Institute Project 4th Semester

iNETS – Communication System for embedded sensors

ControllerX

On-air-gyroscope-mouse and remote media controller

Jin 374385

Adhita Putri Nugraheni 374476

RWTH Aachen University

Elektrotechnik, Informationstechnik und Technische Informatik B.Sc

1. Abstract

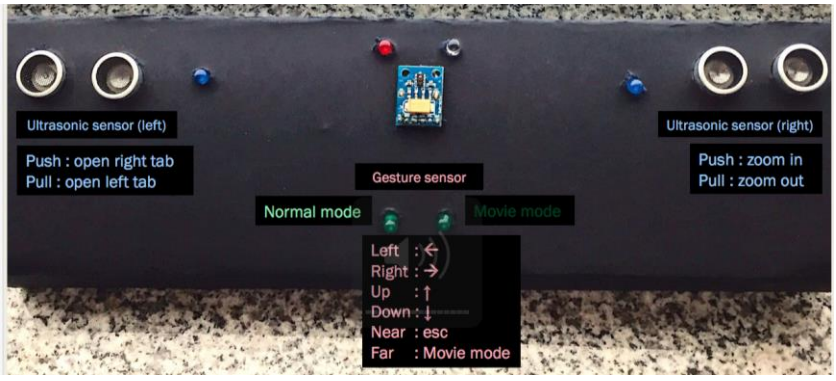
Traditional HID (Human Interface Devices) such as mouse and keyboards have become very popular and necessary in our daily lives. However, most of these items on the market are restricted in certain ways. For instance, a computer mouse is a hand-held device that detects two-dimensional motions relative to a surface. Those motions are typically translated into the motion of a pointer on a display, which allows smooth control of the graphical user interface. To improve the usability of the input device for more versatile circumstances, our group tried to design and develop an input device by using the given sensors: Adafruit LSM9DS1 Accelerometer, Gyro and Magnetometer sensors, HCSR-04 ultrasonic sensors and a motion detector sensor APDS-9960 from Sparkfun.

2. Implementation and Device Design

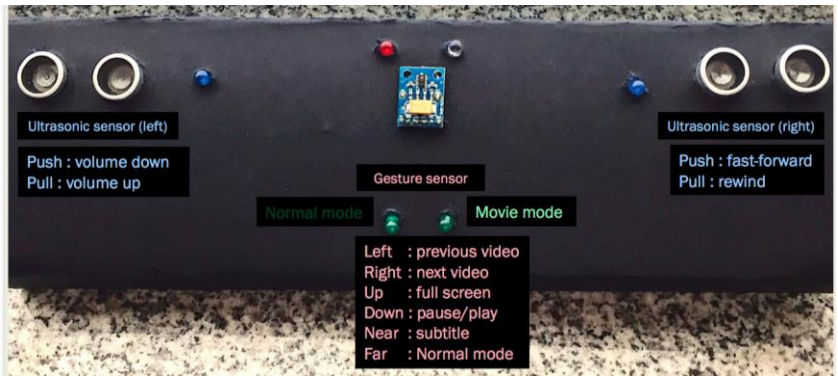
The initial project goal was to design a device that enables multiple smart mirror functions. Few weeks into the project, the smart mirror idea turned out to be too vague. It was costly and too difficult to manage the prototype. Therefore, we changed the project to create an on air-gyro-mouse and a remote media controller instead. The new objective is to enable long-distance computer control that could be used during a presentation or when using a projector.

Remote – Media- Controller

The remote media controller is a modification of our smart mirror idea. Since we have created some sort of a product in the first few weeks of the project, we changed the initial mirror and screen prototype into a wireless, box-shaped product instead.



Remote-media-controller and functions in Normal Mode



Remote-media-controller and functions in Movie Mode

There are two ultrasonic sensors mounted on the far left and right surface of the box, which collect the user's hand distance from the sensors in centimeters, per 500 milliseconds. This information is then transmitted through Bluetooth serial communication and received by the computer. Through a specific way, the user activates the device and a gentle and short hand push or pull would translate into functions like 'zoom in/out' or 'fast forward/rewind' depending on which sensor and which mode the user is in: 'normal mode' or 'movie mode'.

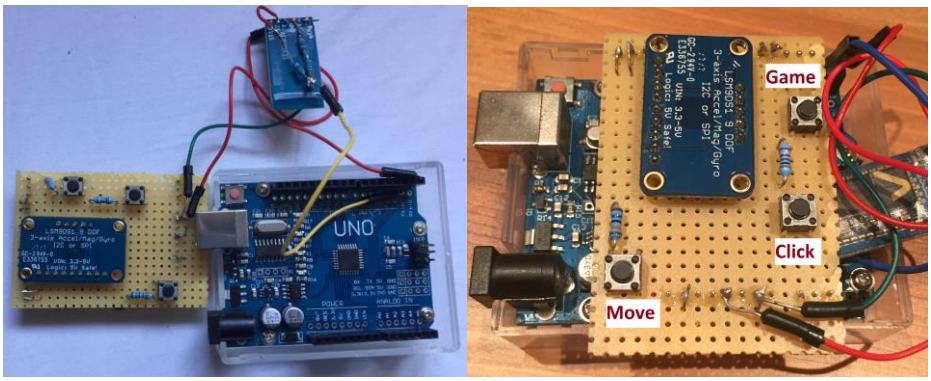
The same concept applies to a gesture sensor seen in the middle part of the box, which is always ready to read the user’s hand movements in six directions: up, down, left, right, near, far; and could control actions such as 'swipe', 'full screen on/off', et cetera.

We use the raw data obtained and experiment a lot with the sensor delays to ensure practicality and user's comfort. Another challenge was to come up with a way to process ultrasonic sensor readings. We limit the sensors from reading negative values and movements further than 40 centimeters away to avoid distortion. To eliminate data caused by random, unwanted objects, we make sure that the ultrasonic sensors will only be activated when the user purposely holds their hand between 15 to 20 centimeters away from the sensor for half a second first until the ‘active mode’ is on, before pushing or pulling it. As for the gesture sensor, we ignore motions that have an unclear direction or are too quick. This is necessary because the computer would have unavoidable delays in running the commands and we would want the sensors to wait until the first command detected is fully executed first before another motion could be read.

On-Air-Gyro Mouse

On-air-gyro-mouse was designed to work without any surface and simply through movements of the users' wrist. The gyroscope sensor, which is embedded into the microcontroller (Arduino Uno), will detect the angular velocity of users' wrist and is able to read the process by using the microcontroller. Due to the low accuracy and delays of the gyroscope module, we implemented a data-smoothing algorithm to acquire and process the data. The average gyroscope data from two axes, which are x and z axes based on the datasheet from the gyroscope sensors, will be calculated. Furthermore, by reading and analyzing the logging data, we are able to find the offset value of the original data.

The next task was to design a certain communication way to give the instructions that would move the cursor and execute the function such as left-click function. It was decided that we implement the Bluetooth serial communication to realize the cursor moving, right click functions and other functions.



On-air gyro mouse and its push buttons

To move the cursor on the monitor, the user only needs to press and hold the ‘move button’. To do a left-click, the ‘move button’ could first be released and then press the ‘click button’ on the desired area. While playing a game such as car-racing, the ‘game button’ could be pressed and held when the player wishes to do a left or right turning motion, as well as tilting the mouse to the sides imitating the motion of steering a car. The ‘move’ and ‘click’ buttons could also be pressed together once to activate a certain game function, for instance to activate the nitrous oxide engine that speeds up the player’s car.

In general, the Bluetooth module that we use in this project is the HC-05, an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for wireless serial connection setup. Compared with the other IOT method, Bluetooth is relatively much more power efficient and versatile.

The microprocessor Arduino Uno will be used as the data collection and process device and we implement several python libraries such as *pynput* and *pyserial* to receive the processed data and execute functions.

The key design of our projects:

- Relatively low cost, around 40 euro for both devices
- Devices are able to communicate with any other devices that has Bluetooth
- Easily customizable
- Feasible in a relatively long range

3. User Manual

Setting up:

- Install the latest Python3
- Install pip, the detailed installation guideline can be found and the relevant document can be downloaded from this link if the installed python version is Python 2 >= 2.7.9 or Python 3 > 3.4
- Use python pip to install two libraries: *pyserial* and *pynput*.
  For example: $\backslash pip\ install\ pyserial$
- Connect both the remote-media-controller and on air-gyro-mouse to two separate portable chargers
- Connect Bluetooth devices: smart_control, password is "hello" and RandomBot, password is "1234"
- Download the python files for each function, online repository link.
- Click the python files and allow them to run

4. Future Improvements and Developments

The process of designing and developing the device as well as the end results satisfy our initial expectations. However, there are some drawbacks to our projects, which can be improved in the future.

The remote media controller might be improved by using better sensors that have higher accuracies. Adding a small LCD on the device would also make the product better as it could notify the user what motion is being detected and which function is being executed, not merely through LEDs. The design could be more efficient if a microcontroller with much smaller size were to be used, along with a better box or case made of rigid and durable material. Other possible improvements are to get rid of the portable charger, to include a simple graphic user interface that allows users customize the sensitivity of their device according to preference (by modifying the sensor delays) or change the available functions to better suit their needs. A voice command ability would also greatly improve the controller as it replaces the action of typing on a computer keyboard.

The gyroscope and motion sensors readings obtained need future time and the readings values are also not precise enough to make unnoticeable delay to the end users. It can however be solved by implementing more sensors and by using an optimized algorithm. Furthermore, the design shield we use for the on-air-mouse might also be the source of disturbing signals, which might eventually deteriorate the accuracy of the device. This problem can, however, be easily solved by redesigning the PCB circus and implement a PCB board for our project. Lastly, to be able to find and evaluate the data we acquire from the sensors, it will be more precise and practical to use some data-logging tool to analyze the data.