules will be in the wireless mouse transmitter software (on the 9S12C32 microcontroller):

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>KERNEL</td>
<td>This module will incorporate MicroC/OS-II for managing tasks.</td>
</tr>
<tr>
<td>MOVEMENT</td>
<td>This module will get the movement data from the sensor.</td>
</tr>
<tr>
<td>BUTTONS</td>
<td>This module will monitor the activity of the left and right buttons.</td>
</tr>
<tr>
<td>ENCODE_RF</td>
<td>This module will encode the mouse movement and button status into RF data packets.</td>
</tr>
<tr>
<td>POWER</td>
<td>This module will put the Optical sensor in different levels of lower power during times of no detected movement of the mouse for certain amounts of time.</td>
</tr>
</tbody>
</table>
The following modules will be in the wireless mouse receiver software (on the 9S12DP256B microcontroller):

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>KERNEL</td>
<td>This module will incorporate MicroC/OS-II for managing tasks.</td>
</tr>
<tr>
<td>DECODE_RF</td>
<td>This module will capture and decode the incoming RF data packets containing the movement and buttons status from the mouse.</td>
</tr>
<tr>
<td>MOUSE</td>
<td>This module will convert the mouse information into a format that the host computer will recognize as mouse input (Microsoft mouse protocol).</td>
</tr>
</tbody>
</table>

**USER INTERFACE**

The user interface for the wireless mouse (shown in figure 5) is quite simple. With the mouse placed on a mostly flat surface, the user will physically move the mouse in the direction that they desire the cursor on their computer terminal to move. The left and right buttons will perform the operation of left and right clicking as would a typical mouse.

![Figure 5 - User Interface](image-url)
The user is required to plug the receiver into the computer’s USB port. The computer will automatically recognize it as a standard two-button mouse.

**DEVELOPMENT PLAN**

Until this point, project development has focused on defining the project and gathering important parts. Once the project definition is complete, I will study the “Embedded Microcontrollers, by Todd Morton” text in order to prepare to write the project software using the C programming language. Once winter quarter begins, I will develop the hardware for the transmitter and receiver. Purchase of the microcontrollers will be done through Professor Todd Morton. The remaining parts will be obtained online or through the parts room in ET340. There are no potential problems with delivery of parts.

Project development will take place primarily in the Electronics Engineering Technology laboratory, ET340. The hardware in the lab that I will use to develop my project includes a digital multimeter, mixed signal oscilloscope, a programmable power supply, and a soldering iron. The software will be developed using Codewright and Noral BDM debugger POD.

The project will be simple to demonstrate. All that is needed is a computer terminal with an available USB port. The Prototype of the mouse will be enclosed in an ergonomic case, probably the case from an existing mouse. The circuitry will most likely be on a printed circuit board because of the limited space inside the mouse enclosure. The receiver will consist of the star12 development board, and the limited external circuitry wired on a breadboard. If time permits, the receiver will be enclosed in a case as well.
Weekly Schedule

Winter break:

Week 1: Order and gather all critical parts.
Week 2: Study “Embedded Microcontrollers”.
Week 3: Study “Embedded Microcontrollers”.

Winter quarter:

Week 1: Assemble and test the optical sensor setup.
Week 2: Study the Optical sensor signal.
Week 3: Build the RF transmitter circuitry.
Week 4: Study the mouse to computer communication.
Week 5: Work on encoding/decoding RF data.
Week 6: Work on encoding/decoding RF data
Week 7: Work on encoding/decoding RF data.
Week 8: Develop power management software.
Week 9: Develop software.
Week 10: Develop software.
Week 11: Develop software.

Spring quarter:

Week 1: Develop software.
Week 2: Develop software/Hardware review.
Week 3: Develop software.
Week 4: Develop software.
Week 5: Develop power management software.
Week 6: Interface receiver with the USB adapter.
Week 7: Enclose mouse in a case.

Week 8: Left open for troubleshooting

Week 9: Code Review.

Week 10: Perform final check on the project

Week 11: Demonstrate the project.

ELECTRICAL SPECIFICATIONS

Project Specifications

- USB 1.1 and 2.0 compliant.
- 916.48 MHz RF link, FCC designation: ISM (Industry, Scientific, Medical).
- Operating range: up to 2 meters.

Mouse Specifications

- Power supply: 2 x AAA batteries
- Sensor frame rate: 2300 frames per second
- Accurate up to 14 inches per second of movement.
- Resolution: 400 counts per inch.
- Maximum weight: 8 ounces.

Power Requirements

- Mouse transmitter: Two AAA batteries.
  Total worst case power dissipation: 109.5 mA
  Estimated worst-case battery life: 6.84 hours
- Mouse receiver: 5 Volts from the USB bus.
  Total worst case power dissipation: 96.5 mA
Special Environmental Requirements

Operating temperature range: 0 ° - 40 ° Celsius

Printed Circuit Board Size Limits

Mouse: 4”x2”x0.5”

Receiver: 4”x3”x1”

PRELIMINARY PARTS LIST

Transmitter

<table>
<thead>
<tr>
<th>Part number</th>
<th>Part</th>
<th>Source</th>
<th>Qty.</th>
<th>Price</th>
<th>Power Dissipation</th>
<th>Lead Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADNS-2051</td>
<td>Optical Sensor</td>
<td>Arrow</td>
<td>1</td>
<td>$3.70</td>
<td>25 mA</td>
<td>0 weeks</td>
</tr>
<tr>
<td>HDNS-2100</td>
<td>Optical Lens</td>
<td>Arrow</td>
<td>1</td>
<td>$0.30</td>
<td>N/A</td>
<td>0 weeks</td>
</tr>
<tr>
<td>HLMP-ED80</td>
<td>Red LED</td>
<td>Arrow</td>
<td>1</td>
<td>$0.20</td>
<td>42 mA</td>
<td>0 weeks</td>
</tr>
<tr>
<td>HDNS-2200</td>
<td>LED Clip</td>
<td>Arrow</td>
<td>1</td>
<td>$0.20</td>
<td>N/A</td>
<td>0 weeks</td>
</tr>
<tr>
<td>N/A</td>
<td>0.1uF cap</td>
<td>digikey</td>
<td>6</td>
<td>$0.168</td>
<td>N/A</td>
<td>2 weeks</td>
</tr>
<tr>
<td>N/A</td>
<td>2.2uF cap</td>
<td>digikey</td>
<td>1</td>
<td>$0.08</td>
<td>N/A</td>
<td>2 weeks</td>
</tr>
<tr>
<td>N/A</td>
<td>4.7uF cap</td>
<td>digikey</td>
<td>1</td>
<td>$0.08</td>
<td>N/A</td>
<td>2 weeks</td>
</tr>
<tr>
<td>CSALS12M0X55</td>
<td>Ceramic resonator</td>
<td>digikey</td>
<td>1</td>
<td>$0.26</td>
<td>N/A</td>
<td>2 weeks</td>
</tr>
<tr>
<td>N/A</td>
<td>15K resistor(LED)</td>
<td>digikey</td>
<td>1</td>
<td>$0.056</td>
<td>N/A</td>
<td>2 weeks</td>
</tr>
<tr>
<td>9S12C32</td>
<td>microcontroller</td>
<td>digikey</td>
<td>1</td>
<td>$11.28</td>
<td>35 mA</td>
<td>2 weeks</td>
</tr>
<tr>
<td>MAX682</td>
<td>Charge pump</td>
<td>Maxim</td>
<td>1</td>
<td>N/A</td>
<td>N/A</td>
<td>2 weeks</td>
</tr>
<tr>
<td>N/A</td>
<td>Push button</td>
<td>N/A</td>
<td>2</td>
<td>N/A</td>
<td>N/A</td>
<td>0 weeks</td>
</tr>
<tr>
<td>TXM-916-ES</td>
<td>RF transmitter</td>
<td>digikey</td>
<td>1</td>
<td>$9.49</td>
<td>7.5 mA</td>
<td>2 weeks</td>
</tr>
</tbody>
</table>

Receiver

<table>
<thead>
<tr>
<th>Part Number</th>
<th>Part description</th>
<th>Source</th>
<th>Qty.</th>
<th>Price</th>
<th>Power dissipation</th>
<th>Lead Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>9S12DP256B</td>
<td>microcontroller</td>
<td>digikey</td>
<td>1</td>
<td>$20.10</td>
<td>65 mA</td>
<td>2 weeks</td>
</tr>
<tr>
<td>DLP-USB245M</td>
<td>USB adapter</td>
<td>Mouser</td>
<td>1</td>
<td>$25.00</td>
<td>25 mA</td>
<td>2 weeks</td>
</tr>
<tr>
<td>RXM-916-ES</td>
<td>RF receiver</td>
<td>digikey</td>
<td>1</td>
<td>$12.80</td>
<td>6.5 mA</td>
<td>2 weeks</td>
</tr>
<tr>
<td>N/A</td>
<td>0.1uF cap</td>
<td>digikey</td>
<td>4</td>
<td>$0.336</td>
<td>N/A</td>
<td>2 weeks</td>
</tr>
<tr>
<td>N/A</td>
<td>USB A-B male</td>
<td>digikey</td>
<td>1</td>
<td>$2.19</td>
<td>N/A</td>
<td>2 weeks</td>
</tr>
</tbody>
</table>