Wireless Hand Sensor

Rotational Tracking to Control a Computer Mouse

Mushfiqur Sarker, Jason Muhlestein and Anton Bilbaeno

Abstract

A low-cost wearable device was developed that tracks the rotational movement of an individual's hand using commonly available inertial navigation sensors fused with Bluetooth transmitting capabilities. The device accurately emulates a computer mouse without the need for a flat surface. An inertial measurement unit (IMU) derives the rotational pitch and yaw movement of an individual's hand. Touch sensors positioned throughout the device gives the person the ability to perform clicking actions. In conjunction, the user is able to control their computer via natural hand movements reducing medical symptoms such as tendonitis, cramping and carpal tunnel.

Table of Contents

List of Illustrations

List of Equations

List of Tables

Introduction

Background

Douglas Engelbart of Oregon State University developed the first computer mouse in 1964. Since its invention, the mouse has been an integral part of the computer. Its usefulness shows no signs of diminishing. Alternate forms of the mouse have been created in the past, adding additional buttons or scroll wheels, but no variations have been successful in providing an intuitive, natural method of control at a low cost. Over prolonged use of a standard mouse, the user's hand will experience stress. In the 21st century, majority of jobs require the use of a computer but proper computer usage techniques are not practiced. By allowing natural movements to control the computer, the Wireless Hand Sensor (WHS) has the potential to reduce stress on an individual's hand.

Solution

The WHS allows a person to wirelessly control a computer via a glove fitted with a printed circuit board (PCB). This project was developed to interface with a computer from afar and without the need of a flat table-like surface. The traditional mouse requires a flat surface where the optical sensors can depict changes to determine mouse movement on screen. The WHS can be used from a distance of 25 feet where the hand is suspended in a floating environment, removing excessive stress on the wrist.

Applications

Any scenario where a standard computer mouse is required could also be controlled by the WHS. For example, individuals giving presentations to a group can use the WHS to control a slideshow without the need to physically be present in front of a computer. Additionally, one can type on a keyboard without the need to remove the device from their hand. This gives the user simultaneous usage of the keyboard and mouse. Further applications in different sectors include:

1) *Rotational Hand Tracking for Medical Research:* The ability to track how the hand moves through space is crucial to mitigating medical injuries such as tendonitis, carpal tunnel, and cramping. Researchers would be able to use the WHS to collect data points of regular rotational hand movements compared to irregular movements as found on patients with wrist related injuries.

2) *Robotics Control:* The WHS provides researchers in the field of robotics the ability to map the trajectory of hand movements to control remote devices. This interface could be utilized in both research and production assembly applications needing precise remote control.

3) *Gaming:* The integration of this system into current and future computer gaming applications is viable. The uniqueness of the WHS's ability to emulate the mouse provides first-person shooter and flight simulator games control via natural human input.

System Level Architecture

The diagram in Figure 1 depicts the system level architecture of the WHS. The main components of this project were the microcontroller, IMU, touch sensors, and feedback. A user interface is present on the parent computer allowing the individual to change the action executed by each touch sensor. Examples of actions that may be performed include mouse clicks, shortcuts such as copy and paste, or device operations such as sensitivity adjustment.

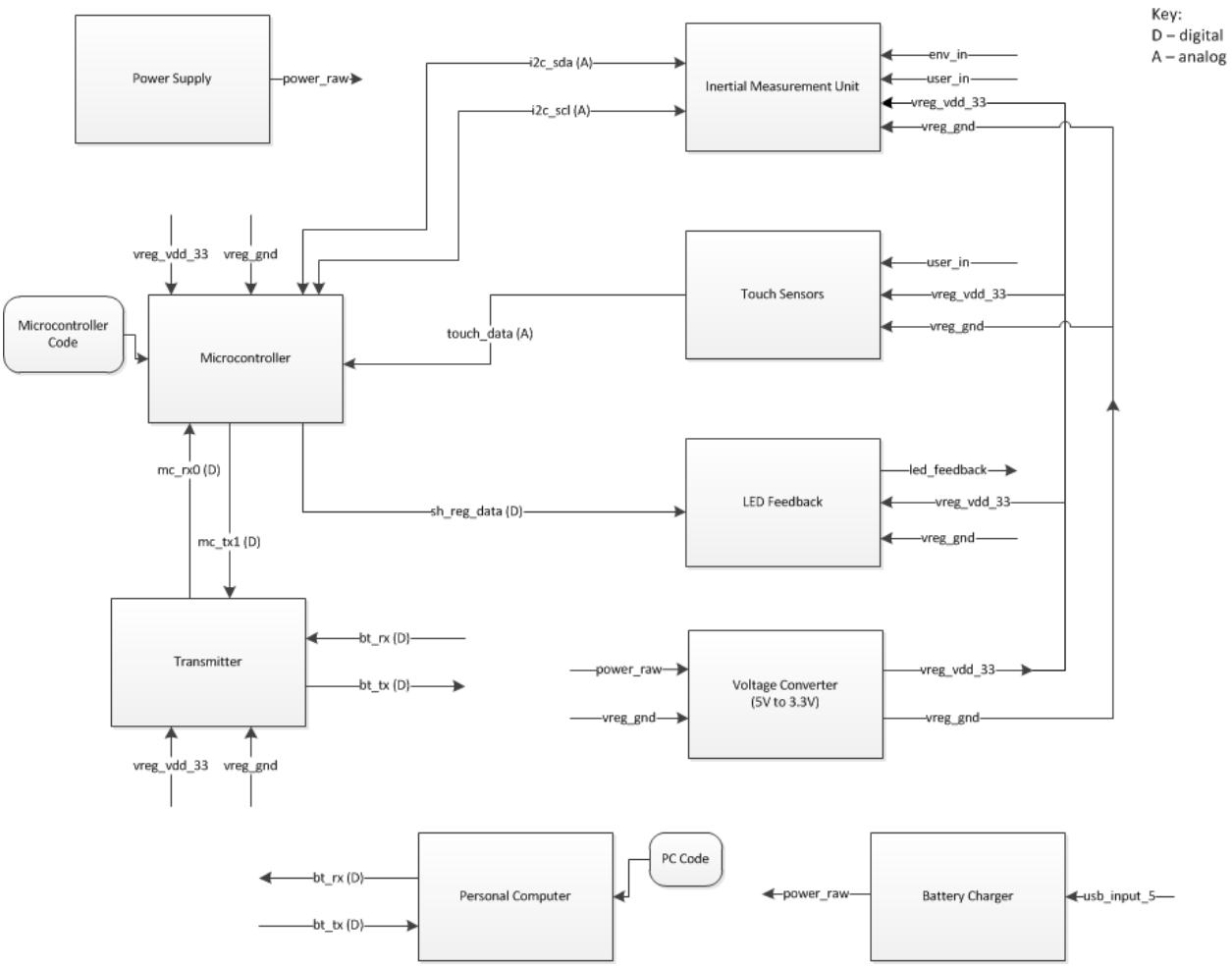


Figure 1 - WHS system level diagram

System Description

Hardware

Glove Design

The glove design required multidisciplinary principles to attach robust touch sensing technologies. After developing various versions with flexible, capacitive, and pressure sensors, conductive fabric was chosen. Conductive fabric is traditional fabric impregnated with metallic conductors, effectively modeled as a sheet of copper [1]. The advantage to this design was the ability to sew fabric onto the glove, providing a durable touch interface. The fabric used in the WHS has a 1 Ω/in^2 of resistance, modeled as a resistor [1]. Using the MSP430F2254's 10-bit ADC peripheral partnered with internal pull-up resistors, a voltage divider is implemented to

sense touches on the different finger sensors. The thumb is equipped with an external pull-up resistor featured on the circuit board pulled high to VCC. When the thumb touches any one of the eight finger sensors shown in Figure 2, an action is performed, for example, left mouse click. Simplicity in a design can be the key to success. With this innovative touch sensing principle the WHS was highly intuitive. Figure 2 displays the bottom side of the glove.

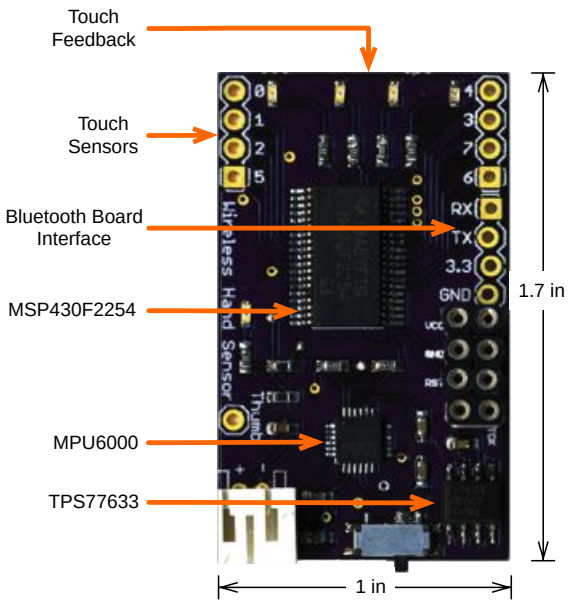
Figure 2 - Conductive fabric on glove fingertips

The main PCB board was placed on top of the glove with all sensors being routed internally with conductive thread. Conductive thread is similar to the fabric but has higher resistance per inch and allowed the routing to me more flexible than wire [2]. The final result is shown in Figure 3. The glove was designed to be robust, but comfortable so users could operate the WHS for hours at a time.

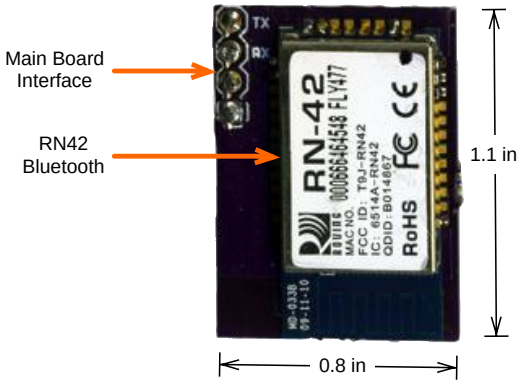
Figure 3 - WHS top side

Printed Circuit Board Design

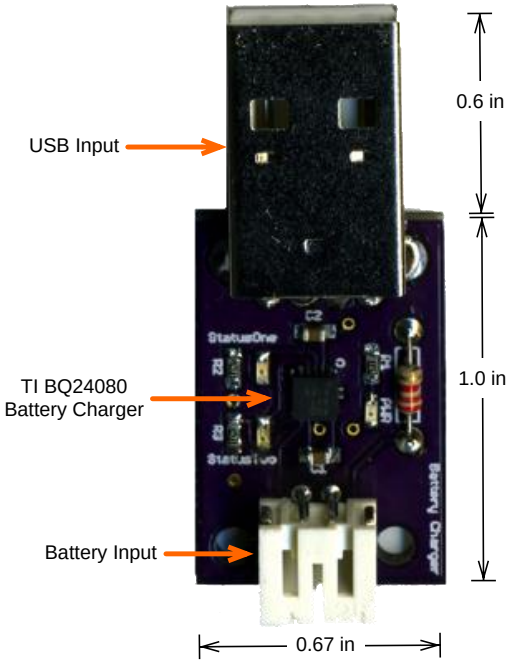
The WHS features two individual boards that attach to the glove, the main board and Bluetooth board. The main board is 1.0in x 1.7in (2.54 cm x 4.32 cm). To reduce the total footprint, the Bluetooth board connects directly on top of the main board. In low volume prototype production, the total cost of the WHS is less than 50 USD. Considering that a typical wireless mouse in 2012 can retail from 30 USD and higher, the WHS has a competitive price point.



1) Main board with IMU: The main board contains the MSP430F2254 microcontroller, IMU, TPS77633 voltage regulator, touch feedback, interface to touch sensors and the Bluetooth board. The IMU is an MPU6000, which is an integrated accelerometer and gyroscope by Invensense [3]. The benefit of an integrated IMU is a major reduction in drift. Individual accelerometers and gyroscopes provide axis misalignment issues, which cannot be removed without knowing the misalignment angles. The MPU6000 has internal alignment, which aids in the reduction of inaccuracies in measurements. A 3.7 V lithium-ion battery coupled with a low-dropout regulator powers the WHS. The TPS77633 voltage regulator provides a supply of 3.3 V to the IMU and MSP430F2254.



2) *Bluetooth board:* The WHS Bluetooth board features an RN42 Bluetooth module from Roving Networks [4]. This Bluetooth module is robust with RF shielding to reduce interference with other frequencies. The MSP430F2254 is able to interface with this device via UART transmission capabilities. The module draws a current of 30mA when transmitting data to the computer [4]. This board directly connects on top of the main board to reduce footprint size.



3) *Battery charger board:* The WHS battery charger board features a BQ24080 Li-ion battery charger IC [5]. The board can charge a single-cell lithium ion battery rated at 3.7 volts [6]. The USB input allows it to be plugged into a standard USB port on a computer. The current provided to the board is limited to keep the USB port safe from overcurrent situations. Li-ion batteries have various stages in a charge cycle. Therefore, two status LEDs are present to show the current status of the cycle. The board features a JST female connector, which provides a strong connection when the battery is inserted. If the battery is not inserted, the board does not initialize the charge sequence, saving power.

Microcontroller Software

The price point and precision in measurements were only possible because of the MSP430F2254 microcontroller. The algorithm to determine rotational movement of an individual's hand required a fast and robust processor, which the MSP430F series of microcontrollers provided. The integrated 16MHz oscillator provided the necessary data manipulation at extremely fast speeds. The IMU algorithm consists of a Complimentary Filter partnered with Zero Gyroscope Drift (ZGD) filter derived by the group.

Basics of an IMU

An IMU is a set of sensors used in conjunction to provide rotational or positional movements relative to a single reference point. The WHS focused on rotational movements using a person's wrist as a reference point. Three rotational movements can be derived using an IMU. The first is yaw, which is the Z-axis rotation and analogous to heading. The second is pitch, which is Y-axis rotation. The third is roll, which is X-axis rotation. In combination, they describe all rotational directions for any object. A typical IMU consist of an accelerometer, gyroscope, and magnetometer. Due to the ZGD algorithm derived by the group, the magnetometer is not required therefore reducing costs significantly. The three sensors can be further explained as:

1) *Accelerometer:* An accelerometer measures rotational acceleration about a point. This application uses an accelerometer for tilt sensing with respect to Earth's surface. This sensor is affected by gravity.

2) *Gyroscope:* A gyroscope measures angular velocity, or how fast an object is spinning about an axis. Gyroscopes are not affected by gravity so they complement accelerometers when fusion algorithms are conducted.

3) *Magnetometer:* A magnetometer is a tool used to measure the direction and strength of magnetic fields in the device's location. These devices are important when one wants to know the change in angle from a starting point.

Complimentary Filter Algorithm

The algorithm derived by Shane Colton of Massachusetts Institute of Technology fuses the accelerometer with the gyroscope to provide accurate measurements [7]. An individual sensor is not enough to determine an angle of rotation. Gyroscopes tend to drift as time increases, imitating low-pass filter characteristics. Accelerometers tend to be inaccurate at short time intervals imitating a high-pass filter. Individually, the two sensors are too inaccurate to determine angles, but together, they can provide accurate angles of rotation.

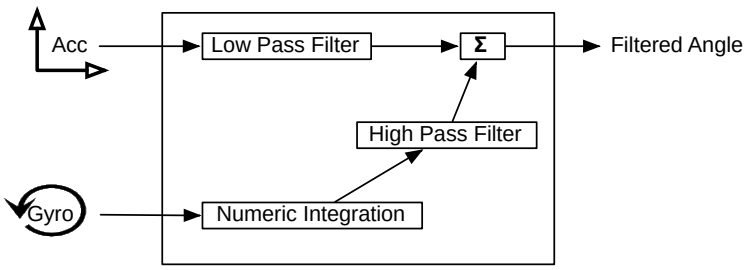


Figure 4 - Complimentary filter

Figure 4 depicts the complimentary filter algorithm process. This algorithm relies heavily on the sampling time of the microprocessor. The MSP430F2254 provides a 16MHz processor allowing for smaller sampling periods and more accurate results [8]. The numeric integration for the gyroscope shown in Figure 4 is more accurate with a higher sampling frequency. With the MSP430F2254, the processor can sample the algorithm at 200 Hz providing highly accurate movements in rotation. The core equation shown in Equation 1 derives the pitch angle.

$$pitch = [(HPFfactor) * (pitch + gyroY * \Delta t)] + (LPFfactor) * (accX)$$

Equation 1 – Complimentary filter equation

HPFfactor: high pass filter scalar
LPFfactor: low pass filter scalar
Δt (dt): sampling time dependent on frequency of operation
gyroY: raw gyroscope y-axis data from sensor
AccX: raw accelerometer x-axis data from sensor

The HPFfactor ranges between 0 and 1 and the LPFfactor is 1 minus the HPFfactor. One needs to determine with the robustness of their sensors which sensor takes precedence. If the gyroscope is able to provide more accurate measurements, then it must take precedence. Precedence can be determined via α, the time constant. Due to the complimentary effect of the accelerometer and gyroscope, one or the other needs to be filtered out giving precedence to the better measurement. This can be done with Equation 2.

$$\alpha = \frac{HPFfactor * \Delta t}{1 - HPFfactor}$$

Equation 2 - Filter precedence equation

Usually α, the time constant, is chosen and then used to determine the *HPFfactor*. For a low-pass filter, signals much longer than the time constant pass through unaltered while signals shorter than the time constant are filtered out. The inverse is true for a high-pass filter. This provides the complimentary effect of one sensor on the other because gyroscopes act as low-pass and accelerometers as high-pass filters. This algorithm successfully calculated pitch and roll but not yaw.

Yaw, the rotational movement along the Z-axis, could not be calculated because when rotating the accelerometer parallel to a surface, such as a table, the accelerometer Z-axis does not change. Gravity acts perpendicular to the Z-axis providing no change in angle relative to a surface. The accelerometer data was unusable for yaw measurement. This is where the gyroscope becomes a factor. Due to its independence of gravity, the gyroscope provided the angular velocity around the z-axis. This required the ZGD algorithm derived by the group to compute yaw measurements.

Zero Gyroscope Drift Algorithm (ZGD)

The removal of the accelerometer in yaw measurements introduces the issue of gyroscope drift. This issue is persistent in all applications that utilize an IMU. To mitigate the issue without regards to cost, a magnetometer in conjunction with a gyroscope can be used. However, a single gyroscope was used to solve the problem because angular velocity is sufficient for the application. With further research, it was determined that gyroscopes tend to increase in a linear

fashion upwards or downwards from their initial points. From testing, the observation was made that the drift occurred due to the hundredths place of gyroscope data increasing or decreasing rapidly thus altering the yaw angle. Therefore, this algorithm was applied:

$$tempGyro = (int)[gyroZ * (1000 * roundingFactor)]$$
$$finalYaw = \frac{tempGyro}{factor}$$

Equation 3 - ZPD equation

gyroZ: raw gyroscope y-axis data from sensor
finalYaw: filtered and stable yaw angle

Equation 3 is an algorithm that rounds the uncontrollable gyroscope z-axis values to a steady value. The *roundingFactor* is in a range from 0.25 to 0.5, which handles the accuracy the application needs. This resulted in the yaw value solely calculated from the gyroscope. Benefits of this include reducing the cost of adding a magnetometer, lowering microcontroller intensive operations, and keeping a small circuit board footprint.

Results of IMU Algorithms

The results of these algorithms can be seen via data points collected for a specific sample time. Figure 5 displays the relationship between the complimentary filter algorithm pitch (red) versus the non-filtered pitch (blue) values from the accelerometer.

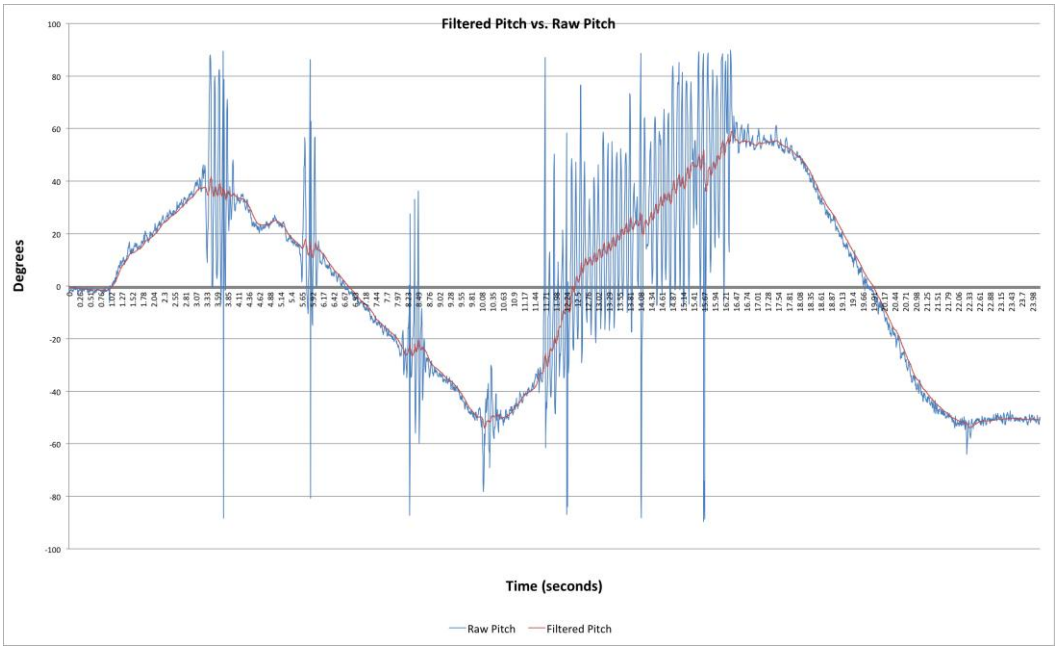


Figure 5 - Filtered pitch verses raw pitch

The output angles range between ±90 degrees. The sensors were secured to a person's hand to replicate the application. The user rotated their hands in a steady manner upwards and downwards and randomly shook their hand rigorously. The filtered pitch stays in a smooth reliable variance pattern. On the other hand, the raw pitch was providing inaccurate peaks that

would displace the mouse on the computer screen drastically. The filter algorithm successfully removed inaccuracies in the data, providing excellent control of the computer mouse.

In Figure 6, the ZGD algorithm results are presented. In this situation, the system was started and the user moved their hand to an angle of 7.5 degrees. The non-filtered values are the gyroscope values that do not go through the ZGD algorithm and decrease in a stepwise linear relationship. This is an indication of drifting error the gyroscope experiences. The ZGD filtered values stay constant at the angle with no alteration. The results in figure 6 show that the ZGD algorithm provided accurate yaw measurements required for the WHS.

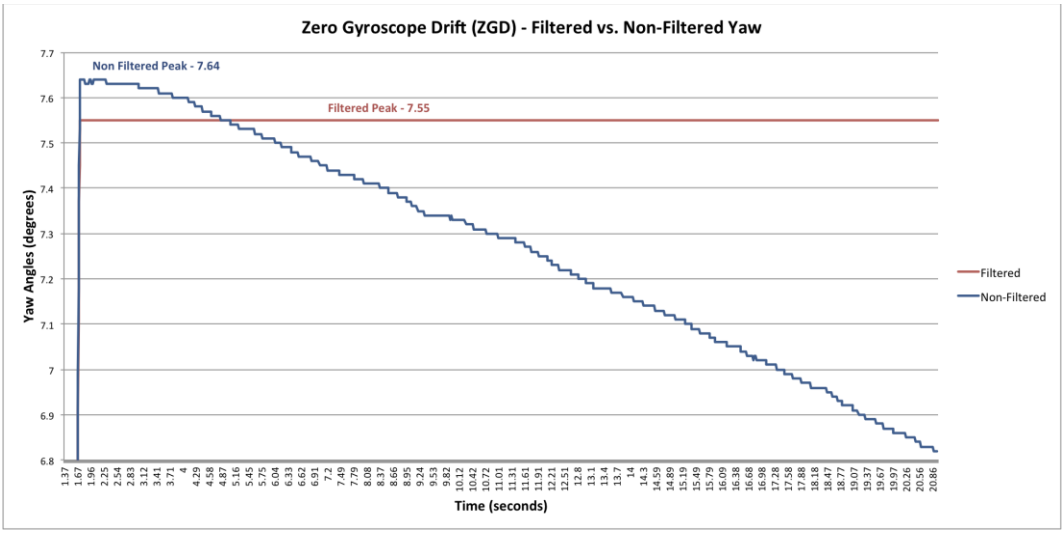


Figure 6 - ZPD algorithm

Consistency and accuracy for the IMU were crucial in the success of this project. Hand rotational tracking requires accuracy in measurements and an inaccurate angle calculation can alter the intuitive feeling of controlling the computer mouse. The WHS development emphasized the accuracy of the IMU and was successful in accomplishing that.

Computer Software

The WHS transmits data to a computer that runs a mouse driver with a user interface for customizing settings written in Python scripting language. This was created so that a wide audience could have access to the device. Once running, the graphical user interface (GUI) provides users with a method of customizing their WHS experience. Settings such as the overall sensitivity of mouse movements on screen can be adjusted to get the best response for a given application. Additionally, users can set the function of each of the eight touch sensors located on the fingers.

A connection is first established over Bluetooth by emulating a serial COM port. This COM port communicates at a baud rate of 115200 $\frac{bits}{sec}$. A two-way handshake takes place, which identifies the transmitter and receiver as the correct devices using a request and confirmation byte. Once a reliable connection is established, the WHS sends data bytes with the following format show in Table 1.

Pitch Angle	Yaw Angle	Roll Angle	Button 1 state	Button 2 state	Button 3 state	…	Button 8 state

Table 1 - Data transmission pattern

The values are read in then converted to floating point variables where computer manipulation begins. The driver works by saving the mouse position on screen along with the initial angles outputted by the WHS. These values are then bound together, mapping those specific angles to X-Y position on screen. As the glove is tilted, the pitch and yaw angles are updated, thus deriving new angles in the process described by Equation 4.

$$Displacement\ Angle = Current\ Angle - Initial\ Angle$$

Equation 4 - Computer angle displacement

This displacement angle identifies how far pitch or yaw has moved since the program started. This angle is sent into Equation 5 which further process the data by assigning a new X-Y position on screen based on the current sensitivity.

$$New\ X\ position = [InitialX - (yawDisplacement * xSensitivity)]$$
$$New\ Y\ position = [InitialY - (pitchDisplacement * ySensitivity)$$

Equation 5 - Computer x-y movement equations

These processes result in accurate hand mapping that is intuitive for the user. The final outcome is the use of yaw and pitch angles derived from algorithms on the WHS, which are mapped into pixel coordinates on a computer screen.

Texas Instruments verses Atmel Version

The initial development of the WHS was created with Atmel ATmega328 microcontroller. The Atmel controllers allowed for a smaller learning curve due to prior knowledge by the group. In the Atmel version, the bill of materials cost was higher due to the requirement of using Texas Instruments 74HC5095 shift register and Texas Instruments 74HC4051 multiplexer. Eight dedicated analog input pins are required to handle the touch sensing capabilities. The ATmega328 microcontroller contained eight analog pins, but two were dedicated to I2C peripheral leaving six pins. This created the requirement for an 8:1 multiplexer [9]. Also, due to the lack of digital pins to drive the feedback technology on the WHS, a shift register was implemented to drive the LED's. This caused major issues in terms of cost, complexity, and PCB footprint.

Engineers strive to design systems that combine effectiveness and complexity to bring down costs of a product. The decision was made to redesign the WHS based on Texas Instruments (TI) parts and it was highly successful. The new TI design reduced costs to under $50.00, making it competitive with a traditional wireless mouse. The new design used three TI analog ICs: MSP430F2254 microcontroller, TPS77633 voltage regulator, and BQ24080 Lithium Ion battery charger.

MSP430F2254

After researching other microcontrollers such as Microchip and other Atmel chips, the team decided to redesign the WHS with TI MSP430F2254 microcontroller. The MSP430 microcontroller allowed the removal of the shift register and multiplexer because of the

controllers dedicated eight analog pins and numerous digital pins. Due to the IMU algorithm requiring fast sampling with 3.3V operation voltage for the MPU6000, the MSP430 controller provided 16MHz at 3.3V. The Atmel controller provided 8MHz at 3.3V. The TI chip allowed doubling of the sampling time giving more precise calculations. The PCB footprint was reduced from 2 by 1.2 inches to 1.7 by 1 inches. The TI version of WHS reduced costs, putting the WHS under $50.

The MSP430F2254's internal oscillator provided extra benefits. The Atmel controller required an external oscillator and the speed was not variable. The MSP430 internal variable oscillator allowed the WHS to be more robust. The 16MHz was used after thorough benchmark research on the benefits on the IMU algorithm with 8MHz verses 16MHz. The ability to change processor speeds puts the MSP430 among the elites in microcontroller technologies. This was a limiting factor with Atmel chips due to being forced to use a specific oscillator.

The Atmel controllers incorporated six ADC inputs. In the first design, a multiplexer was used to provide extra inputs. Alternatively, the MSP430F2254 features eight dedicated ADC inputs, eliminating the need for a multiplexer [8]. The ability to use the analog pins as interrupts allows for future development to introduce more features for the project.

The MSP430F2254 microcontroller provided the WHS project with more options, better performance, and reliability. The new design is far superior to the Atmel version due to cost reduction, size reduction, and increase in efficiency.

TPS77633

The TPS77633 is the only IC on the WHS that was featured in both Atmel and TI designs. Due to its ability reduce the battery input voltage to a steady 3.3V; it was a perfect choice for this project. All of the components on the WHS require a 3.3V supply. The fixed output 3.3V LDO regulator is an ideal solution. The power dissipation of the WHS needed to be within a tolerance of 567mW according to Equation 6 [10]. The WHS requires 40mA maximum at 3.3V, totaling 132mW. This is well within the 567mW tolerance allowing reliable operation.

$$P_{D(max)} = \frac{T_{J(max)} - T_A}{R_{\theta JA}}$$

Equation 6 - Max power dissipation of TPS77633

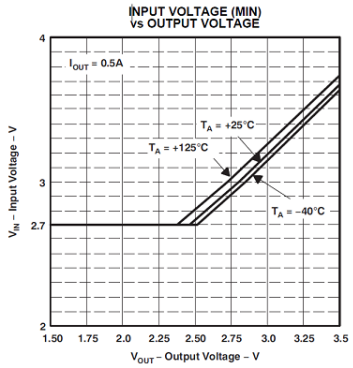


Figure 7 - Vout verses Vin of TPS77633

For further verification, Figure 7 was used. The project used a single-cell lithium ion rechargeable battery rated at 3.7V and 400mAh [6]. Temperature is an important aspect with this project due to the high temperature correlation of the MPU6000. The operating temperature for the device was room temperature or close to $T_a = 25^{\circ}$ Celsius. At the rated input voltage, the output voltage is 3.3V at the operating temperature via Figure 7.

The TPS77633 is a robust regulator that meets all the requirements of the WHS. The eight-pin SOIC package size allows for it to fit into small areas on a PCB. It is an excellent IC that has provided efficient voltage regulation.

BQ24080

The BQ24080 fulfilled the requirement to develop a lithium ion (Li-ion) battery charger [5]. This IC handles the special charging characteristics of Li-ion batteries. As a portable device, the WHS requires a USB based charger rather than an AC wall charger. USB ports source a maximum of 500mA of current and calculations were performed to determine the proper components to add in conjunction with the BQ24080 IC.

$$I_{o(OUT)} = \frac{K_{(SET)} * V_{(SET)}}{R_{SET}}$$

Equation 7 - Current limiting resistance of BQ24080

The crucial component was choosing a proper resistor to limit the output current to less than 500 mA. Using Equation 7, the specific resistance is calculated. Even though USB can source 500mA max, the charger was designed for 400mA to keep all devices in safe operating conditions [5]. To determine the resistance, $I_{o\ (OUT)}$ was set to 400mA. According to the datasheet, $K_{(SET)} = 322$ and $V_{(SET)} = 2.5V$. Therefore, $R_{SET} = 2012$ ohms and closest standard resistor value to R_{SET} is 2000 ohms.

To determine the condition of the charger when a battery is plugged in, the status and power good (PG) pins were attached with LEDs. This provided a visual recognition of the various stages of charging a Li-ion battery. The BQ24080 is an excellent IC that fits all the requirements of the WHS project.

Future Improvements

The WHS project envisions future improvements that would make the system more efficient and reduce costs. One such improvement is the use of a CC2540 Bluetooth system-on-chip solution [11]. This product can allow the WHS to feature cutting-edge technology in wireless transmission. Currently, the use of Roving Networks RN42 Bluetooth module is sufficient but future designs will be built on a single PCB module. With very low production, the CC2540 IC costs $5.62 compared to the RN42 at $15.00. This can significantly lower the price point for the WHS. Apart from price, the CC2540 will allow the WHS to feature all TI products, which are competitive in price and robustness. Due to the vibrant technical community and widespread support of Texas Instruments, this new feature will be realizable.

Another improvement that will increase the responsiveness of the WHS is to switch from ADC inputs to interrupt based inputs. This feature is already complete in terms of testing via the prototype but is not featured in this specific build discussed. The MSP430F2254 features two ports (PORT1 and PORT2) for interrupts and the WHS requires a total of eight interrupts [8]. Therefore, the ability to detect an interrupt from low to high would allow the constant ADC polling to be removed. This will give more processing speed to the IMU algorithms. These are a few ideas that will be explored in the near future to improve the WHS.

Practical Application for Boeing

Due to the precise angle measurements that the WHS provides, a research project of Dr. Huaping Lui at Oregon State University requested the WHS group to aid in a project funded by Boeing [12]. Dr. Liu's research team placed eight ultra-wideband localization sensors around a wheel well of an airplane allowing them to track the three-dimensional position of a wrench. The full system allows Boeing to check if specific bolts have been visited by the wrench prior to oil being pumped into an airplane. This mitigates risks of major leaks that are unsafe for the airplane and the environment. The research team had a significant issue that was not solvable with their current system. The issue mostly occurs when a technician has reached a bolt they must tighten. For example, the system may be successfully tracking the wrench upon reaching a bolt, but after the mechanic has made contact with the bolt and attempts to tighten it, the transceivers on each end of the wrench are hidden due to the hand placement. In cases where the wrench transceiver becomes unreachable via ultra-wideband localization, angle measurements from the WHS would provide movement information.

Solution: Angle to Distance Algorithm

The requirement for the WHS group was to derive the change in distance from a starting point given via a wireless link between the WHS board and an onboard microprocessor. One major requirement by Dr. Liu's team was to be within a five-centimeter range of the actual measurement. The results showed the algorithm derived was within *one centimeter*. The algorithm is derived in ***Appendix E***. The equations derived are shown in **Error! Reference source not found.**.

$$\Delta y = r\theta \cos\left(\frac{\pi}{2} - \frac{\theta}{2}\right)$$
$$\Delta x = r\theta \sin\left(\frac{\pi}{2} - \frac{\theta}{2}\right)$$

Equation 8 - Angle to distance algorithm

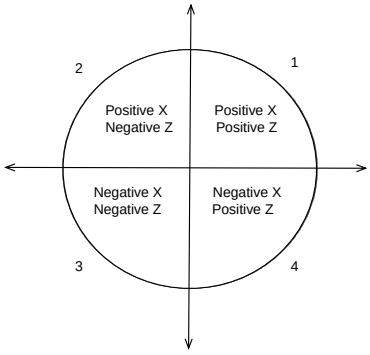


Figure 8 - Accelerometer signs in each quadrant

The WHS required ±90 degrees to control the computer mouse but this application required 360 degrees of rotation. This required an extra step of using the accelerometer raw values provided by the sensor to depict the sign changes as shown in Figure 8. Depending on the quadrant, the accelerometer x and z-axis outputs values are positive or negative. Using this consistent pattern,

angle offsets of 0, 180 and 360 degrees can be added to the filtered measurements to retrieve 360 degrees of rotation.

Results

To retrieve the change in position from a previous calculation to the next in Equation 8, only two variables are required to be known: radius and angle of movement. The radius, r, is a known physical measurement provided by the specifications of the tool being used. The mechanical tool used in ***Appendix E*** for derivation was a wrench and had a length of 25 centimeters. The other unknown, θ, is calculated via the Complimentary Filter and ZPD algorithms. The MSP430F2254 microcontroller's processing speed provided frequent angle measurements that resulted in highly accurate distances ensuing in Boeing implementing the board into their system.

Therefore, by knowing the length of the mechanical tool being used, the WHS board can calculate the exact displacement, giving this project numerous applications. Dr. Liu's research provided a platform to test algorithms, test consistency, and apply the WHS project to a research application.

Conclusion

After in-depth testing to meet requirements of the project, the outcome was a sensor that features the ability to track rotational movements. The rotational movements were used to control a computer mouse in an intuitive way. Though the project was developed for a computer environment, it has numerous applications in the medical sector, gaming, and robotics. The goal of this project was to process all algorithms on-chip, allowing the ability to mimic an application-programming interface (API). Researchers or general users can extract information from the API, allowing them to develop further applications in various sectors. Providing accurate hand movements opens up a new realm of possibilities for future development by individuals.

The core of the WHS relies on the IMU algorithms to provide accurate angles of rotation that are then translated to pixel movements on a computer screen. The user wears a glove, specifically made for this project, equipped with conductive touch sensing technology. The user can interact with the computer via a connection between the thumb to any of the two locations on a finger. Eight inputs are present for the user to customize via a user interface on the computer.

This project would not be possible without the extensive help of professors and teaching assistants of the senior design team at Oregon State University. Components and parts are crucial to development and thus Sparkfun.com and Digikey have been an ideal place to purchase from. Sincere thanks goes to Texas Instruments for putting together a design contest that was a motivator when developing this project. TI's evaluation modules provided the group's vision of the project to become a reality. Altogether, TI has been of much help for the success of this project.

Works Cited

[1] (2012) Sparkfun. [Online]. http://www.sparkfun.com/products/10055

[2] Conductive Thread (Thin) - Sparkfun Electronics. [Online]. http://www.sparkfun.com/products/10118

[3] (2011, Nov.) MPU-6000/6050 Six-Axis (Gyro + Accelerometer) Datasheet. [Online]. http://www.invensense.com/mems/gyro/documents/PS-MPU-6000A.pdf

[4] (2012, Apr.) RN42 Bluetooth Module Datasheet. [Online]. http://www.rovingnetworks.com/resources/download/19/RN_42

[5] BQ24080 Lithium Ion Battery Charger IC Datasheet. [Online]. http://www.ti.com/product/bq24080

[6] Polymer Lithium Ion Battery - 400mAh - Sparkfun Electronics. [Online]. http://www.sparkfun.com/products/10718

[7] Shane Colton. (2007) MIT. [Online]. http://web.mit.edu/scolton/www/filter.pdf

[8] MSP430F2254. [Online]. http://www.ti.com/product/msp430f2254

[9] ATmega328 Microcontroller Datasheet. [Online]. http://www.atmel.com/Images/doc8161.pdf

[10] TPS77633 LDO Regulator Datasheet. [Online]. http://www.ti.com/product/tps77633

[11] CC2540 Bluetooth Low Energy System On Chip. [Online]. http://www.ti.com/product/cc2540

[12] Dr. Huaping Liu. [Online]. http://eecs.oregonstate.edu/people/liu

Appendix

A. Project Images

Figure 9 - WHS PCB front

Figure 10 - WHS full design

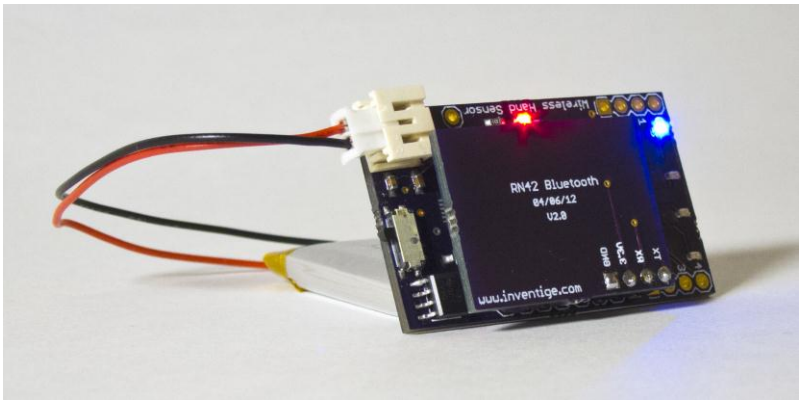


Figure 11 - WHS PCB angled

B. Bill of Materials

Part Name	Description	Price/Unit	Units	Total Cost
Voltage Regulator	TPS77633	$2.32	1	$2.32
MPU6000	MPU6000	$15.00	1	$15.00
MSP430F2254	MSP430F2254	$4.10	1	$4.10
RN42 Bluetooth	RN42	$14.95	1	$14.95
0.1uF capacitor	0.1uF	$0.05	5	$0.25
4.7k resistor	4.7K	$0.03	2	$0.06
10uF capacitor	10uF	$0.07	1	$0.07
2.2nF capacitor	2.2nF	$0.03	1	$0.03
Glove	Glove	$3.00	1	$3.00
Thread	Conductive Thread (per feet)	$0.18	3	$0.54
Fabric	Conductive Fabric (per in^2)	$0.10	3	$0.30
LED	Blue (4) and Red (1) LED	$0.25	5	$1.25
Battery	400mAh capacity	$3.95	1	$3.95
BQ24080 Charger IC	BQ24080	$2.71	1	$2.71
JST Battery Connector	JST	$0.25	2	$0.50
USB connector	USB Type A	$0.75	1	$0.75
Total				***$49.78***

** Price per unit from Digikey, Mouser and Sparkfun.com*

C. Project Demonstration

The project was demonstrated at the 2012 Oregon State University Engineering Expo with an audience of 1500 from kids to industry professionals. The WHS demonstrated its intuitiveness by using Google Earth. People at the event were highly enthusiastic about the project. A video demonstration of controlling Google Earth can be seen at the link below:

http://vimeo.com/user11877976/wireless-hand-sensor-google-earth-demo

D. Recognitions

The WHS received first place in 2012 Oregon State University's Industry Award voted on by Garmin, Intel, Hewlett Packard, Portland General Electric, ON Semiconductor and more. The friendly competition with 35 other projects in Electrical Engineering was ranked in terms of complete product, presentation and technicality.

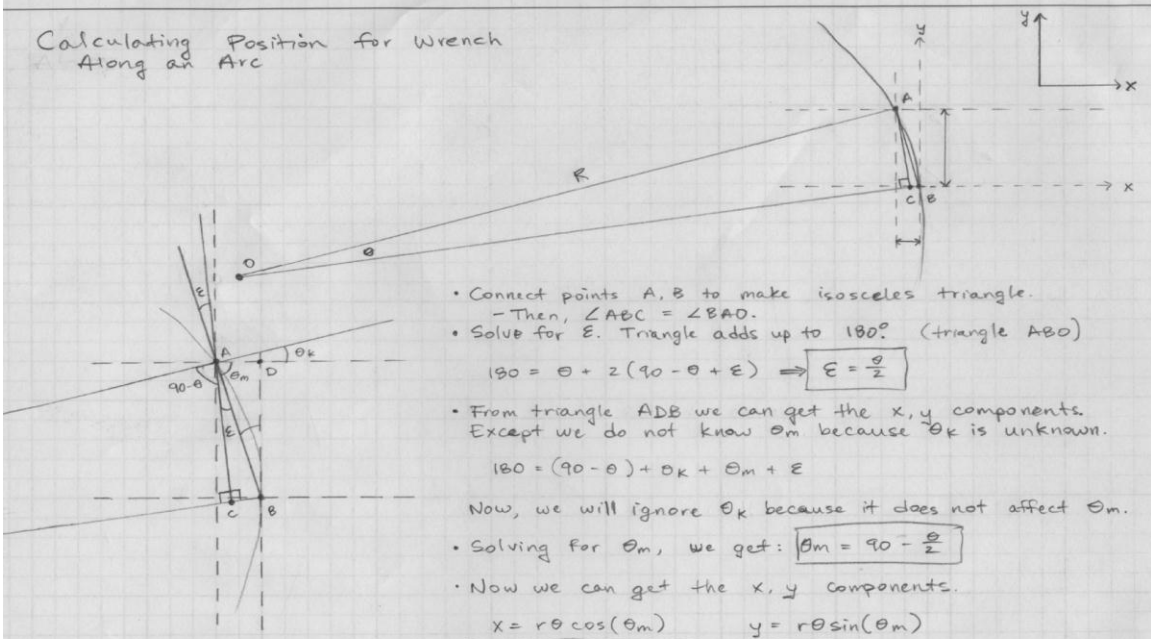
The Oregon State University College of Engineering Newsletter featured an article about the WHS in its June 2012 issue. The article can be viewed below:

http://blogs.oregonstate.edu/engineering/2012/06/06/wireless-glove-controls-computers/

E. Practical Application for Boeing: Algorithm Derivation

1/2

Calculating Position for Wrench Along an Arc



- Connect points A, B to make isosceles triangle.
  - Then, $\angle ABC = \angle BAO$.
- Solve for ε. Triangle adds up to 180° (triangle ABO)

$$180 = \theta + 2(90 - \theta + \varepsilon) \Rightarrow \boxed{\varepsilon = \frac{\theta}{2}}$$

- From triangle ADB we can get the x, y components. Except we do not know θ_m because θ_k is unknown.

$$180 = (90 - \theta) + \theta_k + \theta_m + \varepsilon$$

Now, we will ignore θ_k because it does not affect θ_m.

- Solving for θ_m, we get: $\boxed{\theta_m = 90 - \frac{\theta}{2}}$
- Now we can get the x, y components.

$$x = r\theta\cos(\theta_m) \qquad y = r\theta\sin(\theta_m)$$

$$\boxed{x = r\theta\cos\left(\frac{\pi}{2} - \frac{\theta}{2}\right)} \qquad \boxed{y = r\theta\sin\left(\frac{\pi}{2} - \frac{\theta}{2}\right)}$$

2/2

Summary

- Ignoring θ_k we can estimate the x and y displacement as:

$$x = r\theta\cos\left(\frac{\pi}{2} - \frac{\theta}{2}\right), \quad y = r\theta\sin\left(\frac{\pi}{2} - \frac{\theta}{2}\right).$$

Analysis

- We always knew that we could quickly estimate the x and y displacement as:

$$x = r - r\cos\theta = r(1 - \cos\theta), \quad y = r\sin\theta.$$

- However, we found that this quick estimate is very close to our previous equation:

$r = 25$ cm
$\theta = 10° = \frac{10\pi}{180}$ rad

	DERIVED VALUE	ESTIMATED VALUE	MEASURED VALUE (with protractor)
x	$(25\text{ cm})\left(\frac{10\pi}{180}\right)\cos\left(\frac{\pi}{2} - \frac{1}{2}\frac{10\pi}{180}\right)$ $= 0.3803$ cm	$r\left(1 - \cos\left(\frac{10\pi}{180}\right)\right)$ $= 0.3798$ cm	$x = 0.38$ cm
y	$(25\text{ cm})\left(\frac{10\pi}{180}\right)\sin\left(\frac{\pi}{2} - \frac{1}{2}\frac{10\pi}{180}\right)$ $= 4.297$ cm	$r\sin\left(\frac{10\pi}{180}\right)$ $= 4.341$ cm	$y = 4.2$ cm
% ERROR (to measured value)	x: 0.079%	x: 0.053%	* Our estimated and derived values are within the 1 cm range.
	y: 2.30%	y: 3.36%	