Button Controller Senior Project Proposal

1st Brian Burton
Computer Engineering Dept.
University of Utah
Salt Lake City, United States
u1038667@umail.utah.edu

2st Matthew Hunsaker
Computer Engineering Dept.
University of Utah
Salt Lake City, United States
u0697241@umail.utah.edu

3rd Kashish Singh
Computer Engineering Dept.
University of Utah
Salt Lake City, United States
u1174165@umail.utah.edu

4th Khang Lieu
Computer Engineering Dept.
University of Utah
Salt Lake City, United States
u1037344@umail.utah.edu

***Abstract*—The document is a proposal and report for the University of Utah computer engineering senior project. The proposed product is a tool that is aimed towards those who are disabled and cannot use a conventional gaming controller. Buttons are placed on the ground and are connected to a device that can tell when they are pressed. Those presses are then translated into a format that mimics a conventional controller on a PC.**

***Index Terms*—computer engineering, computer science, software, computer hardware, game controller**

I. Introduction

The video games have become a very prevalent form of media and entertainment over the past several decades with the games increasing in complexity over that time. However, the standard method for playing these games, using a keyboard and mouse or a controller, requires significant use of and dexterity with a user's fingers. This excludes a portion of the population that, for whatever reason, can no longer use their fingers effectively enough to utilize these devices. Our goal is to develop a controller that can be used by such an individual to play video games on a PC using other body parts like their feet or wrists while still providing the full functionality of a conventional video game controller.

II. Project Tasks

In order to accomplish our goal, we are breaking it up into smaller and more manageable pieces. Below is a list of specific tasks that when accomplished will produce our desired product.

- Obtaining required materials listed in the Bill of Materials.
- Connect the button circuits to the Raspberry Pi I/O registers. We will use the forty (40) I/O pins on the Raspberry Pi to read the input from the buttons. One possible method that we have discussed to do this for a single button that we would repeat for all of the buttons is to set an output pin on the Raspberry Pi to constantly output a high voltage that is passed to the input voltage of the button. The output path of the button would be connected to an input pin on the Raspberry Pi, therefore when the button is pressed the circuit is completed and the input pin of the Raspberry Pi detects a logical high voltage that can be read.
- Develop a method to mimic the joystick movement and connect it to the Raspberry Pi I/O registers.
- Write a driver for the Raspberry Pi to convert the button inputs to output on a single USB. This driver would check the inputs from the buttons through the I/O pins of the Raspberry Pi. It would look for a logical high value on the pins and then set the corresponding pin in the USB to an appropriate value. The USB would then output that signal to the game.
- 3D print housings and larger covers for the I/O buttons. These would hold the button circuits and provide a larger mechanism for the user to interact with. They would also need to be fairly sturdy to withstand possibly a literal pounding by the user.
- Optional: Create a small game to show off functionality of the project. This game would be developed using either the Unity or Unreal game engines and would be a simple demo utilizing all of the buttons and some button combinations to better display the

controller's functionality.
- Optional: Add Bluetooth connections for the buttons to allow for wireless configurations.

III. Testing and Integration Strategy

As we progress though our design, we will need to debug each task to safeguard our unique controller. The list below is our testing plan for each of the tasks mentioned in our Project Tasks section.

- Create and implement a single button and first test if the Raspberry Pi receives a signal. Next part is to see if the USB connection sends the signal. (such as print the character 'a' into a text document like notepad)
- Add more buttons and test if they correctly send signals and don't interfere with each other.
- Implement a joystick design and test how the motion operates and if it's processing correctly.
- Implement 3D printed housing and larger covers for I/O buttons and test the durability of each housing and cover.
- Optional: Create a game and go through bug testing and fixes for the game. Then go through bug testing and fixes for implementing the hardware with the game and making sure the entire system functions.
- Optional: Create and test Bluetooth connection with buttons.

Once each of these errands are functioning properly, our work then turns to putting them all together. We will integrate them by connecting our buttons and joystick to the Raspberry Pi and then connect the Raspberry Pi to the computer via USB.

One of our final tests is when the real fun begins. It will include plugging in our controller and actually playing a video game.

IV. Group Communication Plan

Communication via group text messaging and Discord is utilized with meetings happening weekly. Work is saved and organized on different Discord channels as well as a GitHub repository that will hold copies of files and documents as well as eventual software code. Updates and reports on meetings is sent to instructor on a weekly basis.

V. Schedule and Milestones

The following flow diagram demonstrates who will be working on which tasks and the order of which they will be done.

April 26th - Have the full proposal and ready for presentation completed for ECE 3992 course. (Kashish Singh)

April 30th - Create and show proof of button to Raspberry Pi I/O connection and signals. (Brian Burton)

May 31st - Have major buttons connected and tested. (Khang Lieu)

June 30th - Have some work done with joystick. (Matthew Hunsaker)

July 31st - Complete and test joystick implementation and begin working on having all components work together. (Matthew Hunsaker)

August 31st - Goal to have completed all implementation and testing. If it's done faster than expected then goal is to have nearly everything for the base requirements completed before moving on to optional ones. (Entire Team)

August 31st - Have 3D printing files ready and beginning production if need be. (Kashish Singh)

September - December - If any milestones or features have not been fully implemented or tested, we still have this period of 3 months to continue working and fixing any issues and/or working on the additional optional features.

Optional - By December - Have major parts of the software completed such as character movement done and work on testing.

Optional - By December - Work and test on Bluetooth connections with buttons.

VI. Risk Assessment

With each project comes risks. Below describes the different risk levels, the risk nature, and the mitigation plan in case of failure for each of our product tasks.

- Obtaining required materials: Low Risk - We already ordered some of the major parts and others can be mass ordered. In order to avoid any issues however, it'll be a good idea to order materials quickly rather than waiting once we're sure it's what we need.
- Button connections to Raspberry Pi: Medium Risk - A lot of the things we've learned in computer engineering is applicable but it's still using unknown devices and systems.
- Joystick mimicking: Medium Risk - Harder to find components and designs that work for creating and mimicking a joystick. To deal with this, more research and understanding on how joysticks operate and what electronic devices is available to us is

needed. After that understanding how to sense and differentiate between directions.

- 3D printing materials: High Risk - No substantial experience from anyone with 3D modeling or printing. To mitigate this we will need to research more on things like what filaments are idea for our use as well as how to correctly measure and model components.
- Optional: Video game: High Risk - Since this is an optional component it will likely be done later and thus potentially cause time issues. Development with games is not something everyone has done nor using a game engines specifically. So research will need to be done as well as learning the new system and coming up with a simple plan.
- Optional: Bluetooth: High Risk - Will likely be done towards the end so less time. Less people with any experience with Bluetooth so research into different buttons and networking will be needed.

VII. Bill of Materials

Our ability to complete our project hinges on utilizing different products. Below is a complete components list for our primary and secondary vendors.

- Raspberry Pi 4
- At least 20 push-buttons
- 3D printer filament (for U of U 3D printers)
- USB to USB cable
- 2+ Joystick rollers or equivalent
- Breadboard or PCB board
- Optional: Bluetooth compatible buttons

VIII. Conclusion

We have all seen the positive influence that video games can have on a persons life. They just have a way of transporting us into different worlds, dimensions, and universes. They can even help develop our brains by teaching us essential skills like problem solving, perseverance, and teamwork [1]. The only problem is, not everyone is able to experience the games that we love. The gaming industry's standard controller eliminates thousands of players from the game by assuming you have finger mobility. The Button Controller is here to change that!

References

[1] I. Granic, A. Lobel, and R. C. M. E. Engels, "The Benefits of Playing Video Games," *American Psychologist*, vol. 69, no. 1, pp. 66–78, Jan 2014.