Accelerometer-Based, Grip-Free Controller

By

You-Chi (Tyler) Le

Instructor:

Dr. Guilherme DeSouza

ECE 4220 – Real Time Embedded Computing Fall 2011 Department of Electrical and Computer Engineering University of Missouri – Columbia Columbia, Missouri December 9, 2011

Abstract

The project described this report is a hand-orientation sensing controller composed of a two-axis accelerometer affixed to a glove. The accelerometer is connected to a TS-7250 Embedded System using two analog-to-digital conversion (ADC) lines as well as power and ground connections. The project is designed to provide a cheap, flexible, and dynamic controller for both practical and entertainment usages. It provides motion sensing in environments inappropriate for cameras and optical sensors. It also provides an alternative method of vehicular control for people with hand injuries and/or lacking grip strength. The prototype allows a third potential usage in demonstrating an effective yet affordable joystick intended for entertainment and video games. The motivations behind this project include simplifying and optimizing vehicular control for people with disabilities, replacing current joysticks with grip-less motionsensing, and providing a replacement for countless single-purpose controllers. Current existing implementation of motion-sensing and orientation-detection are mostly joystick or camera-based, and the prototype described in this report aims to produce alternative control methods for similar target applications. Such contribution allows cost-saving techniques that can benefit both gamers and people lacking the ability to grip objects. A customizable range of motion is also featured in this project along with two example simulations demonstrating both practical and leisure implementation of the prototype. The project demonstrates an accurate portrayal of the mapping of hand-orientation through visual displays. It also demonstrates the potential to improve upon current technologies by demonstrating effective controlling methods in the accompanying simulations. Simultaneously, this prototype exemplifies a real-time-centric program featuring key concepts presented in this course through its software implementation.

Introduction

This project was originally intended to demonstrate a real time-based software system that contributes to practical applications in real life. However, the resulting design exceeds its intentions and now has the potential to alter current gaming technology and improve vehicular control for disabled persons. Current motion detection and orientation-sensing technologies are primarily based on image processing and physical levers. Although effective in countless implementations and scenarios, they are limited by relatively high costs, lighting of surroundings, separation and spacing requirements, and/or single-purpose designs. The idea of this project is to suggest an accelerometer-based joystick design in situations where the aforementioned technologies are inappropriate or ineffective. While mobile devices such as iPod's and smartphones already feature accelerometers and gyroscope-integrated motion controls, these technologies are rarely implemented on larger-scale applications and prove to be very costly for budget-conscious projects. The introduction of a cheap and flexible control mechanism that bypasses the necessity of gripping has the potential of benefitting a wide variety of individuals. Its benefits range from allowing people with hand-injuries/limitations to operate vehicles without gripping to allowing PC gamers to operate simulated vehicles without buying an array of costly equipment. Rather than utilizing the typical positional joystick or optical sensing controllers, users can simply put on a glove to operate an assortment of devices and simulations. Such glove, when accompanied with a dynamically configured range of motion, allows applications to be more tailored towards the users' comfort and abilities. Ultimately, this project resolves complications of traditional motion detection systems by producing a cheap and customizable control scheme that centers on flexibility and adaptability.

Background

Application

The main purpose of this project is to demonstrate the applicability of an accelerometerbased glove controller. In order to do so, example programs must be packaged to help visualize the potential usage of the prototype. As a result, two programs with graphical displays are designed in parallel with the controller to contribute to visualization of the project's future applications.

Problem Statement

As mentioned before, the target for this project is centered on mapping the accelerometer's analog outputs. The signal levels from the axes must be correctly and accurately interpreted to produce a usable product. Similarly, these signals must also be utilized in such a way so that the potential of the prototype can be shown. Essentially, the goal of this project is to create methods of tying the accelerometer signals to software appropriate for real-life applications.

Alternative Implementations

While utilizing analog accelerometers and ADC's seem to be the most cost-efficient way to implement the prototype, alternative implementation methods are considered. For instance, one possible method of constructing such a controller can be found using digital accelerometers using serial communication rather than analog output. Such a design is limited by the higher cost of the digital accelerometers. Another method of implementing this project can be found in the modification of preexisting hardware such as a Nintendo Wii controller or an iPod Touch. However, as mentioned before, such devices prove to be too costly to be considered, and the idea of benefitting people with hand injuries could not be realized. Other methods such as utilizing the Xbox Kinect or video analysis are overlooked due to the reasons mentioned in the previous sections of this report. Overly complex algorithms also fall outside of the scope of the materials taught in this class.

Proposed Implementation

Hardware:

As the project must revolve around materials associated with the class, the TS-7250 Embedded System is the clear choice for central processing. The accelerometer is affixed to the top of a glove to allow grip-less control of the simulations. The accelerometer is selected to be an analog, 3-axis chip to lower the overall cost of the prototype. The model for the accelerometer is the MMA7361 integrated chip mounted on a breakout board made available at <u>www.moderndevice.com</u>. The chip itself allows three-axis functionality, sensitivity selection, and sleep activation. It also allows an operation voltage from 3.3V to 5V. Power for the chip is drawn for the digital I/O lines of the TS-7250, and analog measurements are made by the onboard, 16 bit ADC's due to the fact that the more accurate MAX197 ADC's are not currently installed on the boards available in lab. Only two of the three axes are currently in use.



Figure 1: Block Diagram of Hardware Setup

Software:

The control registers to operate the on-board ADC's are mapped using mmap() and the /dev/mem file. The result register for the ADC measurements are also mapped using the same method. Four threads are initialized with periodic real time tasks in order to operate in sequence. Such operation produces the necessary steps to measure the output signals, update graphics logic, and print the display. Each of the real time tasks operate with a period of 0.1 seconds to produce an overall refresh and sample rate of 10Hz. Graphics are printed using the Linux Terminal along with ANSI escape sequences for smooth transitions. The ADC measurements are compared to thresholds measured at initialization. Such thresholds are results of the range of motion divided into fixed intervals.

Real Time Structure:

The first two real time tasks poll the output of the 2-axis accelerometer at a rate of 10 Hz to store the information into a shared buffer. The next real time task then pulls the information from the shared buffer to make logic decisions and graphical updates based on the mode of

operation and previous inputs. The last real time task prints the superimposed layers of character arrays to produce a graphical output at the Linux Terminal. Each task prevents its successor from interrupting operation using semaphores. Essentially, the first ADC measurement task has the highest priority, effectively locking out any interruption from the next three tasks. The next two tasks then repeat the procedure to ensure synchronized and consistent operation order.



Figure 2: Real Time Software Implementation Block Diagram

Real Time Concepts Used:

The software used in the two accompanying simulation programs involves several real time concepts studied in class:

- 1. <u>Pthread</u> The program takes advantage of multi-threaded operation for smooth transitioning and encapsulated tasks
- <u>Periodic RT Tasks</u> The program utilizes this to operate a on a fixed schedule with real time constraints
- 3. <u>Semaphores</u> The flagging system maintains the correct order of operation
- 4. <u>Shared Buffers</u> Shared memory allows inter-thread communication of input values
- 5. <u>Producer-Consumer Model</u> The threaded ADC measurements serve as producers that produce data for the graphic processing consumers
- 6. <u>I/O Polling</u> This is implemented in order to read the ADC results
- 7. <u>Priority Scheduling</u> Semaphores maintain a prioritized order of thread execution

Tests/Results:

The results and evaluations of this prototype are based on the graphical outputs of simulation programs. The driving simulation program and the pilot program demonstrate the mapping of user motion through the accelerometer and analog-digital-converter.





As the screenshots from the driving demonstration video show, the user's hand position corresponds with the displayed hand position.

Similarly, the pilot simulation program shows that the displayed model of the plane is consistent with the orientation of the user's hand.



The final cost of the prototype is approximately \$15 dollars assuming generic glove and wires.

This successfully demonstrates a cost-friendly implementation for the project.

Conclusion:

Although the system is implemented using the less accurate and slower on-board ADC's, the simulations demonstrates that the prototype provides enough accuracy to prevent erroneous reading. The simulations demonstrate motion-sensing results consistent with the hand motion of the user, and similarly, the two modes perform as intended as the driving and flying program illustrate practical and leisure implementations of the prototype. From this, a conclusion can be drawn stating the success of the simulation package that accompanies the glove controller. We can assume that the real-time-based, threaded tasks involved in the software implementation of this project also perform as desired. As the graphical outputs of the project demonstrate the potential usage of this prototype, they simultaneously confirm that the sequential and synchronized tasks of reading converted inputs, updating program logic, and printing graphics operate flawlessly. From the successful representation of the orientation of the user's hand, we can also confirm that the ADC control registers are mapped correctly. Essentially, as each component of the project operate as intended by design, we can conclude that the prototype successfully demonstrates a flexible and configurable controller.

The successful demonstration of this project shows promises of revolutionizing motionbased controls through cost-effective implementations. Such implementations have nearly limitless potentials ranging from altering the way injured hands operate vehicles to providing a new generation of PC-gaming joysticks.

Several options can already be foreseen for future optimization. First, a migration to the optional MAX197 ADC (not currently available in the lab) would provide better accuracy and resolution to reduce minimum range of motion. Second, expansion of threads and tasks can provide the ability to utilize multiple axes in two-handed operations. Additionally, utilizing

external display drivers can minimize process time and increase the sampling rate of the program. Wireless integration of the ADC and accelerometer would also provide more freedom for user motion. Last and certainly not least, averaging algorithms can be utilized to prevent accidental acceleration due to unintended sudden motion. All these potential improvements promise a future of improved robotic, vehicular, and gaming control for cost-sensitive users.