

Sistema de movimiento del cursor del computador para personas con discapacidad

Computer mouse movement for persons with disabilities

Aguirre, Iñaki* y Amaro, Manuel

Especialización y Maestría en Ingeniería de Control y Automatización, Escuela de Ingeniería de Sistemas,
Facultad de Ingeniería, Universidad de Los Andes,
Mérida 5101, Venezuela
*iaguirre@ula.ve

Resumen

El presente trabajo describe el modelado y la construcción de un sistema que ayude a personas con discapacidad a interactuar con el computador. El sistema se modela mediante Redes de Petri y permite la ejecución del movimiento del cursor del computador mediante la presión de la tecla alt o del botón central del ratón. Adicionalmente cuenta con una ventana de configuración de la velocidad de refrescamiento de las imágenes, la velocidad del movimiento del cursor y de la selección de la tecla o botón a utilizar.

Palabras clave: Redes de Petri, aplicación, ayuda a personas con discapacidad.

Abstract

This paper describes a system modeling and construction that helps people with disabilities to interact with the computer. Petri nets are used to model the system. The system executes the computer cursor movement by pressing the alt key or middle mouse button. The system has a configuration window that allows determining the refreshing images speed, the cursor movement velocity and selection of the use of the alt key or the middle button.

Key words: Petri nets, applications, aid to disable people.

1 Introducción

En la actualidad existen en el mercado diversos dispositivos para la ayuda a personas con discapacidad. La utilización de estos diversos dispositivos, por parte de los usuarios, les permite tener cierto grado de autonomía para realizar algunas tareas, trayendo como consecuencia mayor independencia y mayor autoestima por parte de los usuarios. Existen diversos tipos de discapacidad, unas más severas que otras pero a todos los usuarios se les deberían proporcionar diferentes opciones de dispositivos para que escojan el que mejor se adapta a cada uno de ellos.

El propósito de este artículo es describir como se modela y se construye una nueva herramienta computacional, que permite al usuario con discapacidad interactuar de una manera sencilla, mediante la presión de un botón, con el computador.

2 Modelado del Sistema

Las Redes de Petri son una herramienta matemática

que puede utilizarse para el modelado de sistemas de diversa naturaleza. Desarrolladas por Carl Adam Petri en su tesis doctoral como un modelo de propósito general, para la descripción de las relaciones existentes entre condiciones y eventos.

Una Red de Petri (RdP) es un grafo orientado formado por dos tipos de nodos, los lugares y las transiciones, unidos alternativamente por arcos orientados. Ver Fig. 1.

Lugar: Representa un estado al que puede llegar el sistema.

Transición: Indica la posibilidad que ocurra un evento que altere el estado del sistema. Se activa una entrada, se detecta una pieza, se cumple una condición, etc.

Arco: Se utilizan para unir los estados con transiciones y transiciones con estados. Se representan por líneas rectas dirigidas.

Disparo de Transiciones: Una transición está activa si están marcados todos sus lugares de entrada y se cumple la condición asociada. El disparo de una transición supone el quitar una marca de cada uno de los lugares de entrada y añadir una marca a todos los lugares de salida. Un lugar

puede poseer un número positivo o nulo de marcas. Una marca se representa mediante la colocación de un punto en el interior del lugar (Aguirre, 2011).

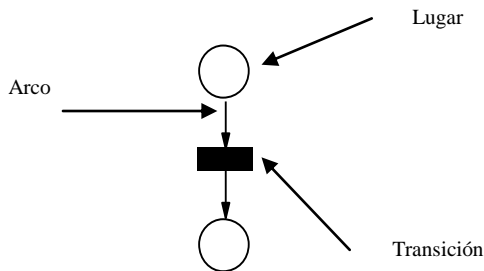


Fig 1: Elementos de una Red de Petri

El conjunto de marcas asociadas a una Red de Petri en un instante dado, constituye el marcado de la Red de Petri en ese instante (García, 2004).

3 Modelado del Sistema mediante Redes de Petri

En este apartado se muestra el modelado del sistema junto con la descripción correspondiente de los estados y transiciones. Adicionalmente se muestran las tablas de las variables utilizadas y la descripción del movimiento. El sistema de modelado está basado en los trabajos (Justo, 2009), (Justo y Aguirre, 2009), (Amaro, 2010).

3.1 Descripción de variables utilizadas en el modelado del sistema:

Tabla 1. Variables utilizadas en el modelo

Variable	Descripción
tecla	Define la tecla o el botón que permite la activación del movimiento del cursor. (Su valor puede ser cambiado desde el panel de configuración). Sus valores posibles son tecla = Tecla ALT o tecla = Botón Central del Ratón
conmutador	Variable entera que detecta el movimiento del cursor. Sus valores pueden ser 1 (cursor en movimiento) o 0 (cursor detenido).
band	Variable entera que permite esperar 1000ms para comenzar a mover el ratón (evitar fallo de arrastre de objetos). Se utiliza para detectar que la tecla denotada por tecla fue presionada con la finalidad de comenzar a mover el cursor, y no con la finalidad de detener su movimiento. Sus valores pueden ser 1 (tecla = presionada y cursor detenido) o 0 (tecla = no presionada y cursor en movimiento).
valor_pixel	Variable entera que denota el movimiento mínimo que realiza el cursor en cada iteración del Temporizador 1, el cual refleja a su vez la velocidad de movimiento del cursor. Su valor puede ser cambiado desde el panel de configuración. Su intervalo de valores es [1,10].
imagen_variable	Variable entera que permite la activación y desactivación de las imágenes de forma continua. Su intervalo de valores es [1,6]

Tabla 2. Descripción de "Dirección D"

Imagen	Dirección	Número de Imagen (D)
	Abajo	1
	Arriba	2
	Izquierda	3
	Derecha	4
	Click	5
	Doble Click	6

3.2 Estados del sistema:

Tabla 3. Estados del sistema

Nombre	Descripción
m1	Inicializar variables, cargar imágenes y formularios de la aplicación
m2	Iniciar Temporizador 1
m3	Leer valores de <i>tecla</i> , <i>conmutador</i> , <i>band</i> e iniciar Temporizador 2
m4	Detener Temporizador 2
m5	Hacer <i>band=1</i>
m6	Leer valor de <i>band</i>
m7	Esperar 1000ms / hacer <i>band=0</i> / hacer <i>tecla = no presionada</i> / hacer <i>conmutador=1</i>
m8	Esperar 20ms
m9	Tomar posición del cursor
m10	Mover cursor un <i>valor_pixel</i> hacia la dirección <i>D</i>
m11	Leer valores de <i>tecla</i> y <i>conmutador</i>
m12	Hacer <i>conmutador=0</i>
m13	Esperar 20ms (Temporizador 1)
m14	Detener Temporizador 2
m15	Esperar 1000ms
m16	Leer posición del cursor
m17	Hacer <i>botón = presionado</i>
m18	Esperar 10ms
m19	Hacer <i>botón = no presionado</i>
m20	Esperar 300ms
m21	Estado que repite m17, m18, m19 y m20 (solo se activa cuando imagen D = 6)
m22	Iniciar Temporizador 2
m23	Leer valor de <i>imagen_variable</i>
m24	Activar imagen 1 y desactivar las demás.
m25	Hacer <i>imagen_variable=2</i>
m26	Activar imagen 2 y desactivar las demás.
m27	Hacer <i>imagen_variable=3</i>
m28	Activar imagen 3 y desactivar las demás.
m29	Hacer <i>imagen_variable=4</i>
m30	Activar imagen 4 y desactivar las demás.
m31	Hacer <i>imagen_variable=5</i>
m32	Activar imagen 5 y desactivar las demás.
m33	Hacer <i>imagen_variable=6</i>
m34	Activar imagen 6 y desactivar las demás.
m35	Hacer <i>imagen_variable=1</i>
m36	Esperar <i>tiempo</i> ms (Temporizador 2)

3.3 Transiciones del sistema:

Tabla 4. Transiciones del sistema

Nombre	Descripción
T0	Variables inicializadas, imágenes y formularios cargados.
T1	Temporizador 1 iniciado
T2	<i>tecla = no presionada</i>
T3	<i>band = 1</i>
T4	$(Tecla = presionada \ \& \ imagen(D) = activada) \ \ (conmutador = 1)$
T5	Temporizador 2 detenido
T6	<i>band=1</i>
T7	<i>band = 0 & tecla = no presionada & (Se ha esperado 1ms)</i>
T8	<i>band=0</i>
T9	Se ha esperado 20ms
T10	Posición del cursor tomada
T11	Posición del cursor escrita
T12	<i>tecla= presionada & conmutador = 1</i>
T13	<i>tecla = no presionada</i>
T14	<i>conmutador = 0</i>
T15	Tecla = presionada & imagen(D) = activa
T16	Temporizador 2 detenido
T17	Se ha esperado 1000ms
T18	Posición del cursor tomada
T19	Tecla = presionada
T20	Se ha esperado 10ms
T21	Tecla = arriba
T22	Se ha esperado 300ms
T23	1
T24	Se ha esperado 20ms (Temporizador 1)
T25	Temporizador 2 iniciado
T26	Imagen_variable = 1
T27	Imagen D = activa, imágenes (Todas - D) = desactivadas
T28	Imagen_variable = 2
T29	Imagen D = activa, imágenes (Todas - D) = desactivadas
T30	Imagen_variable = 3
T31	Imagen D = activa, imágenes (Todas - D) = desactivadas
T32	Imagen_variable = 4
T33	Imagen D = activa, imágenes (Todas - D) = desactivadas
T34	Imagen_variable = 5
T35	Imagen D = activa, imágenes (Todas - D) = desactivadas
T36	Imagen_variable = 6
T37	Imagen D = activa, imágenes (Todas - D) = desactivadas
T38	1
T39	Se ha esperado <i>tiempo_1</i> (Temporizador 1)

Una vez definidas las tablas, mostradas anteriormente, se procede a la construcción de la Red de Petri compuesta por los estados y transiciones. El sistema modelado muestra el comportamiento del sistema con todas las opciones que este posee. En la Fig. 2 se muestra la Red de Petri del sistema de movimiento del cursor de la computadora para personas con discapacidad (Amaro, 2010).

Mediante la descripción del sistema se puede seguir de manera lógica y ordenada el funcionamiento del sistema (Moody y Antsaklis, 1988). El sistema resultante es un sistema sencillo, robusto y fácil de implementar en el computador.

Desarrollos de aplicaciones para personas con discapacidad han sido presentados por los autores en trabajos pre-

vios utilizando las Redes de Petri como herramienta de modelado (Aguirre et al., 2011).

3.4 Red de Petri del Sistema

La Red de Petri resultante del modelo se muestra en la Fig. 2.

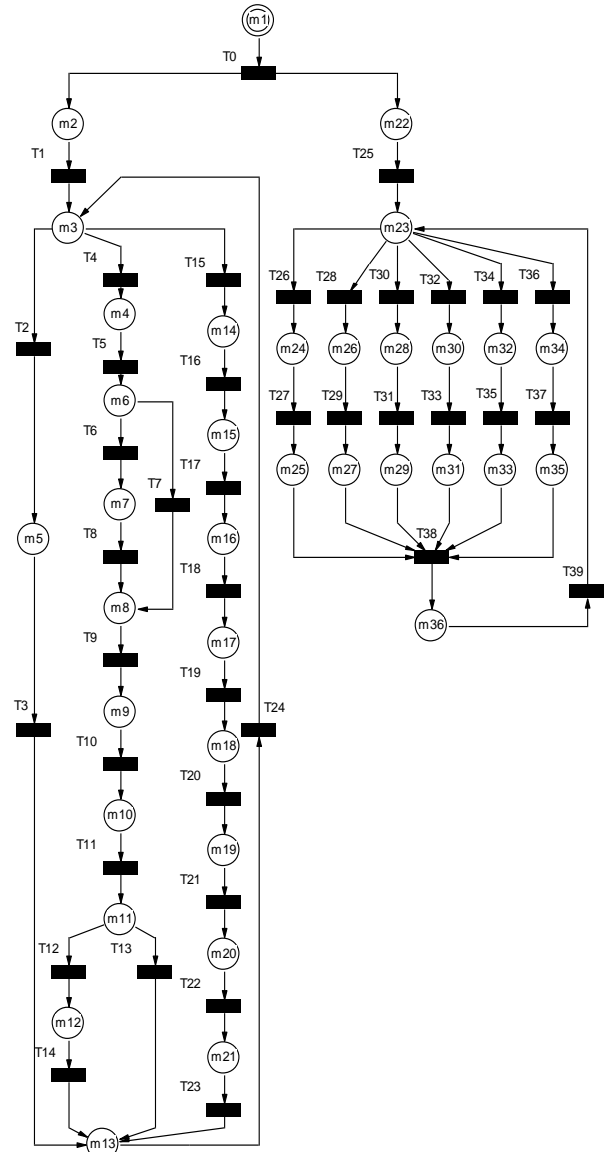


Fig 2: Red de Petri del sistema modelado

4 Implementación del Sistema

Las personas que poseen una discapacidad muy severa, producto de una tetraplegia, en muchos casos solo pueden mover partes muy específicas de su cuerpo como por ejem-

plo: alguno de sus dedos de las extremidades. El crear un sistema computacional, basado en el movimiento reducido de las personas que posean esta discapacidad y sean capaces de presionar una tecla o un botón, les permite interactuar con el computador de una manera sencilla y eficiente. De esta manera los usuarios pueden realizar diversas tareas con el computador que abarcan desde expresarse mediante un teclado virtual y un editor de texto, hasta realizar actividades de distracción, de estudio o de trabajo.

Esta aplicación tendría un efecto muy positivo sobre los usuarios incrementando su autoestima y permitiéndoles desarrollar tareas que debido a su discapacidad no pueden llevar a cabo. Adicionalmente, se les abre una ventana para interactuar con otras personas a través del computador.

El sistema modelado mediante Redes de Petri ha sido implementado siguiendo la lógica propuesta en el algoritmo y ha sido implementado mediante el lenguaje de programación de alto nivel Visual Basic.

5 Resultados

El producto final es un programa de computación que al ejecutarlo genera una ventana siempre visible que se ubica en el lado inferior derecho del monitor. En esta ventana se muestra una de seis imágenes que se corresponden con los movimientos hacia la izquierda, derecha, arriba, abajo, un clic y doble clic. Esta imagen varía cada periodo de tiempo siendo cíclico. Al seleccionar una de estas imágenes, bien sea presionando la tecla alt o mediante el botón central del ratón, el cursor del computador se moverá en la dirección solicitada. El sistema en cuestión vendría siendo un sistema indexado, que con un solo grado de libertad, se pueden realizar seis tareas diferentes. Ver Fig. N° 2.

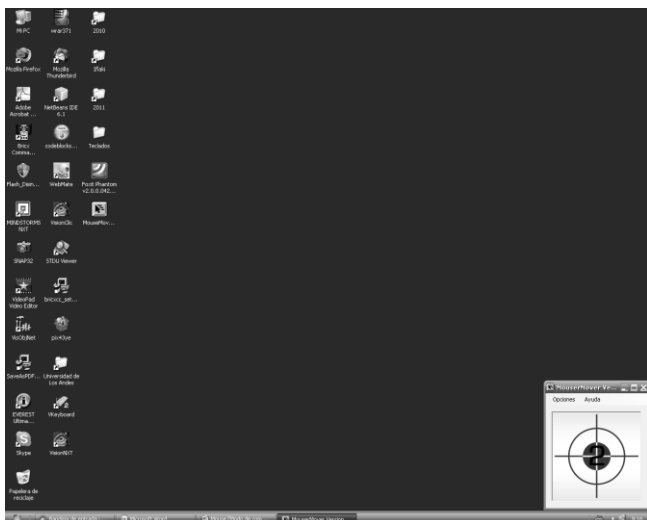


Fig 2: Ubicación de la ventana resultante al ejecutar el programa

El sistema cuenta con una ventana de configuración que facilita y personaliza la aplicación en función de la dis-

capacidad y las funcionalidades del usuario.

Ventana de configuración: Esta ventana muestra los parámetros de configuración del sistema. Entre los cuales se encuentran calibrar la velocidad de refrescamiento de las imágenes que se revelan en la ventana situada en el extremo inferior derecho del monitor del computador, la selección de la velocidad con que se mueve el cursor del computador y el botón o tecla alt para realizar las operaciones sobre el cursor del computador. Lo anteriormente descrito se muestra en la ventana de configuración de la Fig. N° 3.



Fig 3: Ventana de configuración del sistema



Fig 4: Utilización de la tecla *alt* del teclado



Fig 5: Utilización del botón central del ratón

Una vez configurado el sistema, de acuerdo al usuario en cuestión, el sistema comienza a refrescar cada cierto tiempo las imágenes de opción de movimiento del cursor y de realizar un clic o doble clic.

Movimiento hacia la izquierda: al seleccionar esta opción, presionando la tecla alt o el botón central del ratón, el cursor del computador se moverá a la izquierda mientras se tenga presionada esta opción o cuando se presione nuevamente la opción. La ecuación empleada para realizar este movimiento es la siguiente:

$$\text{PosCursor}(X,Y) = (\text{PosCursor}.X - \text{valor_pixel}, \text{PosCursor}.Y)(1)$$

Donde PosCursor.X y PosCursor.Y representan la posición X, Y del cursor respectivamente. Por otro lado, la variable valor_pixel, representa la cantidad de pixeles que se moverá el cursor cada 20 ms (tiempo del temporizador interno del proceso). Esta descripción se aplica de la misma manera para las ecuaciones 2, 3 y 4.



Fig 4: Ventana de movimiento a la izquierda

Movimiento hacia la derecha: al seleccionar esta opción, presionando la tecla alt o el botón central del ratón, el cursor del computador se moverá a la derecha mientras se tenga presionada esta opción o cuando se presione nuevamente la opción. El movimiento se realiza mediante la aplicación de la siguiente relación:

$$\text{PosCursor}(X,Y) = (\text{PosCursor}.X + \text{valor_pixel}, \text{PosCursor}.Y)(2)$$



Fig 5: Ventana de movimiento a la derecha

Movimiento hacia arriba: al seleccionar esta opción, presionando la tecla alt o el botón central del ratón, el cursor del computador se moverá hacia arriba mientras se tenga presionada esta opción o cuando se presione nuevamente la opción. La posición del cursor se mueve mediante la ecuación (3):

$$\text{PosCursor}(X,Y) = (\text{PosCursor}.X, \text{PosCursor}.Y + \text{valor_pixel})(3)$$



Fig 6: Ventana de movimiento hacia arriba

Movimiento hacia la abajo: al seleccionar esta opción, presionando la tecla alt o el botón central del ratón, el cursor del computador se moverá hacia abajo mientras se tenga presionada esta opción o cuando se presione nuevamente la opción. La relación empleada para realizar el movimiento del cursor está dada por (4).

$$\text{PosCursor}(X,Y) = (\text{PosCursor}.X, \text{PosCursor}.Y - \text{valor_pixel})(4)$$



Fig 7: Ventana de movimiento hacia abajo

Clic del ratón: al seleccionar esta opción, presionando la tecla alt o el botón central del ratón, se hace la operación equivalente a realizar un clic en el ratón o presionar la tecla Enter.



Fig 8: Ventana de clic

Doble clic del ratón: al seleccionar esta opción, presionando la tecla alt o el botón central del ratón, se hace la operación equivalente a realizar un doble clic en el ratón.



Fig: 9. Ventana de doble clic

Haciendo uso de estos seis movimientos prefijados por la aplicación, el usuario final puede interactuar de una manera sencilla y eficaz con el computador. Ya sea para actividades de formación, de expresión o de distracción.

6 Conclusiones

En el presente trabajo se muestra una aplicación de computación muy útil para personas con discapacidad, en particular aquellas personas con tetraplegia. El sistema desarrollado está modelado utilizando Redes de Petri e implementado mediante un lenguaje de programación de alto nivel como es el Visual Basic.

El producto final es una aplicación software que permite a personas con poca movilidad interactuar con el computador mediante la acción de presionar un botón o una tecla. No obstante, al permitirle a los usuarios de esta aplicación interactuar con el computador se les amplía el ámbito de comunicación ya que pueden a través de un teclado virtual, ubicado en la pantalla del computador, escribir y poder expresarse.

Adicionalmente se pueden realizar tareas de distracción, de estudio o de trabajo, elevando notablemente su calidad de vida e incrementando su autoestima.

El sistema permite configurar la tecla alt o el botón central del ratón para ejecutar el movimiento del cursor. El sistema también permite, de manera sencilla, configurar la velocidad de refrescamiento de las imágenes que indican el posible movimiento a realizar y la velocidad con que se mueve el cursor.

El sistema es versátil, útil y que ocupa muy poca memoria y adicionalmente tiene una buena aceptación por parte de los usuarios.

Referencias

- Aguirre I, 2011, Análisis y Descripción de Técnicas de Automatización, Talleres Gráficos de la ULA, Mérida, Venezuela.
- Aguirre I, Justo F, Amaro M, Ramírez A, 2011, Descripción de Tres Aplicaciones de bajo costo para Ayuda a Personas con Discapacidad, Revista Ciencia e Ingeniería, Vol. 32, No. 1, 2011.
- Amaro M, 2010, Diseño e Implementación de un Sistema que permita el control de la trayectoria de un vehículo, a través de un Sistema de Visión Artificial, Proyecto de Grado, Facultad de Ingeniería, Universidad de Los Andes, Mérida, Venezuela.
- Justo F y Aguirre I, 2009, Creación de una Herramienta que Permita Mover el Cursor de un Computador a Partir del Movimiento Ocular, Utilizando Técnicas de Visión Artificial, Seventh Latin American and Caribbean Conference for Engineering and Technology, LACCEI 2009, San Cristóbal, Venezuela.
- Justo F., 2009, Creación de una Herramienta que Permita Mover el Cursor de un Computador a Partir del Movimiento Ocular, Utilizando Técnicas de Visión Artificial, Proyecto de Grado, Facultad de Ingeniería, Universidad de Los Andes, Mérida, Venezuela.
- García E, 2004, Automatización de Procesos Industriales. Alfa Omega. España.
- Moody J y Antsaklis P, 1988, Supervisory Control of Discrete Event Systems using Petri Nets. Publicado por Kluwer Academic Publishers. Norwell, MA. Estados Unidos.

Recibido: 06 de julio de 2011

Revisado: 15 de noviembre de 2011

