

PROYECTO IRIS

RESUMEN

Proyecto *IRIS* es un prototipo de un periférico USB que busca representar imágenes mostradas en la pantalla de un computador, dirigido a personas con limitaciones visuales, quienes pueden interpretarlas a través del tacto. El sistema recibe una imagen que se reconstruye en la mano de la persona por medio de vibraciones inducidas por la interacción entre el campo magnético proveniente de los electroimanes de la malla del periférico y un guante receptor construido con imanes. La imagen para ser enviada al periférico sufre un proceso que consiste en la traducción de los colores que componen la imagen en pulsaciones de diferentes frecuencias de campos electromagnéticos.

PALABRAS CLAVES: Representación de imágenes, periférico, limitados visuales, campos electromagnéticos, USB (Bus Serial Universal).

ABSTRACT

The IRIS project is a USB peripheral prototype created to make a representation of images shown on a computer screen. It's oriented to people who have visual limitations, they are able to "see" images by using their sense of touch. The system's input is an image which is reconstructed in the person's hand through vibrations which are induced by the interaction between the magnetic field that comes from the electromagnets in the peripheral's grid and a glove that has magnets built into it. The image goes through a process that consists in the translation of the colors in it into different electromagnetic field frequency pulses.

KEYWORDS: Image representation, peripheral, visually challenged people, electromagnetic fields, USB (Universal Serial Bus).

1. INTRODUCCIÓN

Existe un tipo de afecciones inherentes a la naturaleza humana como son las limitaciones físicas, que pueden ser de nacimiento o adquiridas a lo largo de la vida de una persona, dentro de estas limitaciones se encuentra la visual, para este tipo de afección la medicina no tiene soluciones absolutas mas que la esperanza que puedan brindar operaciones muy costosas en el caso en que existan.

Los sistemas computacionales actuales se han volcado hacia la utilización del monitor como el medio principal de salida del computador, si se consideran todas las posibilidades que los ordenadores proveen hoy en día, teniendo en cuenta fenómenos como Internet y la existencia de periféricos que permiten ver televisión y utilizar cámaras para observar lo que ocurre alrededor del mundo; lograr que las personas con limitaciones visuales pudieran utilizar los computadores significaría reducir el enorme abismo social con el cual los han alejado del mundo, nuestro proyecto busca reducir dicha brecha.

La tecnología tanto software como hardware existente en la actualidad que sirve de soporte para las personas con limitaciones visuales tiene costos muy elevados para ser utilizados por una persona de un ingreso promedio en un país en vía de desarrollo.

El proyecto *IRIS* no tiene las limitaciones de técnicas como la *Braille* que está supeditada a la representación de textos planos, resultando imposible la representación de un documento con contenido gráfico; en cambio este proyecto buscará generalizar la representación de todo lo que se pueda mostrar en una pantalla de computador llegando al punto de poder mostrarle a la persona invidente o de baja visión todo lo que cualquier persona vidente con acceso a un computador puede observar, tal como lo es todo el contenido disponible en Internet, los libros y bibliotecas multimedia, hasta programas de videoconferencia y de captura de televisión, lo que representaría para las personas con limitaciones visuales, poder expandir sus posibilidades actuales.

2. APLICACIONES

Dentro de las aplicaciones del proyecto se encuentran la manipulación de imágenes a través de diferentes módulos desarrollados por software, como son:

- Módulo imagen: tiene la función de cargar una imagen y enviársela a la capa Demonio.
- Módulo pantalla: tiene la función de sacar *screen shots* (capturas de pantalla) periódicamente y enviarlos a la capa Demonio.

MARIA FERNANDA ZÚÑIGA ZABALA

Ingeniera de Sistemas y Computación
Docente auxiliar
Universidad Tecnológica de Pereira
mafe@utp.edu.co

JOHN ALEXIS GUERRA GUERRA

Ingeniero de Sistemas y Computación
Docente auxiliar
Universidad Tecnológica de Pereira
aguerra@utp.edu.co

FELIPE RESTREPO CALLE

Ingeniero de Sistemas y Computación
Docente auxiliar
Universidad Tecnológica de Pereira
pipe_res@utp.edu.co

- Módulo texto *Braille*: tiene la función de leer un archivo de texto y realizar la conversión de cada uno de sus caracteres a la imagen correspondiente en código *Braille* y enviarlas a la capa Demonio.
- Módulo TTY: utiliza el módulo texto *Braille* para representar el texto que aparece en una *tty* o terminal estándar de *GNU/Linux*.
- Módulo video: se encarga de enviar periódicamente *screen shots* de un archivo de un *stream* de video específico a la capa Demonio.
- Módulo Web: permite alternar entre texto y gráficos para poder navegar en los sitios web actuales con alto contenido multimedia.

De los módulos anteriormente mencionados, el modulo imagen esta completamente desarrollado, los demás corresponden a aplicaciones futuras del proyecto.

Otras aplicaciones futuras del proyecto son:

- Cambiar la conexión USB por una conexión inalámbrica.
- Reemplazar por completo un monitor convencional.
- Conectar el periférico a un *PDA* en conjunto con una *WebCam* y de esta forma, permitirle a una persona con limitaciones visuales percibir una representación del mundo exterior mientras se desplaza.

3. DISEÑO DEL SISTEMA

A continuación se muestra el diagrama de bloques del sistema:

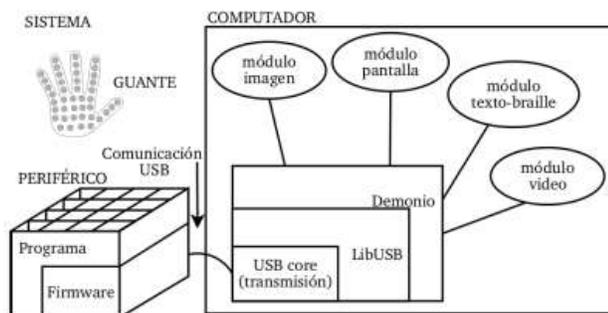


Figura 1. Diagrama de bloques.

El sistema esta compuesto básicamente por una parte hardware en el periférico y una parte software en el computador. En la Figura 1 se observan las diferentes capas del sistema en donde cada una de las capas tiene una responsabilidad específica.

3.1 EL COMPUTADOR

En el lado del computador se dividen las responsabilidades de procesamiento y comunicación mediante capas, teniendo en cuenta que las capas de la parte inferior son de más bajo nivel que las de la parte superior. Las capas, en orden descendente son: módulos software, demonio, LibUSB y USB core.

El software fue desarrollado para la plataforma GNU/Linux, aunque es portable para otras versiones libres de sistemas operativos tipo UNIX, como FreeBSD, OpenBSD y NetBSD.

3.1.1 Módulos Software

Los módulos software son aplicaciones que se encargan de entregarle imágenes para ser procesadas a la capa Demonio, éstas imágenes provienen de diferentes fuentes de información, como son: una imagen cualquiera, la pantalla completa del computador, una imagen de uno o varios caracteres codificadas en Braille, una secuencia de imágenes capturadas de un archivo de stream de video.

Para este prototipo se implementó únicamente el módulo imagen, ya que dicho módulo es la base de todos los demás. Éste módulo consiste en cargar un archivo de imágenes en cualquier formato y entregárselo a la capa demonio. Para la implementación de éste módulo se utilizó el lenguaje Python¹, la librería Python Image (PIL)² y la librería gráfica wxPython³.

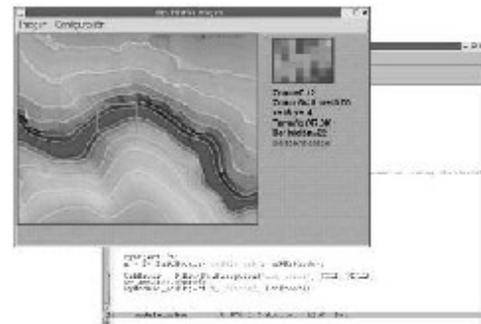


Figura 2. Módulo imagen.

3.1.2 Capa Demonio

Se trata de un programa que se está ejecutando todo el tiempo (servicio), y se encarga de la interfaz con el usuario, el procesamiento y codificación de imágenes que posteriormente serán enviadas al periférico, también tiene la función de interpretar todos los datos enviados por el periférico.

¹ <http://www.python.org>

² <http://www.pythonware.com/products/pil/>

³ <http://www.wxpython.org>

Para el prototipo, el demonio no se ejecutará como un servicio, sino como módulo que tiene que ser invocado para iniciar su ejecución.

El procesamiento de imágenes está compuesto por cuatro etapas: filtrado, codificación, multiplexación y finalmente entramado. Como se puede observar en la siguiente figura:

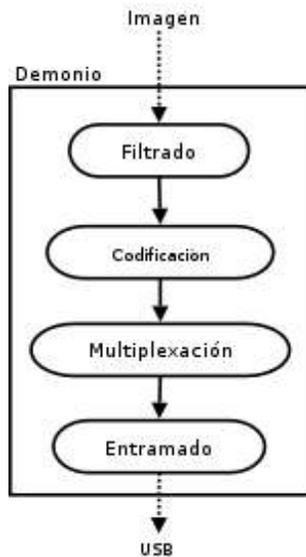


Figura 3. Procesamiento de datos.

3.1.2.1 Etapa de filtrado

En esta etapa, la imagen capturada es convertida a escala de grises, posteriormente es fraccionada de acuerdo a las dimensiones de la matriz de electroimanes (filas x columnas) que se encuentra en el periférico, donde el color de cada una de las fracciones es obtenido al aplicar un algoritmo de ponderación de imágenes estándar, existente en librerías de procesamiento de imágenes.

3.1.2.2 Etapa de codificación

La imagen, una vez convertida a escala de grises y fraccionada se ve internamente como una matriz numérica donde cada número corresponde a un valor de la escala de grises y se encuentra entre 0 y 255. En este punto, se realiza la codificación, dependiendo del valor de la definición con la cual se esté trabajando, por ejemplo, si se trabaja con una definición de 8 tonos, el periférico estaría configurado para representar únicamente 8 tonos de grises diferentes, asignando a cada valor de la escala de grises un tono de salida equivalente en una escala más pequeña, mientras que si la definición es 256, representaría todos los tonos de la escala de grises completamente. El valor de la definición es una variable parametrizable en el *software*.

Posteriormente, cada uno de los tonos de salida (cada una de las posiciones de la matriz numérica) es convertido a una cadena de caracteres de ceros y unos, la cual será traducida como un tren de pulsos que se le envía a uno de los elementos de la malla en el periférico en una etapa posterior, logrando de esta forma obtener diferentes frecuencias de campos electromagnéticos en los electroimanes de la malla.

3.1.2.3 Etapa de multiplexación

Hasta este momento se tienen n (en este caso 48) trenes de pulsos, uno para cada una de las fracciones de la imagen codificada. En esta etapa se busca multiplexar estos trenes de pulsos en el tiempo, armando un nuevo tren de pulsos de 48 bits donde el primer bit corresponde a uno de los bits del primer tren, el segundo bit corresponde a uno de los bits del segundo tren, y así sucesivamente, dependiendo del instante de tiempo en el que se encuentre. Como se muestra en la siguiente figura:

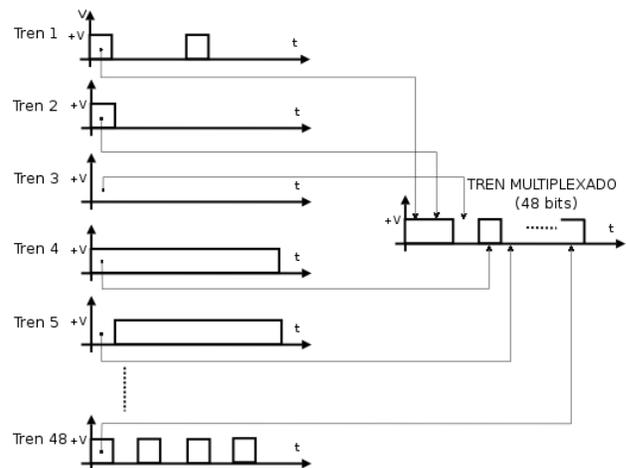


Figura 4. Etapa de multiplexación.

3.1.2.4 Etapa de entramado

En esta etapa se le entrega el tren de pulsos multiplexado a la librería *LibUSB*⁴, la cual se encarga de armar la trama de datos que va a ser enviada al periférico, donde el bloque de datos es el tren de pulsos multiplexado obtenido en la etapa anterior.

En el lado del computador, la librería *LibUSB* es la encargada de realizar todas las tareas que tengan que ver con la comunicación a través del puerto *USB*.

Para la implementación del demonio se utilizó el lenguaje Python y la librería Python Image (PIL).

⁴ <http://libusb.sourceforge.net>

3.1.3 Capa LibUSB

Es una librería que sirve como puente para la comunicación de las capas *USB core* y demonio. Contiene las principales funciones para acceder a dispositivos *USB* desde el espacio de trabajo del usuario, según lo indica la especificación *USB 2.0*⁵.

Se utilizó la librería LibUSB y las cabeceras de Python para generar un módulo dinámico del lenguaje, llamado módulo USB, para hacer llamados a la API de libUSB desde el lenguaje, ya que normalmente solo se tiene acceso a esta librería desde lenguaje C.

3.1.4 Capa USB Core

Es un módulo del sistema operativo *GNU/Linux* que permite la comunicación *USB*.

3.2 COMUNICACIÓN USB

La comunicación entre el periférico y el computador se hace a través del puerto *USB (Universal Serial Bus)*, debido a sus grandes virtudes y su popularidad en la actualidad.

El periférico es un dispositivo *low speed*, es decir, que trabaja a 1.5 Mbps y el tipo de transferencia de datos utilizado fue *interrupt transfer*.

3.3 PERIFÉRICO

El periférico está compuesto por una parte software que está embebida dentro de un microcontrolador y una parte hardware. La parte software del microcontrolador a su vez, se divide en dos capas, el firmware y el programa.

3.3.1 Firmware

Esta capa permite interpretar el protocolo *USB* mediante *software* embebido en el microcontrolador, ya que el microcontrolador seleccionado (Motorola HC08JB8) posee un módulo USB.

3.3.2 Programa

Es la capa de más alto nivel ubicada en el lado del microcontrolador. Es la aplicación final en el microcontrolador, encargada de interpretar la información enviada al periférico desde el computador y representarla tanto en la malla de electroimanes como en la de *leds*.

3.3.3 Hardware

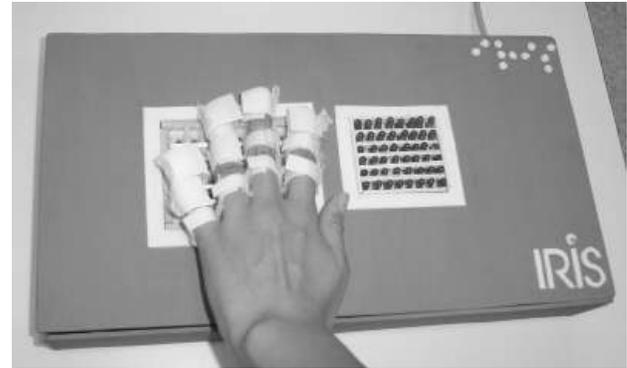


Figura 5. Fotografía del prototipo del periférico.

Está compuesto por tres módulos: módulo del microcontrolador, módulo de conversión serial a paralelo y memoria y el módulo malla. El módulo del microcontrolador, se encarga de recibir, interpretar los datos que se le envían desde el computador y finalmente enviarlos al módulo de conversión serial a paralelo y memoria; este último está compuesto por seis circuitos integrados 74LS259, los cuales son *latches* direccionables de 8 bits, donde cada uno de estos tienen la función de suministrarle señal a una fila de la malla y mantener esta señal hasta que reciba otro dato; el módulo malla, está compuesto por dos mallas conectadas en paralelo, una malla de *leds* y otra de electroimanes, donde la de *leds* sirve para realizar pruebas de funcionalidad con personas videntes, mientras que la malla de electroimanes, es para las personas con limitaciones visuales. Cada uno de los elementos de la malla de electroimanes es un circuito de potencia.

3.4 GUANTE

Es el accesorio que actúa como sensor para que el usuario pueda percibir las señales enviadas por los electroimanes, es necesario debido a que el cuerpo humano no es susceptible a campos magnéticos, éste interactúa con la malla de electroimanes de tal forma que las señales son percibidas como pulsaciones de campos magnéticos con diferentes frecuencias, es así, que se logra establecer diferencias de colores, de acuerdo a la frecuencia de las pulsaciones, obteniendo como resultado, al percibir las señales de la malla en conjunto, establecer diferencias de formas y colores.

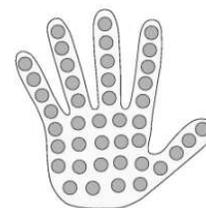


Figura 6. Guante.

⁵ <http://www.usb.org>

4. PRUEBAS

Las pruebas realizadas con el prototipo funcional arrojaron resultados inesperados, como una clasificación importante para el uso del dispositivo de acuerdo a la naturaleza de la limitación visual. Se encontró mayor dificultad de aprendizaje para personas invidentes de nacimiento ya que no poseían ninguna representación del mundo a diferencia de las personas que adquirieron la discapacidad a lo largo de sus vidas quienes podían asociar con mayor facilidad las imágenes.

5. RECOMENDACIONES

Para futuros avances en el proyecto es recomendable que se hagan nuevas investigaciones que permitan encontrar un diseño ergonómico tanto del periférico como del guante para personas con limitaciones visuales, además que cubra factores de diseño industrial al igual que médicos entre otros.

El prototipo debe evolucionar aumentando el número de electroimanes por área cuadrada, disminuyendo el área transversal del núcleo de los electroimanes y reduciendo las secciones de la malla en las cuales no se emite señal, de esta manera se permitan escalar el hardware del periférico y mejorar la definición de los datos. Otra posibilidad que debe ser evaluada para mejorar la escalabilidad es la utilización de dispositivos lógicos programables en vez de microcontroladores.

En desarrollos futuros, se recomienda que el núcleo de procesamiento de imágenes y de conexión al periférico, sea separado en un demonio que corra permanentemente en el sistema (servicio), escuchando peticiones de los módulos software.

6. CONCLUSIONES

Es un avance hacia la independencia de las personas con limitaciones visuales.

Se presenta una alternativa para países en vía de desarrollo.

Se realizó completamente una codificación de gráficos.

Se construyó un prototipo completamente funcional.

Queda trazado el camino hacia la eliminación de la brecha digital permitiendo que personas con limitaciones visuales accedan de una manera mas equitativa a a los beneficios tecnológicos que nos brinda la humanidad.

7. BIBLIOGRAFÍA

- [1] M68HC08 Microcontrollers: MC68HC08JB8 Technical Data (http://www.freescale.com/files/microcontrollers/doc/data_sheet/MC68HC908JB8.pdf), Motorola. 2003.
- [2] MALVINO, Albert Paul, Principios de electrónica, 6ed. España, McGraw-Hill Interamericana S.A. 2000. 1110 p.
- [3] Massachusetts Institute Technology. Dept. of Electrical Engineering, Circuitos magnéticos y transformadores. Estudio amplio de las características físicas de los circuitos magnéticos y de su aplicación al cálculo y diseño, 1965, Editorial Reverte S.A. 697p.
- [4] RAMOS RAMOS, Guillermo, Visualizador de matriz de puntos. Electrónica y Computadores. No. 69, CEKIT S.A. 2000. p 14-19.
- [5] Sitio web oficial de LibUSB (<http://libusb.sourceforge.net>).
- [6] Sitio web oficial de Python (<http://www.python.org>).
- [7] USB Specification 2.0 (<http://www.usb.org>).