

# Proyecto ELISA

## Especificaciones de diseño para puntero remoto y emulador de ratón de computadora

### Resumen del proyecto

Como se abordó en el documento ISR del proyecto, este pretende resolver la problemática de la carencia de un dispositivo confiable que funcione como un ratón remoto, que permite manejar y manipular contenido en una computadora de forma remota. En el documento SR se listaron las especificaciones y requerimientos del proyecto. En este documento se mostrará de manera esquemática cómo sería colocado el dispositivo en el usuario.

### Consideraciones iniciales para etapa pre-alfa

Para la etapa de desarrollo pre-alfa se considera únicamente la estabilización en la captura y lectura de datos usando un sensor y microcontrolador a este, conectados a la computadora. Se considera un éxito en esta prueba cuando es posible mover el puntero del ordenador de forma remota y con suavidad, evitando todo ruido e interferencia en el proceso.

Este proceso de captura y lectura de información se considera en dos partes:

- El proceso de captura, que consiste en cómo el microcontrolador maestro con el sensor procesará y enviará estos datos vía bluetooth hacia el microcontrolador esclavo.
- La otra parte es el proceso de lectura llevada a cabo por el microcontrolador esclavo, el cual hará uso de definiciones matemáticas para posicionar el puntero según los datos recibidos por el sensor desde el microcontrolador maestro.

Durante el proceso será necesario considerar el uso de librerías existentes dentro de la plataforma de Arduino, como SoftwareSerial.h y Mouse.h, además de considerar la escritura de librerías nuevas para agilizar el desarrollo matemático o mejorar la precisión del puntero. Algunas técnicas para el emparejamiento de bluetooth serán necesarias. Respecto al hardware, se recomienda usar placas de pruebas para los componentes usados.

Gran parte del proceso podría parecer complicado o engorroso, pero es necesario recalcar que gran parte de la información se encuentra en internet y que algunos de las soluciones a los problemas que se puedan presentar se resuelven en gran medida por el ingenio e inventiva de los desarrolladores.

## Del proceso de captura de información

Este proceso ocurre dentro del microcontrolador maestro, que es el que se encarga de capturar la información para enviarlo al microcontrolador esclavo. El módulo MPU6050 se conecta a un microcontrolador Arduino Nano o UNO que a su vez tiene un módulo Bluetooth HC-05. Los módulos Bluetooth deben ser configurados previamente para que se emparejen automáticamente en cuanto sean encendidos, esto se logra con un microcontrolador conectado al módulo y usando comandos AT, esto se puede ver a profundidad en la documentación del módulo y en tutoriales por internet. El módulo MPU6050 cuenta con acelerómetro y giroscopio, la información obtenida a partir de este son seis datos diferentes: tres datos de posición y tres datos de ángulo. Debe considerarse que en el módulo se encuentran especificados las direcciones a las que por estándar apunta el módulo. Además se deberán considerar el uso de filtros para la correcciones de errores en la lectura de datos. Los botones actúan de forma ordinaria dentro del sistema, usando booleanos para su activación, pero deben considerarse sólo cuando ya se ha logrado una captura y lectura de posición del cursor en el ordenador de forma óptima. La información capturada debe ser comunicada al microcontrolador conectado en la computadora usando el módulo Bluetooth, para este caso se sugiere revisar la librería SoftwareSerial. Los datos deben ser enviados de tal manera que sean consistentes, por ejemplo usando delimitadores.

En resumen, el microcontrolador maestro envía datos como se especifica en la tabla:

	Dato	Descripción
[0]	X	Componentes son las coordenadas de la mano del usuario con respecto del sistema de referencia ubicado en el pecho.
[1]	Y	
[2]	Z	
[3]	Rotation X	Ángulos de rotación alrededor de los ejes X, Y y Z con respecto al sistema de referencia ubicado en el pecho.
[4]	Rotation Y	
[5]	Rotation Z	
[6]	Enable	El valor de esta variable puede ser de uno o cero, en caso de ser uno la posición del puntero se actualiza constantemente con la calculada por los sensores y en caso de ser cero el puntero se queda fijo en pantalla sin importar los movimientos de la mano.
[7]	Right	El valor de esta variable puede ser de uno o cero, en caso de ser uno es registrado como un click derecho en la posición del puntero.

[8]	Left	El valor de esta variable puede ser de uno o cero, en caso de ser uno es registrado como un click izquierdo en la posición del puntero.
[9]	Scroll	El valor de esta variable puede ser de uno o cero, en caso de ser uno se activa la opción de scroll ya sea ascendente o descendente dependiendo del movimiento de la mano.

## Del proceso de lectura de información

Este proceso ocurre dentro del microcontrolador esclavo, que es el que se encarga de recibir la información y que está conectado a la computadora para posicionar el puntero en el ordenador. El microcontrolador recibe vía Bluetooth desde el microcontrolador maestro una cadena de datos, que son los descritos en la tabla de más arriba, y debe leerlos e interpretarlos de tal manera que sean usados para el cálculo numérico para obtener la posición del puntero. En este microcontrolador se desarrolla el proceso matemático para posicionar el puntero dentro de la pantalla. Este consiste en una transformación matemática de tres dimensiones a una de dos dimensiones. Se debe considerar el uso de matrices de rotación y traslación, esto requiere de una librería de matrices o álgebra lineal. Entre la información requerida para el proceso de localizar el puntero será necesario conocer una distancia ideal de la pantalla virtual, en centímetros, desde el punto de referencia, alto y ancho ideales, en centímetros (de preferencia con una relación de aspecto consistente), las dimensiones del monitor en pixeles y la densidad de estos en centímetros. Con el correspondiente cálculo se debe obtener un par de valores. Este par de valores deben ser usados como coordenadas para el puntero del ratón usando la librería Mouse.h para Arduino Leonardo. Se sugiere leer detenidamente la documentación para esta librería.

## Proyección

Este documento crecerá conforme se encuentre en las etapas de desarrollo. Los requerimientos seguirán extendiéndose, pero en general el estado actual debe dar una idea del funcionamiento básico del sistema.