

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA QUE ACTIVA DISPOSITIVOS
ELECTRÓNICOS MEDIANTE SEÑALES OCULARES PARA MEJORAR LA
COMUNICACIÓN A PERSONAS CON DISCAPACIDAD MOTORA

JOSE RICARDO MUÑOZ ROSERO
JORGE ARMANDO VILLARREAL BUSTOS

UNIVERSIDAD DE NARIÑO
FACULTAD DE INGENIERIA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
SAN JUAN DE PASTO
2012

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA QUE ACTIVA DISPOSITIVOS
ELECTRÓNICOS MEDIANTE SEÑALES OCULARES PARA MEJORAR LA
COMUNICACIÓN A PERSONAS CON DISCAPACIDAD MOTORA

JOSE RICARDO MUÑOZ ROSERO
JORGE ARMANDO VILLARREAL BUSTOS

Trabajo de grado presentado como requisito parcial para optar por el título de
Ingeniero Electrónico

Asesor del proyecto:
ROLANDO BARAHONA CABRERA
Ingeniero electrónico

UNIVERSIDAD DE NARIÑO
FACULTAD DE INGENIERIA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
SAN JUAN DE PASTO
2012

NOTA DE RESPONSABILIDAD

“La Universidad de Nariño no se hace responsable por las opiniones o resultados obtenidos en el presente trabajo y para su publicación priman las normas sobre el derecho de autor”.

Acuerdo 1. Artículo 324. Octubre 11 de 1966, emanado del honorable Consejo Directivo de la Universidad de Nariño.

Nota de aceptación:

Firma del Presidente del Jurado

Firma del Jurado

Firma del Jurado

San Juan de Pasto, Marzo de 2012.

CONTENIDO

	pág.
INTRODUCCIÓN	19
1. TEMA	20
1.1 TÍTULO	20
1.2 MODALIDAD	20
1.3 LÍNEA DE INVESTIGACIÓN	20
2. DESCRIPCIÓN DEL PROBLEMA	21
2.1 PLANTEAMIENTO	21
2.2 FORMULACIÓN	21
2.3 SISTEMATIZACIÓN	21
3. OBJETIVOS	22
3.1 OBJETIVO GENERAL	22
3.2 OBJETIVOS ESPECÍFICOS	22
4. JUSTIFICACIÓN	23
5. MARCO TEÓRICO	24
5.1 LOS MÚSCULOS	24
5.1.1 Contracción muscular	25
5.1.2 Músculos de la cara	25
5.1.2.1 Músculo Orbicular de los párpados	25
5.2 Micrófono	26

5.3 Tarjeta de sonido	27
5.4 EL OJO	29
5.4.1 Músculos propios del ojo	30
5.4.2 Estructuras protectoras	30
5.5 PROCESAMIENTO DE DATOS	31
5.5.1 Filtros digitales	31
5.5.2 Transformadas	32
5.5.3 Procesamiento de imágenes	32
5.5.3.1 Proceso de filtrado	32
5.5.3.2 Posición de la pupila del ojo	35
5.6 INTERFAZ GRÁFICA	36
5.6.1 Tipos de interfaces gráficas de usuario	37
5.6.1.1 Gui's y zooming user interfaz	37
5.6.1.2 Touchscreen user interface	37
5.6.1.3 Interfaz Natural de Usuario	37
5.6.2 Guide de MATLAB	38
5.7 USB	40
5.7.1 Automatización de datos Enumeración USB v 3.1	42
5.8 RADIOFRECUENCIA	43
5.8.1 Clasificación	43
5.8.2 Receptores de RF	44
5.9 ELECTRÓNICA DE POTENCIA	45

5.9.1 Dispositivos Semiconductores de Potencia	45
5.10 ANTECEDENTES	46
6. DESARROLLO	49
6.1 ADQUISICIÓN DE SEÑALES MIOGRÁFICAS	49
6.1.1 Sensores	50
6.1.2 Adquisición	52
6.2 ALGORITMO DE UBICACIÓN DE LOS OJOS	53
6.2.1 Captura de imágenes	55
6.2.2 Ajustes de imagen para encontrar el ojo	55
6.2.3 Limites de figuras	57
6.2.4 Filtros	57
6.2.5 Determinación del objeto de área máxima	57
6.2.6 Determinación de coordenadas	58
6.2.7 Ubicación de los puntos de la imagen	58
6.2.8 Determinación de la posición de los ojos	59
6.3 DESARROLLO DE LA INTERFAZ GRÁFICA	60
6.3.1 Programación	61
6.3.1.1 Determinación de un parpadeo	63
6.4 INTERFAZ USB	63
6.4.1 Diseño del hardware	64
6.4.2 Programación del Firmware	66
6.4.2.1 Generación de las librerías usb_cdc.h y usb_desc_cdc.h	67

6.4.2.2 Generación del driver	69
6.4.3 Programación en el software	69
6.5 ETAPA DE RADIOFRECUENCIA	70
6.6 ETAPA DE POTENCIA	72
7. RESULTADOS	74
7.1 SOFTWARE DE DETECCIÓN DE POSICIÓN DE LA PUPILA	74
7.2 SEÑALES DE PARPADEO	76
7.3 SISTEMA USB	76
7.4 MENÚ GRÁFICO	78
7.5 CIRCUITOS DE RADIOFRECUENCIA	79
7.6 CIRCUITOS DE POTENCIA	80
8. CONCLUSIONES	82
9. RECOMENDACIONES	83
BIBLIOGRAFÍA	84
ANEXOS	86

LISTA DE TABLAS

	Pág.
Tabla 1. Características de las bandas espectrales	43
Tabla 2. Detalles de los resultados del procesamiento de imágenes	74

LISTA DE FIGURAS

	Pág.
Figura 1. Diagrama de una fibra muscular	24
Figura 2. Ubicación del músculo orbicular del parpado	25
Figura 3. Micrófono de condensador electret	26
Figura 4. Componentes de una tarjeta de sonido	28
Figura 5. Puertos de una tarjeta de sonido estándar	28
Figura 6. Partes del ojo	29
Figura 7. Músculos del ojo	30
Figura 8. Ejemplo de filtrado en el dominio de la frecuencia	33
Figura 9. Ejemplo de filtrado promedio con filtro 5x5	34
Figura 10. Imagen original y resulta tras filtros Laplaciano y Sobel	35
Figura 11. Puntos más importantes del Ojo	36
Figura 12. Interfaz touchscreen	37
Figura 13. Interfaz natural	38
Figura 14. Cuadro de dialogo de la interfaz gráfica	38
Figura 15. Interfaz gráfica en blanco	39
Figura 16. Características de los pines y conectores de USB	31
Figura 17. Captura del programa Enumeración de datos USB	42
Figura 18. Módulos Rf Transmisor y receptor a 433 MHz	45
Figura 19. Esquema de un relé de tipo armadura	46

Figura 20. Diagrama de bloques del sistema	50
Figura 21. Sensor para la captación del parpadeo	51
Figura 22. Ubicación del sensor	51
Figura 23. Canales de un plug estereofónico	52
Figura 24. Dos parpadeos consecutivos	52
Figura 25. Cinco parpadeos consecutivos	53
Figura 26. Seis parpadeos consecutivos	53
Figura 27. Diagrama de bloques lineal del sistema de ubicación de los ojos	54
Figura 28. Procesos para ajustar la imagen	56
Figura 29. Recuadro del ojo en escala de grises	56
Figura 30. Imagen del ojo en blanco y negro	56
Figura 31. Imagen del ojo en blanco y negro complementada	56
Figura 32. Imagen filtrada	57
Figura 33. Imagen sin ruido	57
Figura 34. Puntos característicos del ojo en la imagen inicial	59
Figura 35. Puntos característicos del ojo	59
Figura 36. Detalles de la interfaz gráfica	61
Figura 37. Pines de alimentación y datos del puerto USB	64
Figura 38. Conector tipo B	64
Figura 39. Diagrama de conexión del hardware	65
Figura 40. Diagrama de bloques del código inicial del PIC	67
Figura 41. Software de Enumeración de Datos PIC USB	68

Figura 42. Distribución de los pines del módulo transmisor	70
Figura 43. Distribución de pines del módulo receptor	70
Figura 44. Transmisor para un enlace de RF	71
Figura 45. Receptor para un enlace de RF	71
Figura 46. Diagrama de conexión del hardware	73
Figura 47. Resultados porcentuales del algoritmo de localización de la pupila	75
Figura 48. Imagen del parpadeo	76
Figura 49. Mensaje de instalación correcta del hardware.	77
Figura 50. Presentación del hardware USB en el administrador de dispositivos	77
Figura 51. Pantalla inicial del menú	78
Figura 52. Mensaje de hardware sin conexión	79
Figura 53. Pantalla de información del proyecto	79

LISTA DE ANEXOS

	pág.
ANEXO A: Hoja de datos del PIC18F2550	86
ANEXO B: Características eléctricas del circuito HT12E	87
ANEXO C: Características eléctricas del circuito HT12D	88
ANEXO D: Imagen del circuito impreso de la interfaz USB y la etapa de transmisión	89
ANEXO E: Placa montada del USB y transmisor de RF	90
ANEXO F: Imagen del circuito impreso de la etapa de recepción y etapa de potencia	91
ANEXO G: Placa montada del receptor de RF y potencia	92
ANEXO H. Manual de usuario	93

GLOSARIO

ADC: (conversor analógico-digital) sistema que transforma una señal analógica en su equivalente digital.

ALGORITMO: conjunto ordenado y finito de operaciones que permite hallar la solución de un problema.

AXES: es un elemento de la interfaz gráfica de usuario (GUI) con el cual se puede crear áreas gráficas en alguna aplicación.

CODIFICADOR: es un circuito combinacional que cuenta con un número determinado de entradas, de las cuales solo una tiene un estado lógico. Su principal función es presentar en la salida un código binario correspondiente a la entrada activa.

CONTRACCIÓN MUSCULAR: es el proceso fisiológico en el que los músculos desarrollan tensión y se acortan o estiran.

DECODIFICADOR: es un dispositivo que acepta una entrada digital en binario y activa una salida. Su función básica es detectar la presencia de una determinada combinación de bits en sus entradas y señalar la presencia de este código a la salida.

DRIVER: es un programa que habilita aplicaciones para poderse comunicar con el dispositivo. Cada dispositivo sobre el bus debe tener un driver, algunos periféricos utilizan los drivers que trae Windows.

EAGLE: es un software de diseño de circuitos impresos de la empresa CADSOFT.

ENUMERACIÓN: proceso en que el host o computador le pregunta al dispositivo que se presente y le diga cuáles son sus parámetros, tales como el consumo de energía expresada en unidades de carga, número y tipos de puntos terminales, clase del producto, tipo de transferencia, etc.

FILTRO: dispositivo que elimina o selecciona ciertas frecuencias de un espectro eléctrico, acústico, óptico o mecánico.

GUI: (del inglés graphical user interface) es un programa informático que actúa de interfaz de usuario, utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz.

HOST: dispositivo maestro que inicia la comunicación (generalmente el computador).

HUB: dispositivo que contiene uno o más conectores o conexiones internas hacia otros dispositivos USB, el cual habilita la comunicación entre el host y con diversos dispositivos. Cada conector representa un puerto USB.

INTERFAZ: conexión física y funcional entre dos aparatos o sistemas independientes.

IRIS: disco membranoso y coloreado del ojo en cuyo centro está la pupila.

LIBRERÍA: es un conjunto de subprogramas utilizados para desarrollar software. Contienen código y datos, que proporcionan servicios a programas independientes, es decir, pasan a formar parte de éstos.

MATLAB: es un software matemático que ofrece un entorno de desarrollo integrado con un lenguaje de programación propio (lenguaje M). Está disponible para las plataformas Unix, Windows y Apple Mac OS X.

MENÚ: colección de opciones que aparece en la pantalla de un computador.

MICRÓFONO: aparato que transforma las ondas sonoras en corrientes eléctricas para su amplificación.

MIOGRAFÍA: parte de la anatomía que tiene por objeto la descripción de los músculos.

MÚSCULO: órgano compuesto principalmente de fibras contráctiles.

OJO: es un órgano que detecta la luz y es la base del sentido de la vista. Se compone de un sistema sensible a los cambios de luz, capaz de transformar éstos en impulsos eléctricos.

ORBICULAR: músculo en forma de anillo, que rodea una cavidad o abertura, como la boca o la órbita del ojo.

PID: es un número que identifica al dispositivo USB.

PUPILA: abertura circular o en forma de rendija de color negro, que el iris del ojo tiene en su parte media y que da paso a la luz.

PUSHBUTTON: es un elemento de la interfaz gráfica de usuario, con el cual se puede crear un botón para que realice algún evento en una aplicación.

RADIOFRECUENCIA: cada una de las frecuencias de las ondas electromagnéticas empleadas en la radiocomunicación.

RELÉ: aparato destinado a producir en un circuito una modificación dada, cuando se cumplen determinadas condiciones en el mismo circuito o en otro distinto.

RS-232: (Recommended Standard 232: Estándar Recomendado 232): arquitectura de comunicación de datos externa usada en PCs, notebooks, handhelds/palmtops, PLCs, y otros dispositivos que soporten este protocolo (RS-232).

SENSOR: dispositivo que detecta una determinada acción externa, temperatura, presión, etc., y la transmite adecuadamente.

STAND BY: es el consumo en espera de diferentes aparatos electrónicos, tales como televisión, reproductores de audio o vídeo, aire acondicionado, etc. En stand by, el aparato se encuentra conectado a la espera de recibir órdenes, por lo que consume energía eléctrica.

STATIC TEXT: es un elemento de la interfaz gráfica con el cual se puede crear una etiqueta o un letrero.

USB: es una especificación que describe un canal serie que soporta una gran variedad de periféricos de media y baja velocidad, con soporte integral para transferencias en tiempo real como voz, audio y vídeo comprimido.

VID: es un número de 16 bits asignado por el USB que identifica al fabricante.

RESUMEN

Gracias a que las señales miográficas pueden ser generadas de manera voluntaria por las personas y al requerimiento mínimo de sensores y de canales para adquirirlas, representan una excelente opción para ser utilizadas en el control de interfaces entre el hombre y el computador.

En general, el sistema que pretende servir de medio de comunicación, principalmente en personas con alguna discapacidad motriz consta de los siguientes procesos fundamentales:

Presentación de un menú gráfico, donde se muestran algunos objetos o imágenes relacionados con algún aparato electrónico o sistema externo que el usuario desea activar o desactivar.

Ubicación de pupila, mediante la implementación de un software que permite conocer la selección del objeto en el menú usando una cámara web.

Adquisición de señales musculares cercanas al ojo, que determina el parpadeo del usuario y al mismo tiempo la activación/desactivación de los dispositivos externos.

Envío de datos al puerto USB, y a un medio de radiofrecuencia para interactuar con los dispositivos externos.

ABSTRACT

Considering that the myographic signals can be generated voluntarily by the people and the minimum requirement of sensor and channels to acquire, it represents an excellent choice for use in the control of man-computer interface.

In general, the system that aims serve like means of communication, mainly for people with motor disability, it has the following fundamental processes:

Presentation of a graphical menu, which displays some objects or pictures related to any electronic device or external system that the user wants to enable or disable.

Pupil location by means of implementing software that allow knowing the object selection on the menu using a webcam.

Acquisitions of muscle signals near the eye, which determine the flicker of the user, while it is enable/disable some external devices.

Sending of data to the USB port and RF (Radiofrequency) hardware to interact with external devices.

INTRODUCCIÓN

Una interfaz entre el hombre y el computador es un canal de control y comunicación electrónico que no utiliza los nervios periféricos y los músculos, vías de salida normales del cerebro, sino que transforma directamente procesos corporales en acciones. Dichas interfaces analizan la actividad del cuerpo y detectan estados voluntarios del usuario en tiempo real. Luego los transforman en órdenes operativas, ambas sin que la persona ejecute ningún movimiento. Basta con que visualice imágenes que representen acciones, para que se ejecuten sucesos externos al cuerpo.

Las señales del cuerpo son utilizadas principalmente en diagnóstico médico para detectar patologías en los órganos que las producen, pero también pueden ser utilizadas, particularmente aquellas que son generadas de manera voluntaria, para controlar interfaces hombre computador. Las señales miográficas son señales que se producen en los músculos cuando estos se contraen o distensionan. Estas señales, pueden medirse con un equipo adecuado, y esta información puede ser empleada para el control de interfaces hombre – máquina, donde las señales medidas en músculos activados voluntariamente por un usuario, se procesan y se utilizan como comandos para controlar dispositivos electrónicos o interfaces de tipo software.

Teniendo en cuenta lo anterior, este proyecto se enfoca en el análisis y la implementación de un sistema entre el hombre y el computador, con el fin de mejorar la comunicación de personas con capacidades limitadas, activando mecanismos electrónicos como el calefactor o el teléfono sin necesidad de mover sus articulaciones u órganos inmóviles.

1. TEMA

1.1 TÍTULO

Diseño e implementación de un sistema que activa dispositivos electrónicos mediante señales oculares para mejorar la comunicación a personas con discapacidad motora.

1.2 MODALIDAD

El trabajo de grado se desarrolla bajo la modalidad de Investigación, puesto que la información necesaria para desarrollar este proyecto abarca temas que aún no han sido analizados y aplicados, principalmente en nuestra región.

1.3 LÍNEA DE INVESTIGACIÓN

El proyecto se orienta al diseño y construcción de un sistema electrónico que permite la comunicación de personas con carencias motoras mediante el uso de señales provenientes del cuerpo y un computador, por lo tanto abarca temas como: adquisición de señales corporales, tratamiento y análisis de datos, y desarrollos de algoritmos de computador, por lo que el desarrollo del proyecto se incluye dentro de ELECTROMEDICINA.

2. DESCRIPCIÓN DEL PROBLEMA

2.1 PLANTEAMIENTO

Actualmente se observa que muchas personas sufren de ciertas enfermedades que les impide moverse o realizar actividades cotidianas. Con dichas limitaciones es necesario ayudarse con elementos complementarios como bastones o sillas de ruedas, y en el caso de que se tenga restricciones para el desarrollo del lenguaje o manipulación de objetos, perderá la habilidad para comunicarse con su entorno.

La dificultad para interactuar con otros individuos se presenta por la ausencia de movimiento o estimulación de órganos que intervienen en la normal comunicación debido a alguna enfermedad, como la Esclerosis lateral amiotrófica (ELA), caracterizada por la degeneración de las neuronas motoras que controlan los músculos esqueléticos lo que produce atrofia de los músculos que han perdido su inervación, originando debilidad muscular y parálisis en los individuos.

Al no existir un dispositivo que permita el desarrollo de personas con dificultad para comunicarse, se obliga a la persona que padece la enfermedad o alguna incapacidad a estar limitada en cuanto a la manifestación de sus necesidades y la relación con el medio, y a perder por completo el movimiento de articulaciones afectadas si no recupera mediante algún dispositivo que lo estimule.

2.2 FORMULACIÓN

¿Cómo ayudar a personas con discapacidades motoras, a comunicarse con su entorno, mediante la utilización de sistemas electrónicos y las señales provenientes del ojo humano?

2.3 SISTEMATIZACIÓN

- ¿Cómo estimular señales corporales de un paciente con carencia de movimiento o habla para permitirle su comunicación con el medio?
- ¿Cómo determinar la opción de un menú gráfico que el usuario ha escogido y desea activar?
- ¿Cómo relacionar objetos de un menú de computador con el mundo externo para la comunicación de un individuo con su medio?

3. OBJETIVOS

3.1 OBJETIVO GENERAL

Diseñar y construir un sistema electrónico para facilitar la comunicación de personas con discapacidad motora, usando un menú gráfico y señales miográficas, para la activación de sistemas electrónicos tales como un televisor, un radio y una lámpara.

3.2 OBJETIVOS ESPECÍFICOS

- Diseñar una interfaz o menú gráfico que represente diferentes acciones cotidianas para que el paciente las pueda observar en un computador y con ello captar y acondicionar las señales voluntarias producidas cuando desee seleccionar un objeto o acción representada en dicho menú.
- Generar un programa especializado para la detección de la posición de los ojos del usuario que permita conocer la opción elegida y la selección mediante un parpadeo, esto mediante un software, el cual se seleccionará dependiendo de las condiciones que se ajusten mejor a los resultados esperados.
- Elaborar una interfaz de salida de datos del computador para controlar dispositivos electrónicos, entre los cuales están un radio, un televisor y una lámpara, acordes a la opción elegida por el usuario.

4. JUSTIFICACIÓN

Los discapacitados a veces tienen dificultad para ciertas actividades consideradas por otras personas como totalmente normales, como viajar en transporte público, subir escaleras o incluso utilizar ciertos electrodomésticos. Por lo tanto es muy importante hacer que estas personas tengan una vida normal recurriendo a las ventajas que ofrece los elementos electrónicos.

El proyecto tiene en cuenta conocimientos tales como la anatomía de los órganos oculares y el análisis de señales generadas en los músculos del mismo. Igualmente se hace uso de algoritmos computacionales para la clasificación de los tipos de señales producidas por una estimulación visual.

Este trabajo de investigación puede servir de base para otros estudios futuros, de modo que se puedan hacer correcciones, mejoras o llevarlo a otro tipo de aplicaciones.

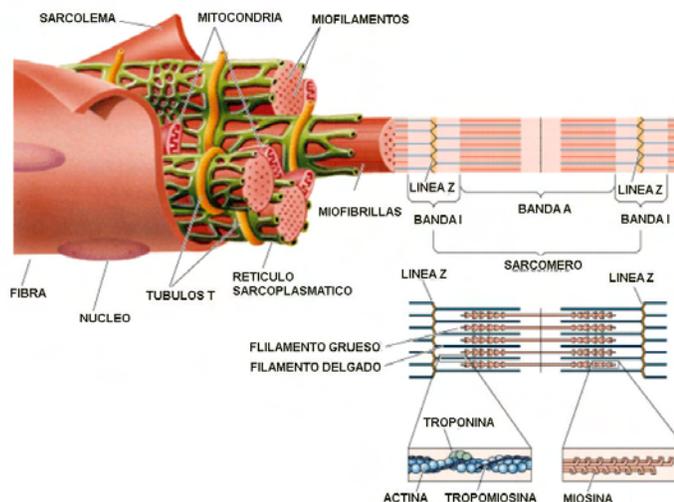
La investigación está centrada en mejorar la comunicación a personas que sufran de alguna discapacidad motriz y con esto mejorar la calidad de vida a un grupo de personas, especialmente a centros de rehabilitación, hospitales y centros de salud.

5. MARCO TEÓRICO

5.1 LOS MÚSCULOS¹

Los músculos son tejidos u órganos del cuerpo caracterizado por su capacidad para contraerse, por lo general en respuesta a un estímulo nervioso. La unidad básica de todo músculo es la miofibrilla, estructura filiforme muy pequeña formada por proteínas complejas (figura 1). Cada célula muscular o fibra contiene varias miofibrillas, compuestas de miofilamentos de dos tipos, gruesos y delgados, que adoptan una disposición regular. Cada miofilamento grueso contiene varios cientos de moléculas de la proteína miosina. Los filamentos delgados contienen dos cadenas de la proteína actina. Las miofibrillas están formadas de hileras que alternan miofilamentos gruesos y delgados con sus extremos traslapados. Durante las contracciones musculares, estas hileras de filamentos se deslizan una sobre otra por medio de puentes cruzados que actúan como ruedas. La energía que requiere este movimiento procede de mitocondrias densas que rodean las miofibrillas.

Figura 1. Diagrama de una fibra muscular



Fibra muscular. Disponible en Internet. <http://www.monografias.com/trabajos57/sistema-muscular/sistema-muscular.shtml>

¹ VALERIUS, Klaus. El libro de los músculos. ARS Médica. 2008. 419 págs.

5.1.1 Contracción muscular

Los músculos esqueléticos realizan dos acciones: contracción y relajación. Al ser estimulado el músculo por un impulso motor, éste se contrae; cuando el impulso se discontinúa, el músculo se relaja.

5.1.2 Músculos de la cara²

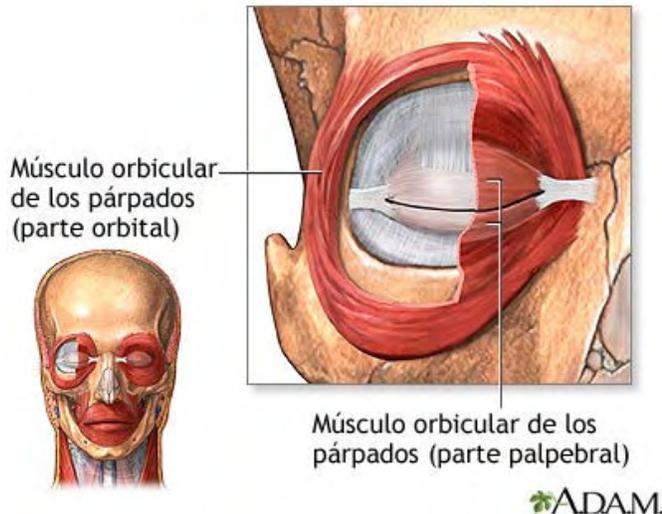
Los músculos de la cara contribuyen a la apertura y la oclusión de los orificios faciales, a la masticación y a la expresión mímica.

En la frente se encuentra el músculo frontal, es un músculo cutáneo del cráneo. Algunos autores lo consideran la porción muscular anterior del músculo occipitofrontal. Se halla inervado por los filetes frontales de la rama de bifurcación superior del nervio facial.

5.1.2.1 Músculo Orbicular de los párpados

El músculo orbicular de los párpados (figura 2) se encuentra delante de la órbita ocular; en forma de anillo, ancho, aplanado y delgado, constituido por dos porciones: una orbitaria y otra palpebral.

Figura 2. Ubicación del músculo orbicular del párpado



² Músculos de la cara. Documentación HTML. Disponible en Internet.
http://es.wikipedia.org/wiki/M%C3%BAsculos_de_la_cara#M.C3.BAsculo_Orbicular_de_los_p.C3.A1rpados

Se inserta, por dentro en el tendón orbicular, que a su vez se inserta en los labios anterior y posterior del canal lagrimal, en el proceso ascendente del maxilar superior y el proceso orbitario interno del hueso frontal y, por fuera, en la cara profunda de la piel.

Su función es ocluir el ojo descendiendo el párpado superior.

5.2 Micrófono³

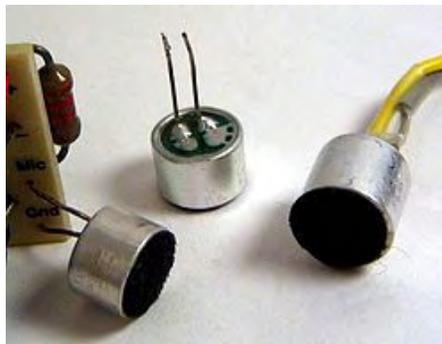
Los micrófonos son transductores electroacústico que se ocupan de transformar la presión sonora ejercida sobre su cápsula en energía eléctrica. La membrana o diafragma es un elemento fundamental que está presente en cada uno de ellos. Las diferencias que estriban entre los diferentes tipos de micrófonos se basan principalmente en la sensibilidad que son capaces de proporcionar, que están directamente ligadas a la capacidad del micrófono de capturar las oscilaciones mecánicas que provienen de la membrana, y transformar proporcionalmente con éxito dichas oscilaciones en energía eléctrica.

Existen algunos tipos de micrófonos entre los cuales se tiene:

Micrófono dinámico o de bobina móvil, poco sensible a las ondas acústicas y muy robusto físicamente. El rango de frecuencias esta entre 40-16.000 hertzios.

Micrófono de condensador o electrostático (figura 3), su rango de frecuencias esta entre 20-18.000Hz. Tienen una captación muy sensible y una respuesta en frecuencia amplia y fiable.

Figura 3. Micrófono de condensador electret



³ Borwick, John, Micrófonos, Tecnologías y Aplicaciones. Escuela de Cine y Video, 1990.

Micrófono de cinta o gradiente de presión, su rango de frecuencias es de 30-18.000Hz. Son micrófonos muy sensibles, con una respuesta en frecuencia muy uniforme, con muy buena calidad sonora, tienen el inconveniente de ser poco robustos y poco compactos, sensibles al viento y fácilmente saturables en distancias cercanas o para fuentes sonoras fuertes.

5.3 Tarjeta de sonido⁴

Es una tarjeta de expansión para computadores que permite la entrada y salida de audio bajo el control de un programa informático llamado controlador.

El típico uso de las tarjetas de sonido consiste en proveer mediante un programa que actúa de mezclador, que las aplicaciones multimedia del componente de audio suenen y puedan ser gestionadas. Estas aplicaciones multimedia engloban composición y edición de video o audio, presentaciones multimedia.

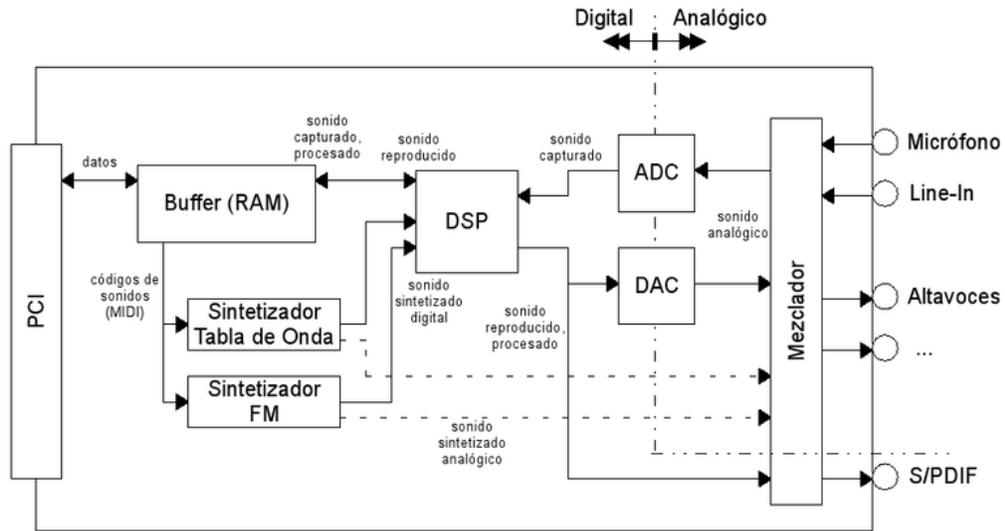
Una tarjeta de sonido típica, incorpora un chip de sonido que por lo general contiene el convertidor digital-analógico, el cual cumple con la importante función de "traducir" formas de ondas grabadas o generadas digitalmente en una señal analógica y viceversa. Esta señal es enviada a un conector (para auriculares) en donde se puede conectar cualquier otro dispositivo como un amplificador, un altavoz, etc. Para poder grabar y reproducir audio al mismo tiempo con la tarjeta de sonido debe poseer la característica "full-dúplex" para que los dos conversores trabajen de forma independiente.

La mayoría de las tarjetas de sonido también tienen un conector de entrada o "Line In" por el cual puede entrar cualquier tipo de señal de audio proveniente de otro dispositivo como micrófonos, reproductores de casetes entre otros y luego así la tarjeta de sonido puede digitalizar estas ondas y guardarlas en el disco duro del computador.

La figura 4 muestra un diagrama simplificado de los componentes típicos de una tarjeta de sonido.

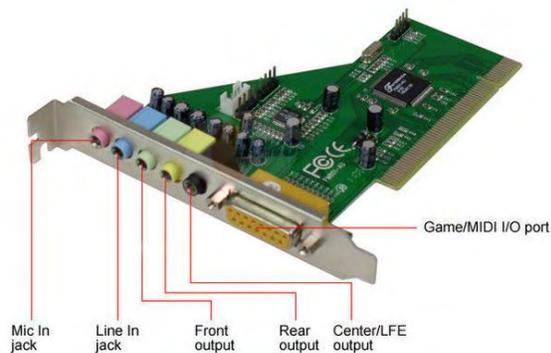
⁴ Tarjeta de sonido, Documentación HTML. Disponible en Internet http://es.wikipedia.org/wiki/tarjeta_de_sonido.htm

Figura 4. Componentes de una tarjeta de sonido



Los conectores más utilizados para las tarjetas de sonido a nivel de usuario son los minijack al ser los más económicos. Con los conectores RCA se consigue mayor calidad ya que utilizan dos canales independientes, el rojo y el blanco, uno para el canal derecho y otro para el izquierdo.

Figura 5. Puertos de una tarjeta de sonido estándar



Otra de las funciones básicas de una tarjeta de sonido es la digitalización; para que el ordenador pueda tratar el sonido, debe convertirlo de su estado original (analógico) al formato que él entiende, binario (digital). En este proceso se realiza lo que se denomina muestreo, que es recoger la información y cuantificarla, es decir, medir la altura o amplitud de la onda. El proceso se realiza a una velocidad fija, llamada frecuencia de muestreo; cuanto mayor sea esta, más calidad tendrá el sonido, porque más continua será la adquisición del mismo.

La frecuencia mínima recomendable es de 44.1 KHz, con la que se puede obtener una calidad comparable a la de un disco compacto.

La audición humana está limitada a los 16 ó 17 KHz, pero si los equipos se extienden más allá de este límite se tiene una mejor calidad, también que la frecuencia de muestreo (del convertidor) debe ser de más del doble que la banda que se pretende utilizar, esta afirmación es conocida como el teorema de Nyquist. Finalmente los nuevos formatos de alta definición usan frecuencias de muestreo de 96 KHz (para tener una banda de 40 KHz) y hasta 192 KHz, no porque estas frecuencias se puedan oír, sino porque así es más fácil reproducir las que si se oyen.

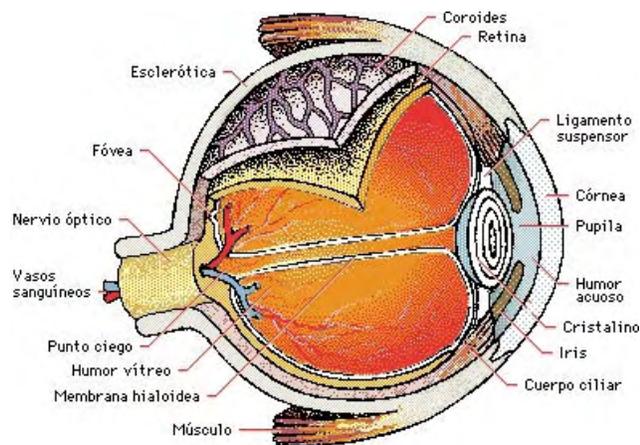
5.4 EL OJO ^{5, 6}

Es un órgano casi esférico de 2,5 cm de diámetro que detecta la luz y es la base del sentido de la vista. Se compone de un sistema sensible a los cambios de luz, capaz de transformar éstos en impulsos eléctricos.

El ojo humano (figura 6) posee un lente llamado cristalino que es ajustable según la distancia, un diafragma que se llama pupila cuyo diámetro está regulado por el iris y un tejido sensible a la luz que es la retina.

El iris es una estructura pigmentada suspendida entre la córnea y el cristalino y tiene una abertura circular en el centro, la pupila. El tamaño de la pupila depende de un músculo que rodea sus bordes, aumentando o disminuyendo cuando se contrae o se relaja, controlando la cantidad de luz que entra en el ojo.

Figura 6. Partes del ojo



El ojo. Disponible en internet. <http://www.profesorenlinea.cl/Ciencias/OjoVision>

⁵ Tortora, Derrickson. Principios de Anatomía y Fisiología, 11ª edición. 2006.

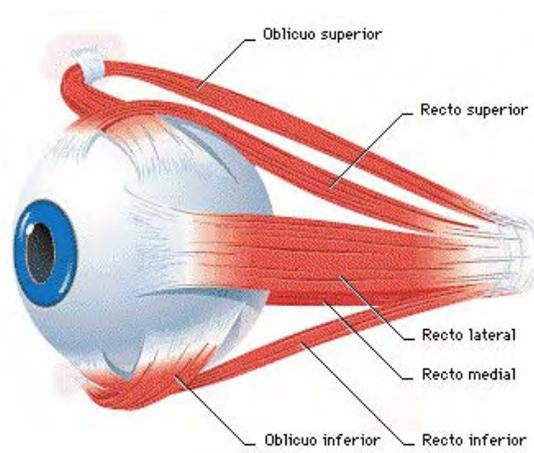
⁶ Ross, L.H. Histology: a text and atlas. 5th edición. Baltimore: Williams & Wilkins. 2006.

La luz penetra a través de la pupila, atraviesa el cristalino y se proyecta sobre la retina, donde se transforma gracias a unas células llamadas fotorreceptoras en impulsos nerviosos que son trasladados a través del nervio óptico al cerebro.

5.4.1 Músculos propios del ojo

La figura 7 muestra la parte lateral del ojo, donde se puede observar los músculos extrínsecos unidos directamente al globo ocular que permiten el movimiento del ojo. Los cuatro rectos están alineados con sus puntos de origen, mientras que los dos oblicuos se insertan en la superficie ocular formando un ángulo.

Figura 7. Músculos del ojo



5.4.2 Estructuras protectoras

Diversas estructuras, que no forman parte del globo ocular, contribuyen en su protección. Las más importantes son los párpados superior e inferior. Estos son pliegues de piel y tejido glandular que pueden cerrarse gracias a unos músculos y forman sobre el ojo una cubierta protectora contra un exceso de luz o una lesión mecánica.

Un parpadeo en el ojo humano es un acto reflejo que se produce más o menos cada seis segundos; pero si el polvo alcanza su superficie y no se elimina por lavado, los párpados se cierran con más frecuencia y se produce mayor cantidad de lágrimas.

5.5 PROCESAMIENTO DE DATOS

5.5.1 Filtros digitales^{7, 8}

Un filtro digital es un algoritmo implementado en hardware y/o software que opera sobre una señal de entrada digital (discreta en tiempo y cuantizada en amplitud) y genera una señal digital de salida, con la finalidad de efectuar un proceso de filtrado. El término “filtro digital” se refiere al hardware o software específico que ejecuta el algoritmo. Los filtros digitales trabajan sobre valores numéricos asociados a muestras de esas señales analógicas previamente digitalizadas por conversores A/D o simplemente sobre un conjunto de números almacenados en la memoria de un computador o microprocesador.

La señal de entrada, limitada en banda, se muestrea periódicamente y se convierte en una serie de muestras $x[n]$, $n = 0, 1, \dots$. El procesador digital convierte la sucesión de entrada $x[n]$ en una sucesión de salida $y[n]$ de acuerdo al algoritmo de cómputo. El conversor digital-analógico convierte la salida digital $y[n]$ a valores continuos en tiempo que son procesados por un filtro analógico para suavizar la forma de onda y remover componentes no deseadas de alta frecuencia.

Los filtros digitales juegan un papel muy importante en el procesamiento digital de señales. En gran número de aplicaciones, como compresión de datos, procesamiento de señales biomédicas, procesamiento de señales de voz, procesamiento de imágenes, etc.

Un tema muy importante es considerar las limitaciones del filtro de entrada debido al Teorema de muestreo de Nyquist-Shannon que en pocas palabras; si se quiere procesar hasta una frecuencia de 10KHz, se debe muestrear a por lo menos 20 KHz.

Uno de los filtros electrónicos más básicos es el filtro de Butterworth, diseñado para producir la respuesta más plana que sea posible hasta la frecuencia de corte.

El filtro de Butterworth más básico es el típico filtro pasa bajo de primer orden, el cual puede ser modificado a un filtro pasa alto o añadir en serie otros formando un filtro pasa banda o elimina banda y filtros de mayores órdenes.

⁷ PROAKIS, John G y MANOLOKIS, Dimitris G., Tratamiento digital de señales: Principios, algoritmos y aplicaciones, 3ra edición, Pearson - Prentice Hall, 2005.

⁸ Ifeachor, E., Jervis, B. Digital Signal Processing: a practical approach, Addison-Wesley Longman Ltd, Edingburg Gate, Harlow, Inglaterra. 1993.

5.5.2 Transformadas

Una transformada es la operación matemática que convierte una función de un dominio al otro, sin pérdida de información.

La transformada de Fourier se utiliza para pasar al dominio frecuencial una señal para así obtener información que no es evidente en el dominio temporal. Se demuestra matemáticamente que una señal periódica se puede descomponer en una suma de senos y cosenos formando una base ortogonal, de esta forma, señales como la voz o las ondas se pueden descomponer en un sumatorio de señales trigonométricas. El conjunto de constantes que multiplican a cada frecuencia forman el espectro de frecuencias.

La transformada de Fourier también es utilizada en el ámbito del tratamiento digital de imágenes, como por ejemplo para mejorar o definir más ciertas zonas de una imagen fotográfica o tomada con un computador.

5.5.3 Procesamiento de imágenes^{9, 10}

El procesamiento digital de imágenes es el conjunto de técnicas que se aplican a las imágenes digitales con el objetivo de mejorar la calidad o facilitar la búsqueda de información.

5.5.3.1 Proceso de filtrado

Es el conjunto de técnicas englobadas dentro del pre procesamiento de imágenes cuyo objetivo fundamental es obtener, a partir de una imagen origen, otra final cuyo resultado sea más adecuado para una aplicación específica mejorando ciertas características de la misma que posibilite efectuar operaciones del procesado sobre ella.

Los principales objetivos que se persiguen con la aplicación de filtros son:

- *Suavizar la imagen*: reducir la cantidad de variaciones de intensidad entre píxeles vecinos.
- *Eliminar ruido*: eliminar aquellos píxeles cuyo nivel de intensidad es muy diferente al de sus vecinos y cuyo origen puede estar tanto en el proceso de adquisición de la imagen como en el de transmisión.
- *Realzar bordes*: destacar los bordes que se localizan en una imagen.

⁹ F Acharya, T., Ray, A. K. (2005). Image processing: principles and applications. John Wiley & Sons.

¹⁰ González, R.C., Wintz, P. (1996). Procesamiento digital de imágenes. Addison-Wesley.

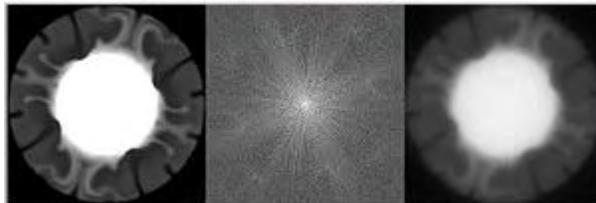
- *Detectar bordes*: detectar los píxeles donde se produce un cambio brusco en la función intensidad.

Por tanto, se consideran los filtros como operaciones que se aplican a los píxeles de una imagen digital para optimizarla, enfatizar cierta información o conseguir un efecto especial en ella. El proceso de filtrado puede llevarse a cabo sobre los dominios de frecuencia y/o espacio.

- **Filtrado en el dominio de la frecuencia**

Los filtros de frecuencia procesan una imagen trabajando sobre el dominio de la frecuencia en la Transformada de Fourier de la imagen.

Figura 8. Ejemplo de filtrado en el dominio de la frecuencia



Existen básicamente tres tipos distintos de filtros que pueden aplicarse:

- **Filtro pasa bajo**: atenúa las frecuencias altas y mantiene sin variaciones las bajas. El resultado en el dominio espacial es equivalente al de un filtro de suavizado, donde las altas frecuencias que son filtradas se corresponden con los cambios fuertes de intensidad. Consigue reducir el ruido suavizando las transiciones existentes.
- **Filtro pasa alto**: atenúa las frecuencias bajas manteniendo invariables las frecuencias altas. Puesto que las altas frecuencias corresponden en las imágenes a cambios bruscos de densidad, este tipo de filtros es usado, porque entre otras ventajas, ofrece mejoras en la detección de bordes en el dominio espacial, ya que estos contienen gran cantidad de dichas frecuencias. Refuerza los contrastes que se encuentran en la imagen.
- **Filtro pasa banda**: atenúa frecuencias muy altas o muy bajas manteniendo una banda de rango medio.

- **Filtrado en el dominio del espacio**

Las operaciones de filtrado se llevan a cabo directamente sobre los píxeles de la imagen. En este proceso se relaciona, para todos y cada uno de los puntos de la imagen, un conjunto de píxeles próximos al píxel objetivo con la finalidad de

obtener una información útil, dependiente del tipo de filtro aplicado, que permita actuar sobre el píxel concreto en que se está llevando a cabo el proceso de filtrado para, de este modo, obtener mejoras sobre la imagen y/o datos que podrían ser utilizados en futuras acciones o procesos de trabajo sobre ella.

Este filtro puede ser de varios tipos:

- **Filtro pasa bajo** (suavizamiento): utilizados para eliminar ruido o detalles pequeños de poco interés puesto que sólo afecta a zonas con muchos cambios y pueden ser:
 - ✓ *Promedio*: promedio de píxeles vecinos.
 - ✓ *Paso bajo en frecuencia*.
 - ✓ *Media*: reemplaza cada píxel por el valor medio de sus contiguos.
 - ✓ *Mediana*: sustituye por el valor de la mediana de los píxeles vecinos (normalmente se comporta mejor que el de promedio).
 - ✓ *Gaussiano*: aproximación a la distribución gaussiana.

Figura 9. Ejemplo de filtrado promedio con filtro 5x5



- **Filtro paso alto** (atenuamiento): intensifica los detalles, bordes y cambios de alta frecuencia, mientras que atenúa las zonas de tonalidad uniforme. Esto permite una mejor identificación posterior de los objetos que se encuentren en la imagen, puesto que el brillo se hace mayor en las zonas con frecuencias más altas, al mismo tiempo que se oscurecen las zonas de frecuencias bajas. Es común la aparición de ruido tras el proceso.
- **Realce de bordes por desplazamiento y diferencia**: sustrae de la imagen original una copia desplazada de la misma. Así, es posible localizar y hacer resaltar los bordes existentes.
- **Realce de bordes mediante Laplace**: este tipo de filtros realza los bordes en todas direcciones (los resultados que se obtienen pueden considerarse como una “suma” de los obtenidos tras aplicar todos los modelos del tipo anterior).

- **Resalte de bordes con gradiente direccional:** empleado para destacar y resaltar con mayor precisión los bordes que se localizan en una dirección determinada. Trabaja con los cambios de intensidad existentes entre píxeles contiguos.
- **Detección de bordes y filtros de contorno (Prewitt y Sobel):** al igual que los anteriores, se centra en las diferencias de intensidad que se dan pixel a pixel. Son utilizados para obtener los contornos de objetos y de este modo clasificar las formas existentes dentro de una imagen. Este tipo de filtros requieren un menor coste computacional.

Figura 10. Imagen original y resulta tras filtros Laplaciano y Sobel



5.5.3.2 Posición de la pupila del ojo^{11, 12}

a) Detección de iris

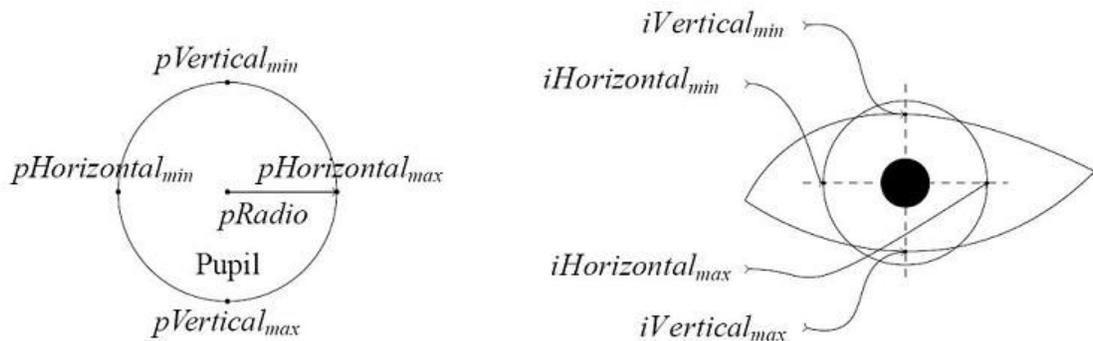
El primer paso en el procedimiento de reconocimiento de Iris precisa de una correcta detección de la Pupila. Considerando las características geométricas de la Pupila, (casi circular), es obviamente innecesario utilizar todo el contorno como resultado de la segmentación. Por el contrario, se propone el uso de únicamente cinco puntos: pCenter, pVerticalmin, pVerticalmax, pHorizontalmin y pHorizontalmax.

Estos puntos (figura 11) engloban toda la información necesaria para afirmar no solo donde se encuentra la pupila sino también hasta qué punto se encuentra dilatada.

¹¹ Daugman, 'How Iris Recognition Works', IEEE Transactions on Circuits and Systems For Video Technology, Vol. 14, n 1, January 2004.

¹² Santos Sierra, C. Sánchez Ávila, E. Marchiori, Iris Recognition: Segmentation enhancement by using Morphological Operators, Master Final Thesis, June, 2007. Amsterdam.

Figura 11. Puntos más importantes del Ojo: Puntos de Pupila (Izquierda), Puntos de Iris (Derecha)



Con $pCenter$ y $pRadio$, se pueden describir una circunferencia de forma precisa. Sin embargo, existen situaciones donde la Pupila puede estar desplazada en la imagen (debido a la concavidad ocular), perdiendo parte de sus propiedades circulares. Esto no quiere decir que la pupila deje de ser circular, sino que debido a la captura de la imagen, la perspectiva puede ofrecer una pupila con cierta excentricidad, más similar a una elipse que a una circunferencia. En estas situaciones se requerirán más puntos para extraer toda la información de la pupila,

5.6 INTERFAZ GRÁFICA¹³

La interfaz gráfica de usuario, conocida también como GUI (del inglés graphical user interface) es un programa informático que actúa de interfaz de usuario, utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso, consiste en proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una máquina o computador.

Habitualmente las acciones se realizan mediante manipulación directa, para facilitar la interacción del usuario con el computador. Surge como evolución de los intérpretes de comandos que se usaban para operar los primeros sistemas operativos y es pieza fundamental en un entorno gráfico. Como ejemplos de interfaz gráfica de usuario, cabe citar los entornos de escritorio Windows, el X-Window de GNU/Linux o el de Mac OS X, Aqua.

En el contexto del proceso de interacción persona-ordenador, la interfaz gráfica de usuario es el artefacto tecnológico de un sistema interactivo que posibilita, a través del uso y la representación del lenguaje visual, una interacción amigable con un sistema informático.

¹³ Shneiderman, Ben. Designing The user interface, Strategies for effective Human-computer interaction. Addison-wesley. 1998.

5.6.1 Tipos de interfaces gráficas de usuario

5.6.1.1 Gui's y zooming user interfaz

Los tipos de GUIs que se encuentran en juegos de computador, y los GUIs avanzados basados en realidad virtual, se usan con frecuencia en tareas de investigación.

5.6.1.2 Touchscreen user interface

Algunos GUIs son diseñados para cumplir con los rigurosos requisitos de los mercados verticales. Éstos se conocen como "GUIs de uso específico." Un ejemplo de un GUI de uso específico es el ahora familiar Touchscreen o Pantalla Táctil (pantalla que al ser tocada efectúa los comandos del ratón en el software).

Figura 12. Interfaz touchscreen



5.6.1.3 Interfaz Natural de Usuario

Las NUI naturales son aquellas en las que se interactúa con un sistema, aplicación, etc., sin utilizar dispositivos de entrada como ratón, teclado, lápiz óptico, etc. En lugar de éstos se utilizan las manos o las yemas de los dedos.

Figura 13. Interfaz natural



5.6.2 Guide de MATLAB

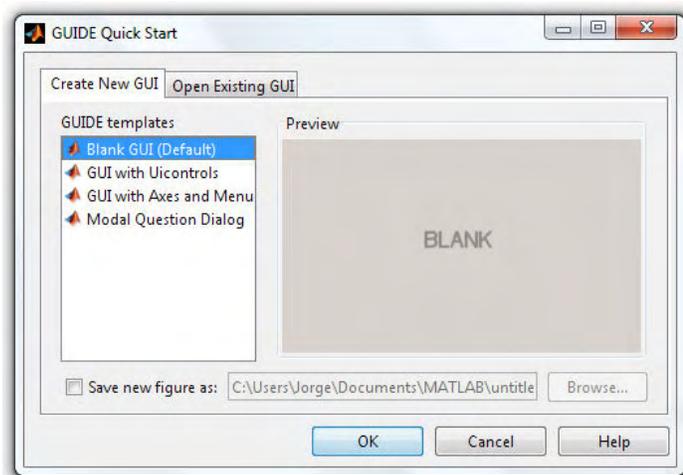
GUIDE es un entorno de programación visual disponible en el software MATLAB para realizar y ejecutar programas que necesiten ingreso continuo de datos. Tiene las características básicas de todos los programas visuales como Visual Basic o Visual C++.

Al iniciar una función GUIDE se presenta el cuadro de dialogo mostrado en la figura 14, el cual presenta las siguientes opciones:

- **Blank GUI (Default)**

La opción de interfaz gráfica de usuario en blanco (viene predeterminada), presenta un formulario nuevo, en el cual se puede diseñar el programa.

Figura 14. Cuadro de dialogo de la interfaz grafica



- **GUI with Uicontrols**

Esta opción presenta un ejemplo en el cual se calcula la masa, dada la densidad y el volumen, en alguno de los dos sistemas de unidades. Se puede ejecutar este ejemplo y obtener resultados.

- **GUI with Axes and Menu**

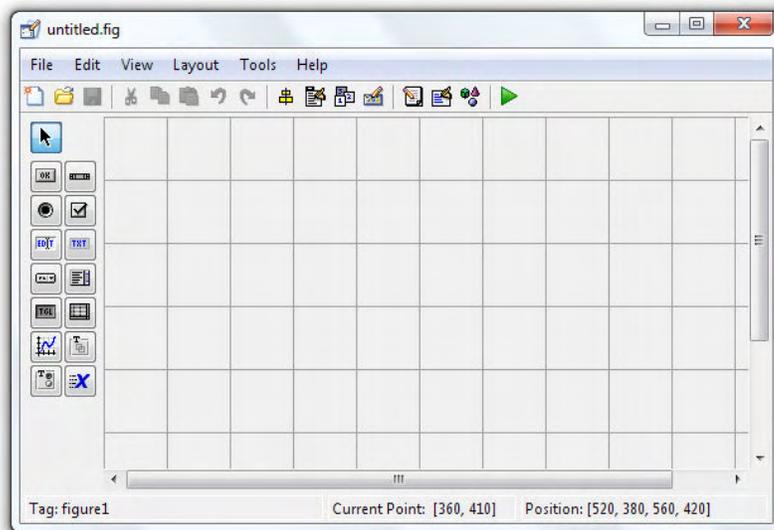
Esta opción es otro ejemplo el cual contiene el menú File con las opciones Open, Print y Close. En el formulario tiene un Popup menu, un push button y un objeto Axes, se puede ejecutar el programa eligiendo alguna de las seis opciones que se encuentran en el menú despegable y haciendo click en el botón de comando.

- **Modal Question Dialog**

Con esta opción se muestra en la pantalla un cuadro de diálogo común, el cual consta de una pequeña imagen, una etiqueta y dos botones Yes y No, dependiendo del botón que se presione, el GUI retorna el texto seleccionado (la cadena de caracteres 'Yes' o 'No').

La pantalla o interfaz en blanco se muestra en la figura 15. En esta se muestra en la parte izquierda una barra de herramientas de todos los componentes que se puede utilizar, y en la derecha se presenta el área de diseño, la cual se la puede ampliar o reducir según las necesidades.

Figura 15. Interfaz gráfica en blanco



Una aplicación GUIDE consta de dos archivos: .m y .fig. El archivo .m es el que contiene el código con las correspondencias de los botones de control de la

interfaz y el archivo .fig contiene los elementos gráficos. Cada vez que se adicione un nuevo elemento en la interfaz gráfica, se genera automáticamente código en el archivo .m.

Todos los valores de las propiedades de los elementos (color, valor, posición, string, entre otros) y los valores de las variables transitorias del programa se almacenan en una estructura, los cuales son accedidos mediante un único y mismo identificador para todos éstos. El identificador se asigna en:

```
handles.output = hObject;
```

handles, es el identificador a los datos de la aplicación. Esta definición de identificador es salvada con la instrucción *guidata(hObject, handles)*.

guidata, es la sentencia para guardar los datos de la aplicación.

Teniendo en cuenta que guidata es la función que guarda las variables y propiedades de los elementos en la estructura de datos de la aplicación, como regla general, en cada subrutina se debe escribir en la última línea lo siguiente:

```
guidata(hObject,handles)
```

Esta sentencia garantiza que cualquier cambio o asignación de propiedades o variables quede almacenado.

5.7 USB^{14, 15}

(Universal Serial Bus) es una especificación que describe un canal serie que soporta una gran variedad de periféricos de media y baja velocidad, con soporte integral para transferencias en tiempo real como voz, audio y vídeo comprimido.

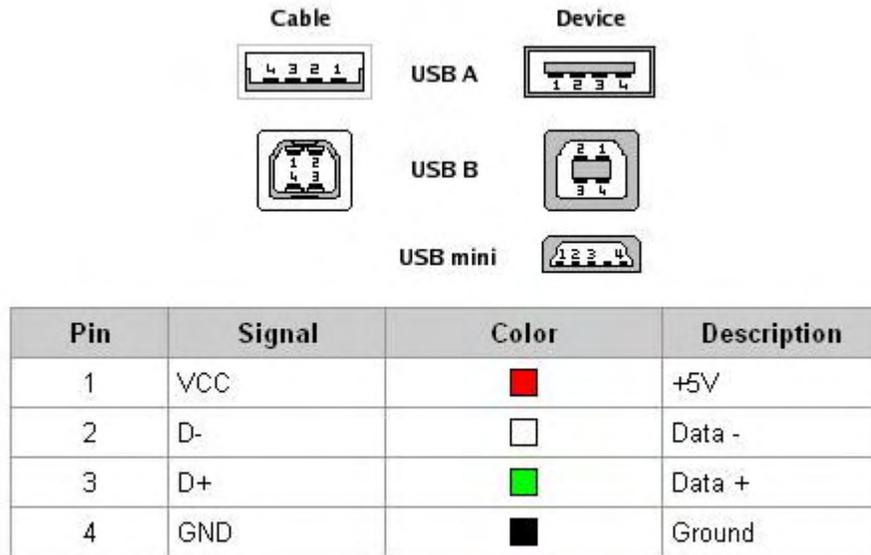
El USB es un bus punto a punto, dado que el lugar de partida es el host (PC o hub), el destino es un periférico u otro hub. No hay más que un único host (PC) en una arquitectura USB.

A nivel eléctrico, el cable USB transfiere la señal y la alimentación sobre 4 hilos (figura 16). A nivel de alimentación, el cable proporciona la tensión nominal de 5 V. Es necesario definir correctamente el diámetro del hilo con el fin de que no se produzca una caída de tensión demasiado importante en el cable.

¹⁴ Hyde, John, USB Design by Example, A practical guide to building I/O devices. Intel University Press, p.28-35

¹⁵ Axelson, Jan, USB Complete, Third Edition, Lakeview Research. Madison.

Figura 16. Características de los pines y conectores de USB



Cuando se conecta un dispositivo USB al PC se produce el Proceso de Enumeración, el cual es inicializado por el host cuando detecta que un nuevo dispositivo ha sido adjuntado al Bus. El host le asigna una dirección al dispositivo adjuntado al bus y habilita su configuración permitiendo la transferencia de datos sobre el bus.

En la serie 18F de los microcontroladores PIC de Microchip, hay unos micros que poseen módulos de USB. Estos módulos son de gran importancia en el acontecer actual porque el USB se encuentra en la mayoría de los dispositivos electrónicos que requieren comunicarse externamente.

Por otro lado y dependiendo de las características de la transmisión, es necesario configurar un driver del lado del ordenador, a fin de que el sistema operativo reconozca al PIC como un dispositivo de transacción de datos, para que el driver sea configurado correctamente, el ordenador requiere de una secuencia de datos con información relativa al dispositivo, esta secuencia de datos se encuentran organizadas en el archivo instalador .inf.

A nivel de usuario, se resume a 5 los datos para personalizar los archivos instaladores .inf. Estos datos son:

- Nombre
- Fabricante
- Descripción

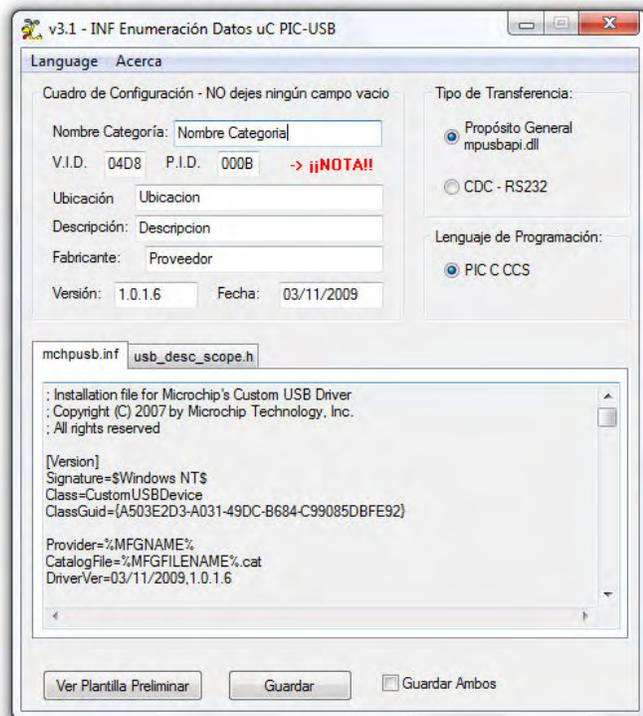
- VID (identificador de fabricante)¹⁶
- PID (identificador de producto)¹⁶

5.7.1 Automatización de datos Enumeración USB v 3.1¹⁷

Es un programa libre que permite introducir datos como Nombre, VID, PID, Fabricante, Descripción, entre otros, para configurar el instalador que necesita Windows cuando se conecta el dispositivo por primera vez.

Esta identificación se muestra al momento de instalación y cuando se accede al administrador de dispositivos en Windows.

Figura 17. Captura del programa Enumeración de datos USB



¹⁶ García Breijo, Eduardo. Compilador C CCS y simulador Proteus para microcontroladores PIC. Marcombo, Barcelona, España, 2008, p.256

¹⁷ Enumeración USB. Documentación HTML. Disponible en internet. <http://www.unpocodelectronica.net.au.net/generador-de-inf-para-los-drivers-usb-de-microchip/>

5.8 RADIOFRECUENCIA¹⁸

El término radiofrecuencia, también denominado espectro de radiofrecuencia o RF, se aplica a la porción menos energética del espectro electromagnético, situada entre unos 3 Hz y unos 300 GHz. Las ondas electromagnéticas de esta región del espectro se pueden transmitir aplicando la corriente alterna originada en un generador a una antena.

5.8.1 Clasificación¹⁹

La radiofrecuencia se puede dividir en las bandas del espectro contenidas en la tabla 1.

Tabla 1. Características de las bandas espectrales

Nombre	Abreviatura inglesa	Banda ITU	Frecuencias	Longitud de onda
			< 3 Hz	> 100.000 km
Extra baja frecuencia Extremely low frequency	ELF	1	3-30 Hz	100.000–10.000 km
Super baja frecuencia Super low frequency	SLF	2	30-300 Hz	10.000–1.000 km
Ultra baja frecuencia Ultra low frequency	ULF	3	300–3.000 Hz	1.000–100 km
Muy baja frecuencia Very low frequency	VLF	4	3–30 kHz	100–10 km

¹⁸ BOYLESTAD, Robert. Electrónica: Teoría de Circuitos y dispositivos electrónicos. 8 Ed. México: Pearson Education, 2003.

¹⁹ Radio frequencies of the astrophysically most important spectral lines. Documentacion HTML disponible en internet: <http://www.craf.eu/iaulist.htm>

Cuadro 1. Características de las bandas espectrales (continuación)

Baja frecuencia Low frequency	LF	5	30–300 kHz	10–1 km
Media frecuencia Medium frequency	MF	6	300–3.000 kHz	1 km – 100 m
Alta frecuencia High frequency	HF	7	3–30 MHz	100–10 m
Muy alta frecuencia Very high frequency	VHF	8	30–300 MHz	10–1 m
Ultra alta frecuencia Ultra high frequency	UHF	9	300–3.000 MHz	1 m – 100 mm
Super alta frecuencia Super high frequency	SHF	10	3-30 GHz	100–10 mm
Extra alta frecuencia Extremely high frequency	EHF	11	30-300 GHz	10–1 mm
			> 300 GHz	< 1 mm

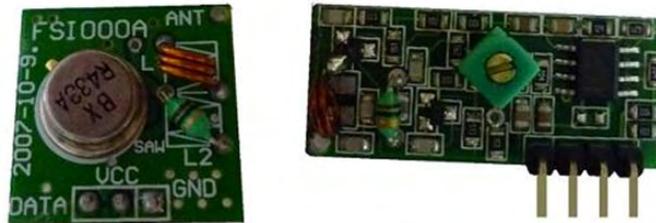
A partir de 1 GHz las bandas entran dentro del espectro de las microondas. Por encima de 300 GHz la absorción de la radiación electromagnética por la atmósfera terrestre es tan alta que la atmósfera se vuelve opaca a ella, hasta que, en los denominados rangos de frecuencia infrarrojos y ópticos, vuelve de nuevo a ser transparente. Las bandas ELF, SLF, ULF y VLF comparten el espectro de la AF (audiofrecuencia), que se encuentra entre 20 y 20.000 Hz aproximadamente. Sin embargo, éstas se tratan de ondas de presión, como el sonido, por lo que se desplazan a la velocidad del sonido sobre un medio material. Mientras que las ondas de radiofrecuencia, al ser ondas electromagnéticas, se desplazan a la velocidad de la luz y sin necesidad de un medio material.

5.8.2 Receptores de RF

Para comunicarse por medio de un enlace de RF es posible usar receptores integrados muy pequeños y de bajo costo, como el RWS-433, o el RXLC-434, y

otros similares, que trabajan en frecuencias de entre 303 y 433 Mhz. La elección de los transmisores dependerá de la distancia entre receptor y emisor, pero si se trata de áreas acotadas es posible utilizar los módulos transmisores hermanados con los anteriores, como el TWS-433 y el TXLC-434.

Figura 18. Módulos Rf Transmisor y receptor a 433 MHz



5.9 ELECTRÓNICA DE POTENCIA²⁰

Es la rama de la ingeniería electrónica que consigue adaptar y transformar la electricidad, con la finalidad de alimentar otros equipos, transportar energía, controlar el funcionamiento de máquinas eléctricas, etc.

Se refiere a la aplicación de dispositivos electrónicos, principalmente semiconductores, al control y transformación de potencia eléctrica. Esto incluye tanto aplicaciones en sistemas de control como de suministro eléctrico a consumos industriales o incluso la interconexión sistemas eléctricos de potencia.

El principal objetivo de esta disciplina es el procesamiento de energía con la máxima eficiencia posible, por lo que se evitan utilizar elementos resistivos, potenciales generadores de pérdidas por efecto Joule. Los principales dispositivos utilizados por tanto son bobinas y condensadores, así como semiconductores trabajando en modo corte/saturación (on/off).

5.9.1 Dispositivos Semiconductores de Potencia

Se han desarrollado una serie de dispositivos semiconductores de potencia, todos los cuales derivan del diodo o el transistor. Entre estos se encuentran los siguientes:

- Rectificador controlado de silicio (SCR)
- Triac

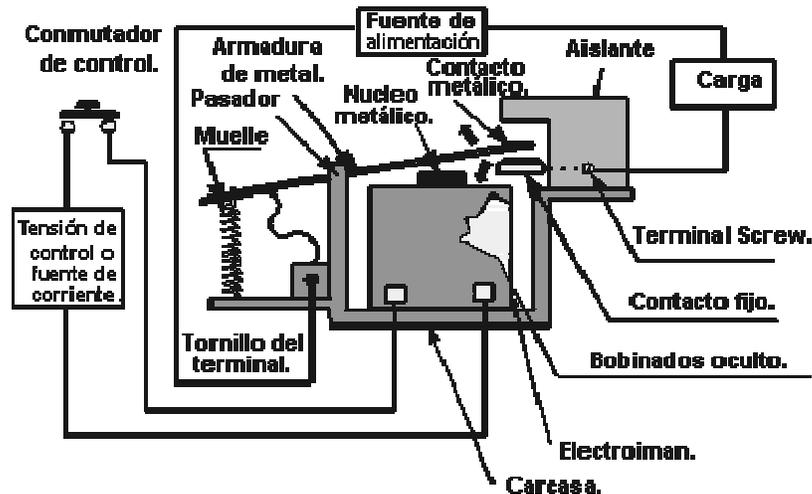
²⁰ MUHAMMAD, Rashid, Electrónica de Potencia, Circuitos, dispositivos y aplicaciones, Tercera Edición, Pearson Prentice Hall.

- IGBT
- IGTC

Aparte de estos existen dispositivos como el relé que realizan una operación similar.

El relé (figura 19) es un dispositivo mecánico capaz de comandar cargas pesadas a partir de una pequeña tensión aplicada a su bobina. Básicamente la bobina contenida en su interior genera un campo magnético que acciona el interruptor mecánico, que puede ser N.O ó N.C (normalmente abierto o normalmente cerrado). Ese interruptor es el encargado de manejar la potencia en sí, quedando al circuito electrónico la labor de mover la bobina. Permite así aislar mecánicamente la sección de potencia de la de control.

Figura 19. Esquema de un relé de tipo armadura



Relé tipo armadura. Disponible en http://www.ugr.es/~amroldan/enlaces/dispo_potencia/relés.htm

5.10 ANTECEDENTES

En diciembre de 2004 en Proceedings of the National Academy of Sciences ²¹, el profesor Jonathan R. Wolpaw, del Wadsworth Center de Nueva York, y sus colegas han publicado una investigación que demuestra lo que se ha avanzado en los métodos de electrodos externos. En ella, cuatro pacientes con diferentes tipos de parálisis pueden operar con un cursor en una pantalla fácilmente y con rapidez sin necesidad de implantes quirúrgicos.

²¹ Noticias de tecnología. Documentación HTML, disponible en internet. <http://www.laflecha.net/canales/ciencia/noticias/200502083>

Para aprender a variar la intensidad de las ondas del cerebro, cada sujeto ha aprendido a mover el cursor en cualquier dirección a diferentes velocidades.

Se trata de un sistema externo que constituye una alternativa de control y de comunicación para enfermos con parálisis severas, consecuencia de daños cerebrales o de la espina dorsal. Las señales emitidas por el cerebro son traducidas por un ordenador a señales físicas, como mover un cursor en una pantalla de ordenador.

En el llamado sistema Wadsworth, los usuarios llevan una especie de “gorro con un electrodo que detecta la actividad electroencefalográfica del cuero cabelludo y recoge las ondas específicas del cerebro. Un algoritmo adaptable analiza las señales y se centra en aquéllas que permiten a los pacientes un mayor control según van aprendiendo a usar sus pensamientos para dirigir el cursor informático a un objetivo determinado de la pantalla. Según va mejorando el aprendizaje, el algoritmo se adapta de nuevo.

El trabajo de estos científicos sugiere ya por lo tanto que un sistema de este tipo, que no requiere implantación quirúrgica de electrodos en el cerebro, podrá mejorarse hasta realizar tareas más sofisticadas, como el manejo de un brazo artificial o el de un ratón virtual de ordenador.

Otra referencia tomada es la del desarrollo de un sistema MONITOR EMG CON CONEXIÓN INALÁMBRICA ²², llevado a cabo por el estudiante Daniel Abad Navarro y dirigido por Francisco Carlos Bogónez Franco en la universidad Autónoma de Barcelona, en Septiembre de 2008, el cual trata de un dispositivo portátil de bajo consumo capaz de obtener 4 señales EMG de un paciente, las señales son acondicionadas, muestreadas y enviadas por medio inalámbrico o a través de una conexión serie hacia un PC, para allí ser procesadas mediante un programa diseñado específicamente para este proyecto.

A una referencia regional es el desarrollo del proyecto de grado DISEÑO E IMPLEMENTACIÓN DE UN MÓDULO ELECTROMECAÁNICO AUTÓMATA PARA LA REHABILITACIÓN FÍSICA DE LA MANO DERECHA AFECTADA COMO SECUELA DE UN ACCIDENTE CEREBROVASCULAR realizado por Carlos Andrés Criollo Paredes y David Eduardo López Pantoja del departamento de Ingeniería Electrónica de la Universidad de Nariño.

²² Navarro Daniel Abad. Monitor EMG con conexión inalámbrica, ingeniería Electrónica, Universidad autónoma de Barcelona. Septiembre de 2008

El proyecto se basa en el desarrollo de un dispositivo electromecánico que ayuda a los pacientes con ACV en la rehabilitación de la mano derecha, la cual se basa en la obtención y procesamiento de señales EMG, adquiridas del paciente por medio de electrodos superficiales.

De cada una de estas referencias se tomaron datos muy importantes para el pleno desarrollo del presente proyecto.

6. DESARROLLO

Este proyecto que pretende ayudar a mejorar la comunicación de individuos con ciertas limitaciones presenta las siguientes características:

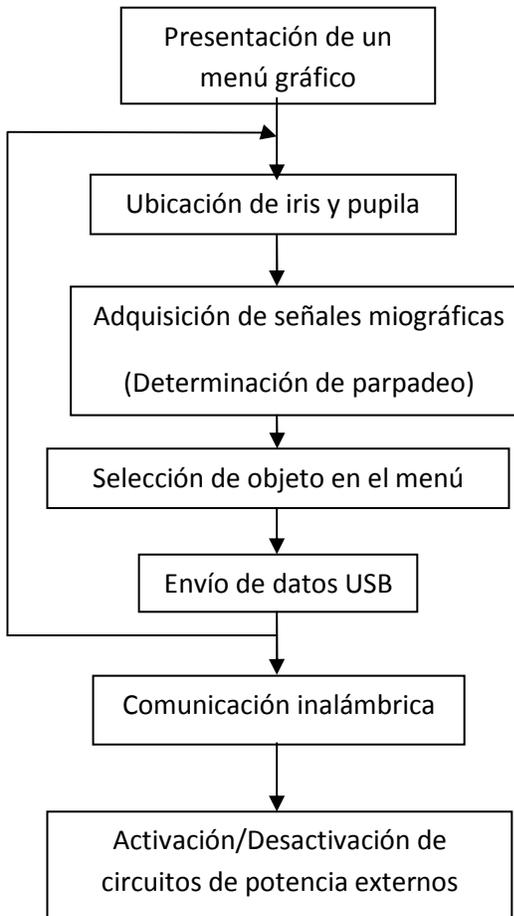
- El usuario o persona con limitación de movimiento de extremidades se debe ubicar frente a la pantalla de un computador para que pueda interactuar con un menú y al mismo tiempo con el entorno.
- La cámara web ubicada frente al individuo captará imágenes de su rostro para analizar mediante un software la posición de los ojos.
- La persona debe llevar sensores cerca a uno de sus ojos, para captar señales miográficas correspondientes a parpadeos cuando desee seleccionar un objeto del menú, a manera de un clic convencional.
- La interfaz de usuario muestra imágenes de un televisor, un radio y una lámpara, las cuales se pueden activar o desactivar mediante el parpadeo.
- El computador genera datos de activación de los dispositivos electrónicos, usando el puerto USB y circuitos de radiofrecuencia para encenderlos a una cierta distancia sin necesidad de cables.

El proceso del sistema que activa los dispositivos electrónicos mediante señales oculares se lleva a cabo como se indica en el diagrama de bloques de la figura 20.

6.1 ADQUISICIÓN DE SEÑALES MIOGRÁFICAS

La primera parte del proyecto es la adquisición de la señal miográfica del ojo para permitir su procesamiento en un PC y determinar si el usuario o paciente hace un parpadeo voluntario. Muchos fenómenos biomédicos generan un ruido acústico característico, su medición provee información acerca del fenómeno que lo produce. Una contracción muscular produce sonidos (ruido muscular) y debido a que la energía acústica se propaga a través del medio biológico, dicha señal se puede adquirir desde la superficie utilizando transductores acústicos como micrófonos. Un micrófono ubicado correctamente cerca al ojo, da excelentes resultados, y cuya información generada es fácilmente capturada por un canal de la tarjeta de sonido, para luego ser procesada.

Figura 20. Diagrama de bloques del sistema



6.1.1 Sensores

Debido a que el movimiento voluntario de los músculos produce diminutas señales de sonido, estas pueden ser captadas con micrófonos de alta sensibilidad, llegando así a obtener los resultados deseados.

El micrófono empleado es de tipo condensador o electrec, dadas sus ventajas de respuesta plana, gran fidelidad, buen comportamiento en agudos, buenas relaciones señal/ruido, poco sensibles a las vibraciones y reducido tamaño.

Para mantener fijo el sensor cerca al ojo se lo acondicionó con una espuma adhesiva, para evitar movimiento y falsas lecturas (figura 21).

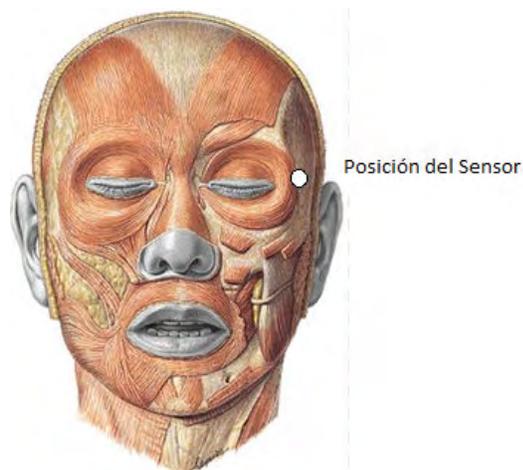
Figura 21. Sensor para la captación del parpadeo



a) Posición del sensor

La posición del sensor es muy importante en la recolección de información que indique un parpadeo. En esta etapa se realizó pruebas con el hardware y software para identificar las regiones musculares cercanas al ojo que permita detectar las señales de un parpadeo, llegando a la conclusión de que el sensor se debe ubicar aproximadamente a 1.5 cm del extremo saliente del ojo en el musculo orbicular (figura 22):

Figura 22. Ubicación del sensor



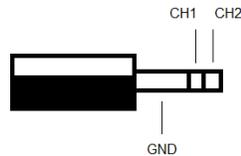
b. Número de sensores

Debido a que el proyecto solo hace uso de un parpadeo, únicamente se utilizó un micrófono como sensor para la ubicación en cualquier ojo, el cual corresponde a un canal.

c. Conexión del sensor al computador

La señal proveniente del micrófono fue llevada al computador mediante la conexión al canal CH1 de un plug (figura 23) y en el software se seleccionó el mismo canal para la adquisición. Cualquiera de los dos canales es válido, siempre y cuando en el software se escoja el mismo de la conexión física.

Figura 23. Canales de un plug estereofónico



6.1.2 Adquisición

Una vez capturadas, las señales fueron llevadas al computador, para ser visualizadas en pantalla, mediante la tarjeta de sonido que se incluye en todos los equipos actuales. El equipo que se utilizó para la adquisición y las demás etapas fue un computador portátil modelo Hewlett Packard dv4 1225 con tarjeta de sonido ATI RS690 HDMI @ ATI SB700 - High Definition Audio Controller.

Para que el computador pueda tratar los datos adquiridos, debe convertirlos de su estado original (analógico) al formato binario (digital). Aquí se realiza lo que se denomina muestreo, que es recoger la información y cuantificarla, es decir, medir la altura o amplitud de la onda. Este proceso se realizó a una frecuencia de muestreo de 11 KHz, parámetro que es incluido en software.

Para que la tarjeta de sonido pueda leer el puerto, en el programa se creó un objeto wavrecord(n,Fs), donde n es el número de muestras a una frecuencia Fs, que por defecto es 11025 Hz.

Con esta variable se logró graficar las señales provenientes del sensor, lo que implica que se ha producido movimientos de parpadeo, como puede apreciarse en las siguientes figuras desarrolladas en Matlab:

Figura 24. Dos parpadeos consecutivos

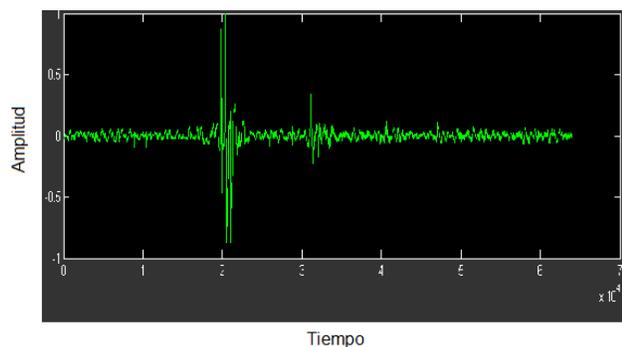


Figura 25. Cinco parpadeos consecutivos

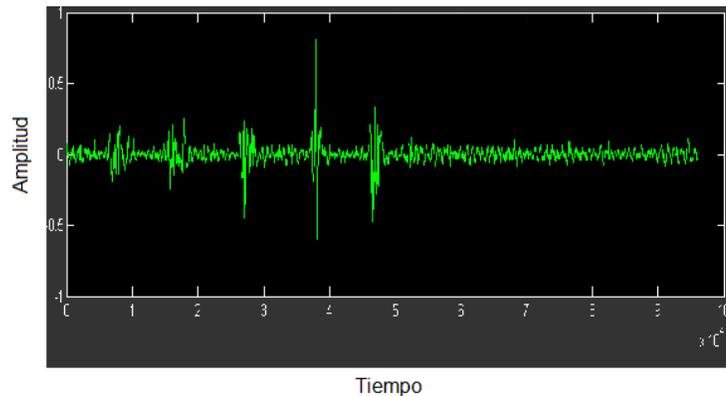
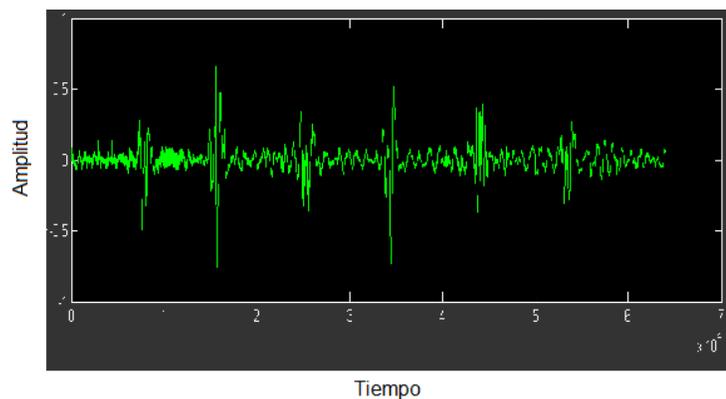


Figura 26. Seis parpadeos consecutivos



Como se puede observar, consecuencia del parpadeo se tiene señales con picos bastante claros, que son utilizados adecuadamente en otra etapa para seleccionar un objeto dentro de un menú gráfico.

Adicionalmente, se tiene que estos picos presentan una buena amplitud, debido a que los parpadeos son voluntarios, por eso un parpadeo natural no llevaría esa amplitud y por consiguiente se eliminaría el problema de seleccionar objetos de forma errónea e inesperada.

6.2 ALGORITMO DE UBICACIÓN DE LOS OJOS

La implementación de algoritmos en visión por computador resulta muy costosa en tiempo ya que se requiere de la manipulación de punteros, gestión de memoria, etc. MATLAB permite desarrollar algoritmos desde cero o trabajar con interfaces complicadas a bibliotecas externas. El lenguaje técnico, y cientos de funciones en cajas de herramientas (toolboxes) convierten a MATLAB en lo más adecuado para aplicaciones matemáticamente intensivas que requieran análisis de datos,

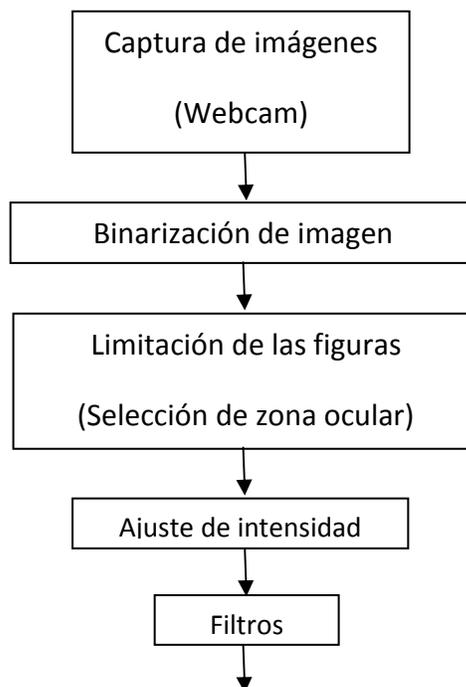
procesamiento de señales e imágenes, modelado de sistemas o técnicas numéricas avanzadas ²³.

El problema del tiempo puede ser resuelto si la implementación es realizada en MATLAB utilizando su toolbox de procesamiento de imágenes, con ello el tiempo de implementación se convierte en el mínimo con la confianza de utilizar algoritmos científicamente probados y robustos.

Por consiguiente los procesos de adquisición de datos del sensor, de ubicación de iris y pupila y algoritmos de procesamiento de datos fueron llevados a cabo mediante la implementación de un software m-file.

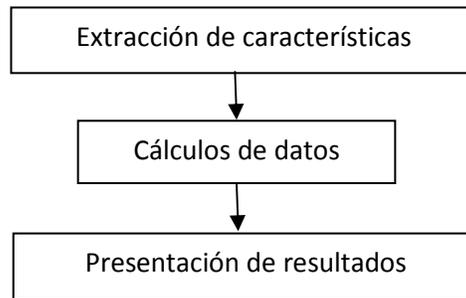
La etapa de ubicación de la pupila del ojo se ejecutó mediante un procesamiento de imágenes y la aplicación de varios procesos importantes: *captura de las imágenes, binarización de imagen, limitación de figuras, ajustes de intensidad, filtros, extracción de características, cálculos de datos y presentación de resultados.*

Figura 27. Diagrama de bloques lineal del sistema de ubicación de los ojos



²³ Part-Enander, E., Sjoberg, Melin, B. and Isaksson, P. The MATLAB Handbook Addison-Wesley, New York, 1996.

Figura 27. Diagrama de bloques lineal del sistema de ubicación de los ojos (continuación)



6.2.1 Captura de imágenes

Actualmente los computadores, principalmente los portátiles disponen de una unidad llamada webcam, que permite adquirir video e imágenes dentro de un espacio limitado, que por lo general lo adquiere desde la parte superior de la pantalla donde está ubicada la cámara. También se utilizan cámaras USB externas para la adquisición de información.

Inicialmente para el desarrollo de este proceso se hace uso de la instrucción *rostro= imread('cara.jpg')*, donde *cara.jpg* es una imagen de la región ocular tomada a una resolución de 160x120 y que está ubicada en el disco duro. Posteriormente, cuando se incluye la interfaz gráfica se crea un objeto denominado *camara = videoinput('winvideo', 2,'RGB24_160x120')*, *winvideo* contiene las características de la webcam que se esté utilizando. *RGB24_160x120* es el formato soportado por una webcam externa ubicada especialmente frente al ojo del usuario e indica la resolución, en este caso 160 pixeles de ancho por 120 pixeles de alto.

Para obtener una fotografía del anterior objeto, se usó la función *getsnapshot(camara)*, la cual devuelve una imagen en forma de matriz con las mismas dimensiones de la resolución mencionada anteriormente.

6.2.2 Ajustes de imagen para encontrar el ojo

Como el usuario siempre debe estar frente a la pantalla del PC, la cámara fue acondicionada para capturar la región ocular, y se limitó la imagen capturada de tal forma que se consiguió una matriz recuadro donde esté aproximadamente una región de cualquiera de los ojos (se escogió la región del izquierdo, debido a que el rostro es simétrico y los dos ojos se mueven al mismo tiempo y en las misma dirección, con lo que la utilización de una región u otra es indiferente). El comando para realizar este proceso es *ojo=rostro((15:100),(1:160),:)* donde *rostro* es la imagen capturada previamente, el primer intervalo limita horizontalmente y el segundo lo hace en forma vertical, los dos puntos (:) implica que toma los tres planos RGB de la imagen.

Para obtener una imagen en blanco y negro del ojo se recurrió al proceso mostrado en la figura 28.

Figura 28. Procesos para ajustar la imagen



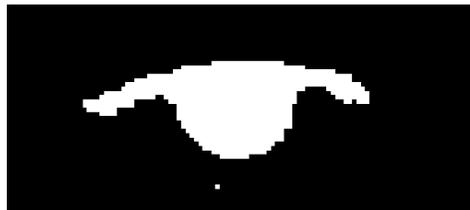
Figura 29. Recuadro del ojo en escala de grises



Figura 30. Imagen del ojo en blanco y negro



Figura 31. Imagen del ojo en blanco y negro complementada



6.2.3 Limites de figuras

Los bordes de las imágenes pueden contener información importante que permitirá encontrar algunos puntos para la localización del iris y los extremos del ojo que se analiza. Al utilizar una cámara USB externa se logró obtener los puntos izquierdo y derecho (xi y xf) del ojo de una forma sencilla, puesto que con la ubicación de la cámara siempre se tendrá los límites del ojo en los límites de la imagen capturada.

6.2.4 Filtros

Como las imágenes binarizadas pueden tener ruido, que son pixeles u grupos de ellos que no son deseados para el análisis de imágenes, se los eliminó haciendo uso de `bwareaopen(adj_gris,10)`, el número 10 implica que se van a eliminar todos los objetos de menos de 10 pixeles (figura 32). Igualmente, para tal fin, se usó la instrucción `se = strel('disk',2)` que hace posible que la imagen pierda pixeles de ruido y mantenga aquellos que sean parte importante de la imagen. Con `bw = imfill(bw,'holes')`, se encuentra algunos agujeros, cuyas propiedades pueden ser usadas para estimar el área encerrada por los límites de los objetos que contenga la imagen (figura 33).

Figura 32. Imagen filtrada



Figura 33. Imagen sin ruido



6.2.5 Determinación del objeto de área máxima

Adicionalmente se puede obtener algunas mediciones de los objetos encontrados, como son: área, centroide, ejes mayor y menor, etc. Las propiedades que se quiere calcular se van poniendo una detrás de otra, seguidas de una coma, dichas propiedades son:

- **Área:** Calcula el área en píxeles cuadrados de la región.
- **BoundingBox :** Calcula la posición y dimensiones del mínimo rectángulo que envuelve a la región.
- **Centroid:** Posición del centroide de la región.
- **ConvexHull:** Matriz con la posición de los píxeles que definen el casco convexo que envuelve la región.
- **ConvexImage:** Imagen binaria con la forma del casco convexo.
- **Eccentricity:** Número escalar que da la excentricidad de la imagen.
- **EulerNumber:** Escalar que da el número de Euler de la región.
- **MayorAxisLength:** Longitud del eje de mayor longitud de la región.
- **MinorAxisLength:** Longitud del eje de menor longitud de la región.

6.2.6 Determinación de coordenadas

La función $s=regionprops(bw,'Centroid')$, permitió encontrar la posición central de la región de interés en una estructura, que corresponde al centro del iris del ojo y al mismo tiempo al punto central de la pupila debido a que son concéntricos.

Una estructura es un conjunto de datos, que en dicha forma no se los puede analizar cómodamente, para solucionar este problema se transformó esa estructura a un arreglo con la sencilla función $sArray=struct2array(s)$, consiguiendo así los puntos requeridos en esta etapa del proyecto que forman las coordenadas de la pupila o el iris:

$$x= 1+ sArray(1,1)$$

$$y= 15+ sArray(1,2)$$

Como se observa en las funciones, se agregó los numeros 1 y 15, puesto que, como se mencionó con anterioridad, se realizó un recorte de la imagen, donde se quitó el mismo valor de uno de sus extremos.

6.2.7 Ubicación de los puntos de la imagen

Una vez encontradas las coordenadas de la pupila o el iris y los extremos laterales de los ojos, conseguidos previamente, se procedió a graficarlos, dando como resultado la localización de las características oculares en la imagen del rostro.

Figura 34. Puntos característicos del ojo en la imagen inicial



Para graficar los puntos característicos se usó la siguiente secuencia:

hold on

```
plot(x,y,'ro','LineWidth',2)
```

```
plot(x,y,'r*','LineWidth',2)
```

```
plot(xi,y,'b*','LineWidth',2)
```

```
plot(xf,y,'b*','LineWidth',2)
```

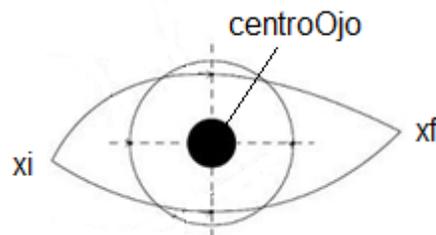
hold off

x,y,x_i,x_f , son respectivamente coordenada x de la pupila, coordenada y de la pupila, extremo inicial y extremo final del ojo. $LineWidth$ es el ancho del punto, para este caso es 2. La parte *hold on* y *hold off* permiten dibujar sobre una imagen cargada anteriormente.

6.2.8 Determinación de la posición de los ojos

El último proceso en esta etapa consiste en presentar los resultados diciendo en qué posición están localizados los ojos en un instante pequeño. Para ello se tuvo en cuenta un punto denominado *centroOjo*, calculado a partir de la ecuación 1.

Figura 35. Puntos característicos del ojo



$$centroOjo = \frac{x_f - x_i}{2} + x_i \quad (1)$$

Con este punto encontrado, se presentó los resultados de tal forma que si la coordenada x de la pupila era mayor al centro del ojo, adicionado 20 puntos ó

pixeles, entonces el ojo estaba hacia la derecha, en caso contrario estaba en el centro.

Lo mencionado anteriormente fue representado en el software de la siguiente manera:

```
if x>CentroOjo+4
```

```
    disp('Derecha')
```

```
else
```

```
    disp('Centro')
```

```
end
```

La instrucción disp muestra en la pantalla un mensaje correspondiente: Derecha o Centro.

6.3 DESARROLLO DE LA INTERFAZ GRÁFICA

Una interfaz es un dispositivo que permite comunicar dos sistemas que no hablan el mismo lenguaje. Una interfaz se refiere a la cara visible de los programas tal y como se presenta a los usuarios para que interactúen con la máquina. La interfaz gráfica implica la presencia de un monitor o pantalla constituida por una serie de menús e iconos que representan las opciones que el usuario puede tomar dentro del sistema.

Puesto que el proyecto consta de un menú con tres opciones gráficas, se utilizó un Axes, que es un cuadro donde se muestra las imágenes, y en él se cargara la imagen correspondiente a un televisor, a un radio y a una lámpara. Este se denominó im1 y se ubicó en el centro de la pantalla, ya que representa el objeto que está listo para ser activado o desactivado (figura 36).

Un segundo Axes denominado im2 corresponde al siguiente elemento del menú, es más pequeño que el anterior y se ubicó a la izquierda del mismo.

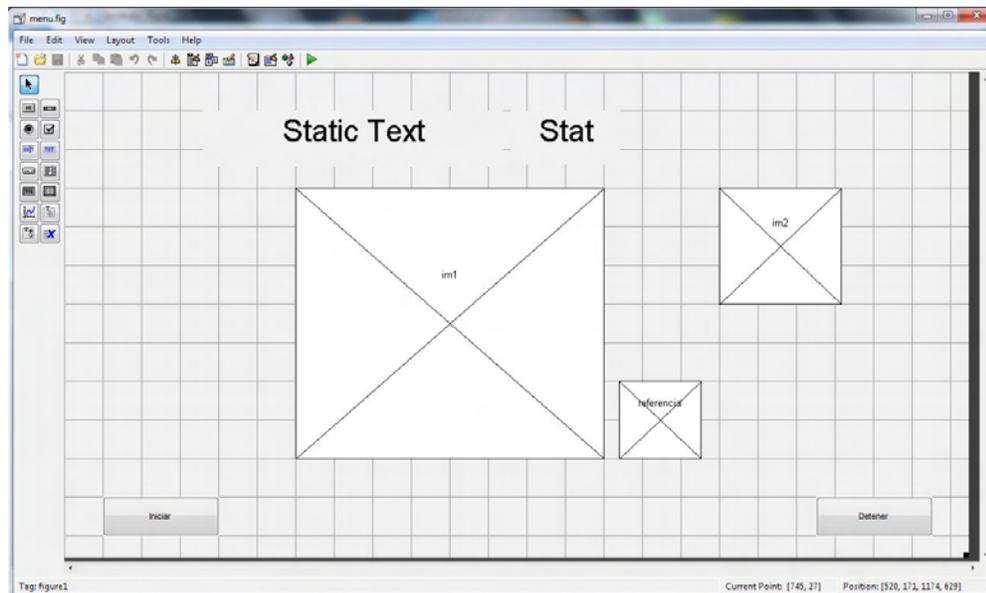
Un último Axes nombrado *referencia* se utilizó para que el usuario se ubique de la manera correcta frente a la pantalla del computador y el sistema pueda ejecutarse óptimamente. Este es el más pequeño de los Axes y se colocó a la izquierda de im1 y en él debe aparecer el ojo izquierdo del usuario.

Dos Static Text ubicados en la parte superior indican el respectivo nombre el objeto seleccionado y el estado (on-off) del mismo.

Para cargar la webcam y todos los componentes se colocó un PushButton denominado Iniciar y otro con nombre Detener, que como lo indica detiene la

adquisición de imágenes por parte de la cámara web y todos los procesos relacionados con el programa.

Figura 36. Detalles de la interfaz gráfica



6.3.1 Programación

Cuando se ejecuta el programa por primera vez, se carga las imágenes iniciales y los estados de todos los dispositivos se presentan como off o apagados. Se definió una variable cámara como global puesto que será utilizada en otros eventos.

Como los estados de los dispositivos inicialmente están en off se los igualó a 0 y corresponden respectivamente al televisor, al radio y la lámpara:

```
handles.tele=0;
```

```
handles.radi=0;
```

```
handles.lam=0;
```

El estado del ojo es 0 cuando está en el centro y 1 cuando este a la derecha (este estado indica que el usuario observa la imagen del objeto correspondiente al siguiente dispositivo en la parte derecha y por consiguiente el menú desplazara cada vez que observe a esa posición). Inicialmente al cargar el programa *handles.estadoOjo* será 0.

Del mismo modo, la imagen inicial que se carga en el Axes *im1* sera *handles.imagIni=1*, donde el número 1 corresponde al televisor, en el caso del radio será 2 y la lámpara 3. En el Axes *im2* seguirá la imagen del radio, que es el siguiente dispositivo del menú. La imagen de la lámpara no aparecerá hasta que se desplace el menú.

En las cajas de texto ubicadas en la parte superior se escribió el primer dispositivo y sus estado inicial así:

```
set(handles.dispositivo,'String','Televisor');
```

```
set(handles.estado,'String','Off');
```

Lo primero que aparecerá en el Axes referencia será un cuadro de color blanco que indica que aún no se ha iniciado la aplicación.

Con tan solo hacer clic en el botón Iniciar, el usuario queda libre de utilizar cualquier medio de entrada como el teclado o el ratón, puesto que el sistema trata de interactuar con la maquina únicamente con la posición de los ojos y el sensor de parpadeo.

En este proceso también se declararon como datos globales a las variables cámara, foto, activar y stopbut. La variable cámara inicia y configura la webcam del equipo, la variable foto permite capturar una imagen de la cámara usando el comando *getsnapshot*, *activar* y *stopbut* permiten mantener como verdadero o falso algunas acciones posteriores.

Luego de ello, se adicionó al programa *.m* el código correspondiente al algoritmo de ubicación de los ojos, detallado anteriormente, con el cual se conseguía determinar si los ojos observaban a la derecha o al centro, generando los respectivos datos.

Una vez determinadas estas acciones, se procedió a actuar sobre las imágenes que aparecerán en el menú con sus estados específicos, esto se logró con la sintaxis *switch* que es un medio de ejecutar el código condicionalmente. En particular, *switch* ejecuta declaraciones seleccionadas de un número arbitrario de alternativas denominadas *casos*.

En su sintaxis básica, si la variable después de *switch* es igual a alguna de las alternativas se ejecuta el código correspondiente al caso, entonces se utilizó con *switch handles.imagIni*, donde la segunda variable representa la imagen principal que aparece en el Axes más grande. Seguido de esto se colocó las diferentes alternativas que corresponde a *case 1*, *case 2* y *case 3* pertenecientes respectivamente a los casos en que se muestra el televisor, el radio o la lámpara.

Cabe resaltar que en el momento que se presiona sobre el botón Iniciar se muestra una serie de imágenes correspondientes al ojo derecho que sirve de

referencia para la ubicación del usuario. Dicha secuencia de imágenes aparecen en el Axes denominado referencia.

Para finalizar esta etapa, en la programación en el botón Detener se incluyó las variables globales activar como falso y stopbut como verdadero que inicialmente tenían valores opuestos. Al presionar sobre el botón Detener, este se deshabilita con `set(handles.detener,'Enable','off')` y activa el de Iniciar con `set(handles.iniciar,'Enable','on');`

6.3.1.1 Determinación de un parpadeo

El parpadeo implica que el usuario desea seleccionar el objeto del menú que está ubicado en el cuadro principal, esto quiere decir que mediante un parpadeo se cambia de estado del dispositivo determinado y actúa sobre los dispositivos a distancia prendiéndolos o apagándolos.

Mediante la ubicación exacta de los sensores y el desarrollo del software se obtiene con detalle los momentos de un parpadeo, pero en esta etapa solo se trata de obtener los datos de la señal y determinar mediante la búsqueda de valores máximos la producción de un parpadeo.

Esto se logró con una frecuencia de muestreo $F_s = 11025$, y para obtener más información, se utilizó 10000 muestras de la señal de audio capturada por la tarjeta de sonido del computador. Los comandos utilizados fueron `SenalOjo=wavrecord(10000,Fs,'double')` y `SenalMax=max(SenalOjo)`, el tipo doble implica que la información adquirida varía entre -1 y +1 y con max se obtuvo el valor máximo con el que se puede determinar el parpadeo cuando sobre pase un límite de 0.2.

Una vez más se utilizó la función switch, pero en esta ocasión para definir es estado del dispositivo. Con cada parpadeo, el elemento principal del menú cambia de estado, es decir de encendido a apagado y viceversa. La asignación del texto que determina el estado viene dada por `set(handles.estado,'String','Estado')`, donde Estado puede ser on-off.

De acuerdo a lo descrito anteriormente y teniendo en cuenta que la interfaz desarrollada interactúa con un sistema y no utiliza en toda su ejecución dispositivos convencionales como ratón y teclado se la puede clasificar como una Interfaz Natural de Usuario o NUI.

6.4 INTERFAZ USB

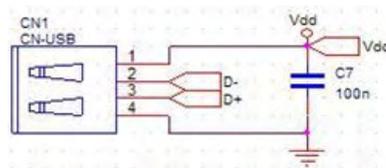
Buscando entre distintas opciones para poder comunicar el software con el medio exterior se llegó a la conclusión de que una interfaz con USB resulta muy conveniente, puesto que los costos no son tan elevados como los serían con módulos como el Bluetooth y que además son difíciles de conseguir en el mercado

regional. Teniendo en cuenta lo anterior se optó por implementar un circuito USB para la salida de los datos generados por el software del proyecto.

6.4.1 Diseño del hardware

Para iniciar con este tema se va a tener en cuenta que la alimentación del dispositivo será a través del bus USB, que suministra una tensión de 5 V, con esto se soluciona el problema de utilizar fuentes externas, las cuales pueden generar defectos en la transmisión de radiofrecuencia en la siguiente etapa. En la figura 37 se muestra cómo deben estar conectados los pines de datos del microcontrolador para que la comunicación sea posible.

Figura 37. Pines de alimentación y datos del puerto USB



El pin 1 corresponde a la alimentación Vcc (el cable es de color rojo), los pines 2 y 3 son de datos D-(blanco) y D+(verde) respectivamente, el pin 4 se lo conecta como tierra (negro).

Existen dos tipos de conectores USB, tipo A y tipo B. El que se ha utilizado es de tipo B (figura 38), por lo que el cable empleado para conectar el dispositivo con el computador será tipo A - tipo B.

Figura 38. Conector tipo B



En cuanto al microcontrolador, para su elección se consideró algunas características fundamentales:

- Debe disponer de un puerto de comunicaciones USB.
- La longitud de palabra será de 8 bits ya que no se maneja gran cantidad de información.
- La memoria de programa debe ser Flash.
- Número de entradas y salidas reducido.

Teniendo en cuenta lo anterior, se ha optó por el PIC18F2550 de encapsulado tipo DIP, ya que dispone de un puerto USB que puede funcionar tanto a baja velocidad (1,5 Mbps) como a velocidad completa (12 Mbps) y cuya hoja de datos se muestra en el Anexo A.

Las letras que hay en los diferentes modelos de cada PIC, en este caso la F (18F2550), indican el tipo de memoria de programa que utiliza. En esta cuestión se tiene una memoria de tipo Flash, que es una memoria no volátil, de bajo consumo y que se puede escribir y borrar eléctricamente.

Dependiendo de si se utiliza USB de baja velocidad o de velocidad completa, la señal de reloj del módulo USB deberá ser de 6 MHz o de 48 MHz, respectivamente, por tanto en el sistema se usara el de 48 MHz, la que será configurada en el firmware, mediante la utilización de un cristal de 12MHz.

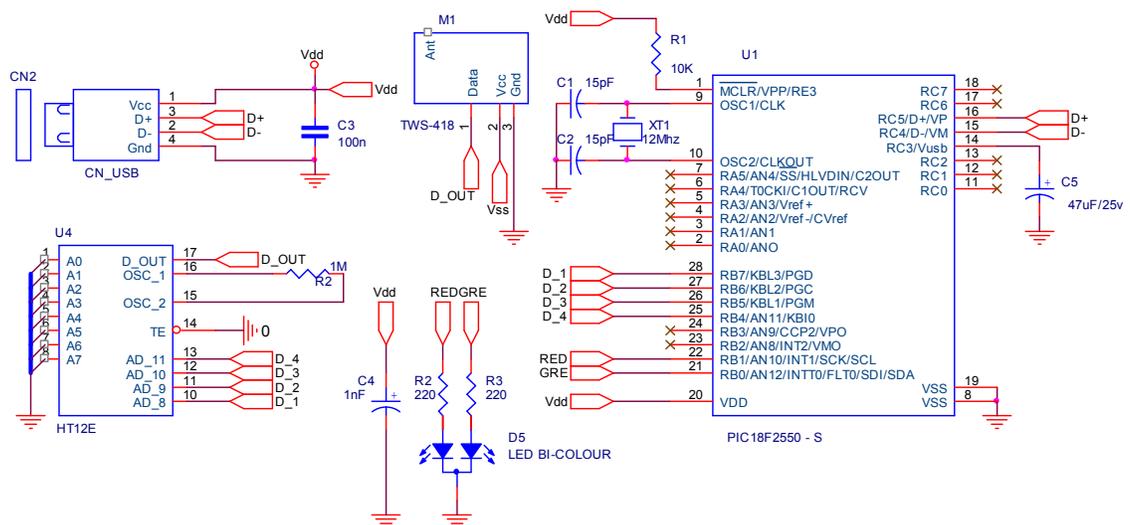
De acuerdo con la hoja de datos de este microcontrolador PIC los valores de los condensadores que acompañan al cristal deben ser cercanos a 27 pF, y puesto que estos valores son orientativos se utilizó dos de 15 pF, sin que se alterara el funcionamiento.

Adicionalmente una configuración típica para este circuito consta de un capacitor de 47 uF conectado a V_{USB} con el que se garantiza el óptimo funcionamiento del regulador interno de 3.3V. Y una resistencia de 10KΩ conectada a MCLR permite inicializar el PIC de forma correcta.

El diagrama de conexión del hardware se muestra en la figura 39.

El diseño del circuito impreso o PCB se realizó en el programa EAGLE 5.7 y su diagrama en Mirror se encuentra en Anexo D.

Figura 39. Diagrama de conexión del hardware



De acuerdo a las características eléctricas de los componentes que conforman esta etapa se logró determinar los parámetros máximos de operación del sistema así:

PIC18f2550 (Anexo A)

Voltaje de operación: 2-5.5 V

Corriente máxima en todos los puertos: 200 mA

HT12E (Anexo B)

Voltaje de operación: 2.4-12 V

Corriente máxima de operación: 1.6 mA

Transmisor 433

Voltaje de operación: 3-12 V

Corriente máxima de operación: 45 mA

LEDs (2)

Corriente de Operación para encendido normal: 20 mA

Sumando los valores máximos se tiene una corriente máxima de 286.6 mA suficiente para la especificación USB 2.0 que soporta un máximo de 500 mA.

6.4.2 Programación del Firmware

El firmware corresponde al programa que será grabado en el microcontrolador y tiene todas las instrucciones necesarias para comunicarse con el Host o computador y aquellas que gestionan la activación de los puertos del PIC.

Para la programación del firmware del PIC se optó por utilizar el compilador CCS versión 4.068 ya que es una herramienta que cuenta con librerías con métodos para funciones específicas del PIC como conversión A/D, entrada y salida, configuración de timers, etc.

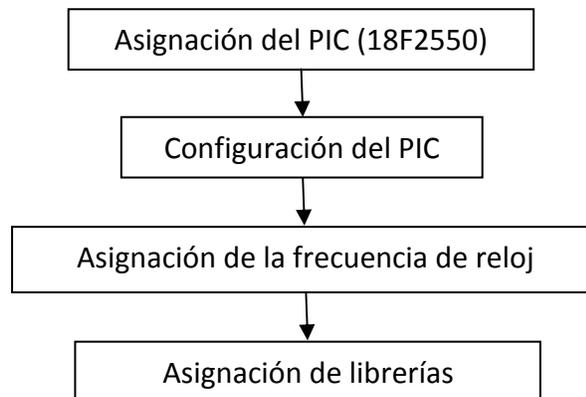
Microchip permite mediante la programación de sus PIC realizar uno o varios puertos COM virtuales e incluso la realización de USB 2.0 real mediante la CDC (Communication Device Class), clase que se utilizó en esta etapa del proyecto, puesto que con esto se disminuye el software al tratar el puerto USB como uno serial y se mantiene la misma la placa de hardware.

Una clase USB es un grupo de dispositivos (o interfaces dentro de un dispositivo) con ciertas características en común. Típicamente, dos dispositivos pertenecen a la misma Clase si ambos utilizan formatos similares en los datos que reciben o

transmiten, o si ambos utilizan una misma forma de comunicarse con el sistema. Tal es el caso de esta clase que permite a la función (ó dispositivo) comunicarse con el COM virtual a través del controlador HOST de USB en el computador.

El programa en sí, se inició de acuerdo al diagrama de la figura 40.

Figura 40. Diagrama de bloques del código inicial del PIC



Entre las funciones más importantes que se utilizó están:

Usb_init(): Inicializa el hardware USB. Espera en un bucle infinito hasta que el periférico USB es conectado al bus. Habilita y utiliza las interrupciones USB,

usb_cdc_init(): Configura la velocidad y parámetros de la comunicación.

Usb_task(): Cuando el PIC es conectado o desconectado del bus, esta función inicializa el periférico USB o resetea el periférico.

Usb_enumerated(): Devuelve un TRUE si el dispositivo ha sido enumerado por el PC, y en este caso, el dispositivo entra en modo de operación normal y puede enviar y recibir paquetes de datos.

Dentro de las librerías para CDC, se utilizó una función específica:

Usb_cdc_getc(c): Recibe un carácter del buffer de transmisión, y en caso de estar vacío espera hasta que se reciba.

6.4.2.1 Generación de las librerías `usb_cdc.h` y `usb_desc_cdc.h`

La librería `usb_cdc.h` es la misma que cuenta el compilador en sus archivos y de esta cabe destacar algunas instrucciones importantes:

```
usb_cdc_line_coding.dwDTERate=9600
```

Permite establecer la velocidad de comunicación entre el computador y el dispositivo USB. Las siguientes líneas permiten configurar respectivamente el formato, la paridad y el número de bits de los datos:

```
usb_cdc_line_coding.bCharFormat=0;
```

```
usb_cdc_line_coding.bParityType=0;
```

```
usb_cdc_line_coding.bDataBits=8;
```

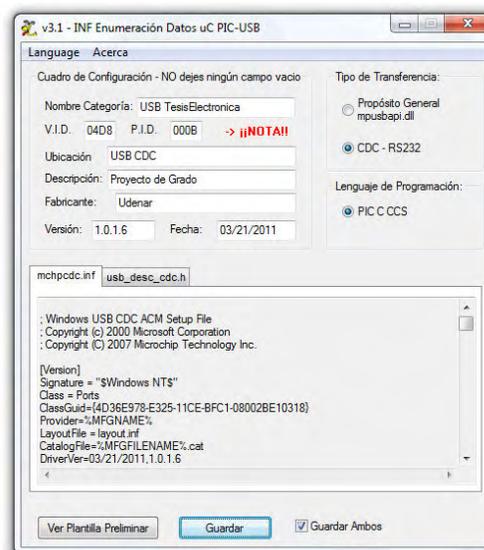
Las librerías `usb_cdc.h` y `usb_desc_cdc.h` permiten la comunicación a través del puerto USB y además son las encargadas de la creación de un puerto COM virtual en el PC.

`usb_desc_cdc.h` identifica al dispositivo con el Host. Es decir le indica todas las características para que el Host pueda identificar al dispositivo como un puerto de comunicaciones virtual. Esta librería se la generó con el programa `v3.1-INF Enumeración de datos uC PIC USB` creado por el usuario Palitroquez, del foro `todopic.com` (figura 41).

Básicamente en el programa se indicó los siguientes datos:

- **Nombre Categoría:** USB TesisElectronica
- **V.I.D. y P.I.D.:** 04D8 y 000B respectivamente.
- **Descripción:** Proyecto de grado.
- **Fabricante:** Udenar
- **Tipo de transferencia:** CDC – RS232.
- **Lenguaje de programación:** PIC C CCS.

Figura 41. Software de Enumeración de Datos PIC USB



6.4.2.2 Generación del driver

El driver es el lazo de comunicación entre el microcontrolador, que mediante sus descriptores establece comunicación con el HOST. Dicho driver también fue creado a partir del mismo software INF Enumeración de Datos, y presenta las siguientes partes fundamentales:

- Dos secciones, una para Windows 2000/XP/Vista de 32 bits y otra para Windows a 64 bits, el cual funciona óptimamente en Windows 7. Cualquiera de las dos partes permite instalar el driver en el directorio correspondiente una vez que el dispositivo es enumerado.
- La parte más importante es una sección que enlaza los drivers del PIC y el PC mediante sus VID y PID, la cual se denota así: %DESCRIPTION % = DriverInstall, USB\VID_04D8&PID_000B.
- Y por último se definió los String Definitions que no son más que cadenas de datos que identificarán al microcontrolador dentro del administrador de dispositivos:

[Strings]

MFGFILENAME="mchpcdc"

DRIVERFILENAME ="usbser"

MFGNAME="Udenar"

INSTDISK="Proveedor - Disco Instalación"

DESCRIPTION="Proyecto de Grado"

SERVICE="USB RS-232 Emulation Driver"

6.4.3 Programación en el software

Para usar este puerto es necesario establecer los parámetros de su funcionamiento, como baudrate, bits de datos, bit de parada, etc.

Con *SerPIC* = *serial('COMx')* se creó un puerto serie asociado al puerto COMx que se lo determina en el momento de instalar el dispositivo USB.

BaudRate 9600, DataBits 8, Parity none, StopBits 1, FlowControl none son las configuraciones para establecer la comunicación entre el Computador y el sistema USB.

fopen(SerPIC) conecta el puerto con el dispositivo y *fprintf* envía los datos generados por el programa, que en esta caso son letras, que reconocerá y

procesará el PIC. Luego de realizar la comunicación se cerró el puerto con *fclose(SerPIC)* y *delete(SerPIC)*.

6.5 ETAPA DE RADIOFRECUENCIA

La comunicación inalámbrica o sin cables es aquella en la que extremos de la comunicación (emisor/receptor) no se encuentran unidos por un medio de propagación físico, sino que se utiliza la modulación de ondas electromagnéticas a través del espacio. En este sentido, los dispositivos físicos sólo están presentes en los emisores y receptores de la señal, entre los cuales se encuentran: antenas, computadores portátiles, PDA, teléfonos móviles, etc.

En esta parte del proyecto que implica el envío de información a través de radiofrecuencia se utilizó los módulos de RF a 433 MHz que son de bajo costo y muy buen desempeño y cuyas características son:

Transmisor

- Voltaje de operación: 3-12 Vdc
- Corriente de operación: 5-45 mA
- Frecuencia de operación: 433 MHz
- Rango de Transmisión: 1000m
- Dimensiones: 19mm x 38mm (figura 42)

Receptor

- Voltaje de operación: 5Vcd
- Frecuencia de operación: 433 MHz
- Dimensiones: 14mm x 38mm (figura 43)

Figura 42. Distribución de los pines del módulo transmisor

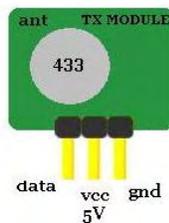
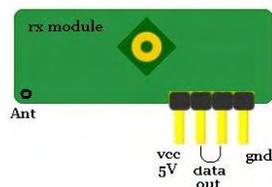


Figura 43. Distribución de pines del módulo receptor



Para lograr la comunicación entre dispositivos fue necesario usar un par de chips para codificación y decodificación, un HT12E y HT12D, respectivamente. Este juego de integrados codifica y decodifica una palabra de 12 bits, compuesta por una dirección de 8 bits y una sección de datos de 4 bits. Con esta cantidad de bits se pueden comandar 256 dispositivos diferentes, enviándoles hasta 16 comandos distintos a cada uno.

A continuación se pueden observar los circuitos utilizados.

Figura 44. Transmisor para un enlace de RF

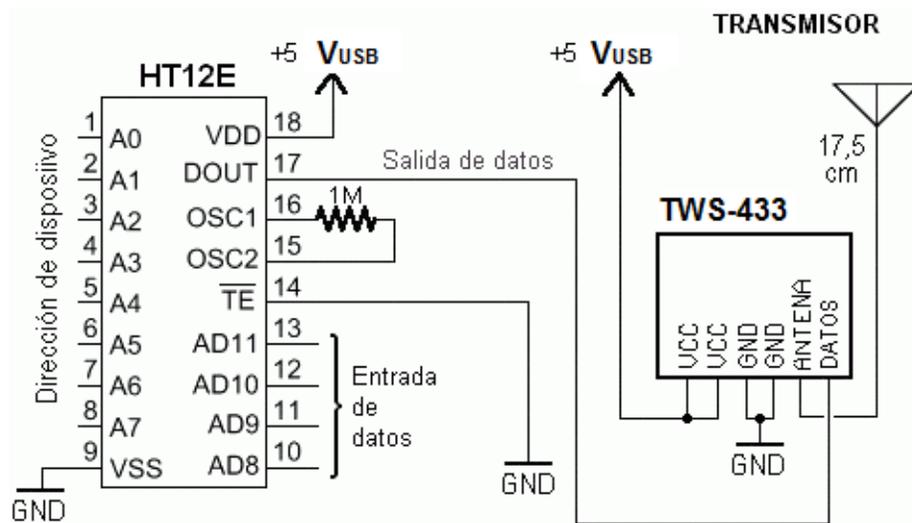
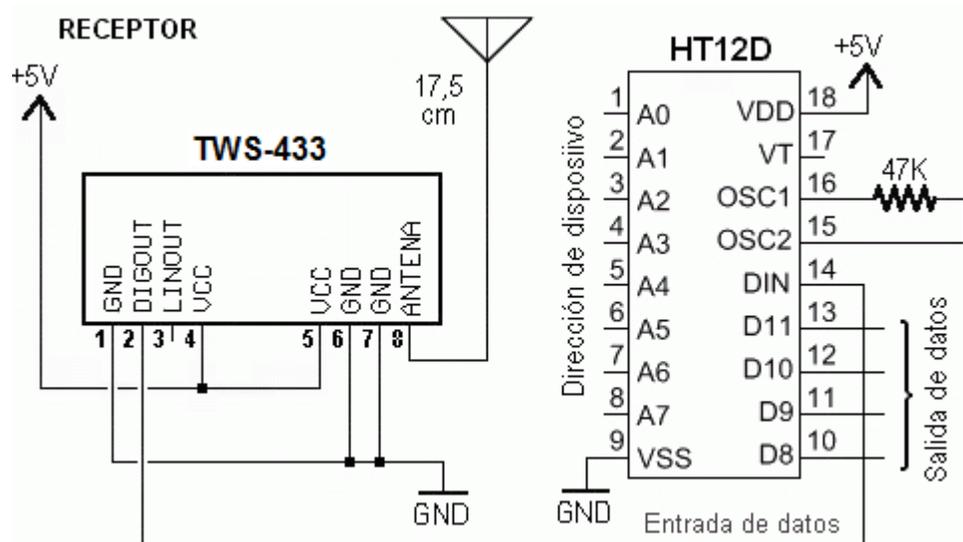


Figura 45. Receptor para un enlace de RF



El circuito transmisor permite el uso de una tensión de alimentación entre 5V y 12V, pero en este caso se dispone de los 5V provenientes del puerto USB. El receptor, por las características técnicas del chip decodificador HT12D, debe funcionar exclusivamente con 5V, obtenidos a partir del regulador 7805 y un adaptador de 9V.

De antena, tanto en el receptor como en el transmisor, de acuerdo a configuraciones típicas, se utilizó un alambre de 17 cm. de longitud.

La dirección de dispositivo (que se ingresa a través de las patas 1 a 8 en ambos chips) se estableció conectando todos estos pines a tierra, lo que indica la dirección 0, y lo cual permite la correcta comunicación entre los puntos transmisor y receptor.

6.6 ETAPA DE POTENCIA

El objetivo de esta fase consiste en controlar equipos que manejan potencias mayores a las utilizadas en los circuitos y puertos del computador, por ejemplo, la lámpara que se controlará generalmente disipa 100 W, por lo que consume aproximadamente 1 A, corriente que es muy elevada y de tipo alterna.

Para solucionar el inconveniente del control de los dispositivos de potencia, en los que se incluye un radio, un televisor y una lámpara, existen diferentes posibilidades como bloques con transistores de potencias, relés, TRIACs, entre otros, de los cuales se utilizó los relés de tipo armadura JQC-3F-1C-12VDC debido a sus características especiales:

- Aislamiento entre los terminales de entrada y de salida.
- Adaptación sencilla a la fuente de control.
- Posibilidad de soportar sobrecargas, tanto en el circuito de entrada como en el de salida.
- Cerrado a las influencias exteriores por su recubrimiento plástico.
- Bajo costo
- Corriente máxima en contactos de 15A
- Vida útil mecánica de 1×10^6 ciclos
- Vida útil eléctrica de 2×10^5 ciclos

Para controlar los tres relés correspondientes a cada dispositivo fue necesario la implementación de una etapa con transistores de tipo NPN de referencia 3904 que manejan una corriente de colector de 200 mA, con los que se puede accionar las bobinas de los relés (las cuales experimentalmente necesitan 22,1 mA para operar), manteniendo una pequeña corriente en su base proveniente del decodificador de la fase de radiofrecuencia, para que así actúen como interruptores.

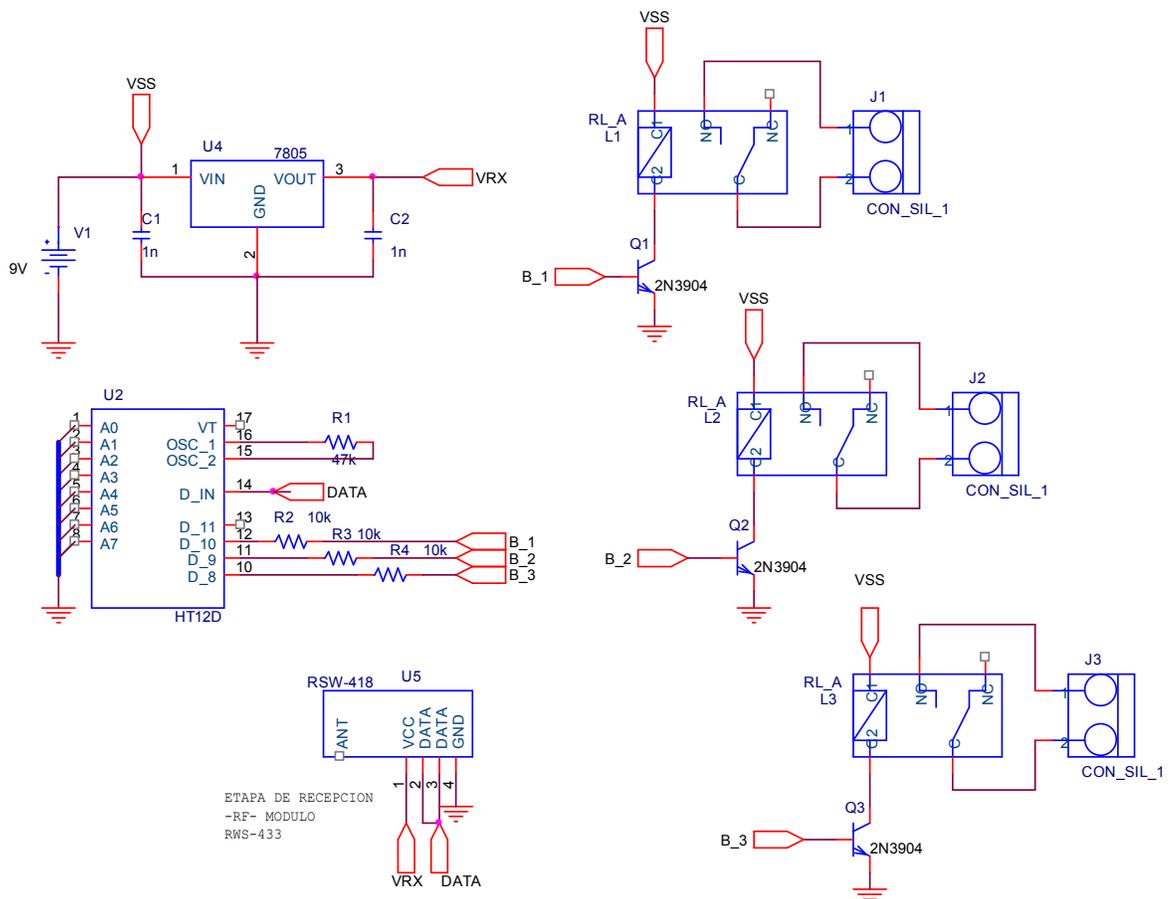
Los contactos de los relés son los encargados de manejar directamente los dispositivos de potencia, cerrándolos para encender el artefacto o abriéndolos para apagarlo. Para mayor comodidad del usuario dichos terminales de los contactos se los conectó a un tomacorriente convencional, para facilitar la manipulación de los dispositivos.

También se incluyó un fusible de protección para evitar que el equipo se dañe en caso de sobrecargas.

El diagrama de conexión se muestra en la figura 46.

El circuito PCB en mirror que incluye en una misma tarjeta la etapa recepción de radiofrecuencia y potencia de se encuentra en el Anexo F y la placa ensamblada en el Anexo G.

Figura 46. Diagrama de conexión del hardware



7. RESULTADOS

7.1 SOFTWARE DE DETECCIÓN DE POSICIÓN DE LA PUPILA

Se utilizó un computador portátil Hewlett Packard 1225dx, con un sistema operativo Windows 7 Professional de 64 bits y una memoria RAM de 4 GB.

En esta fase únicamente se utilizó la parte de software que involucra el procesamiento de imágenes para determinar la posición de la pupila y establecer si esta se encuentra en el centro o en la derecha, para interactuar con el menú.

Para lograr esto, se tomó algunas imágenes con una webcam Genius USB iSlim 300x a una resolución de 160x120 pixeles y en las cuales se varió algunos parámetros importantes como el lugar de captura y la intensidad de luz, la cual afecta considerablemente el procesamiento de imágenes en el software.

La tabla 2 muestra los parámetros para la adquisición de las imágenes y los resultados generados por el software.

Tabla 2. Detalles de los resultados del procesamiento de imágenes

Prueba	Características	Posición del ojo	Imagen resultante	Dato generado
1	Fondo oscuro y luz natural de una ventana sobre el rostro	Centro		Centro
2	Fondo oscuro y mayor luz natural de una ventana sobre el rostro	Centro		Centro
3	Fondo oscuro y mayor luz natural de una ventana sobre el rostro	Derecha		Derecha
4	Fondo claro y mayor luz natural de una ventana sobre el rostro	Centro		Centro
5	Bastante luz natural sobre el rostro	Derecha		Centro
6	Bastante luz natural sobre el rostro y apertura mayor del ojo	Derecha		Derecha
7	Exceso de luz sobre el ojo proveniente de una linterna	Derecha		Sin dato

Tabla 2. Detalles de los resultados del procesamiento de imágenes (continuación)

8	No hay imagen de un rostro	No aplica		Centro
9	Ausencia de luz sobre el rostro	Centro		Centro
10	En un sitio abierto con luz directa del sol	Derecha		Sin dato

Como se puede observar, en la tabla se muestran las imágenes del ojo con su respectivo punto característico, la pupila, donde el punto rojo correspondiente a la pupila se aproxima al punto verdadero, pero presenta variaciones que en la mayoría de los casos no afecta al dato final generado por el software.

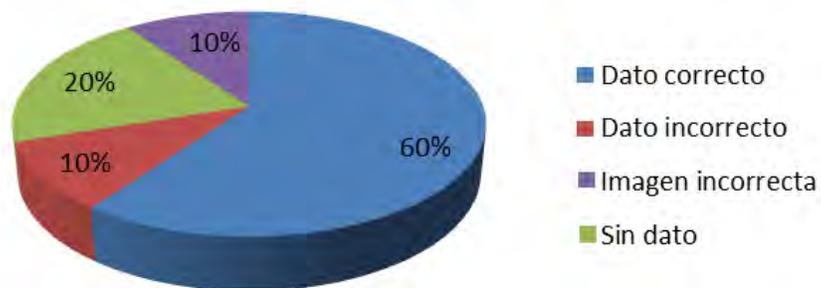
Se observa además que en la prueba número 8 que no corresponde a una imagen del ojo, se genera un dato de “Centro”, este error se presenta porque el algoritmo ubica ciertos puntos y los procesa y da el resultado que pertenece a ese dato final.

En las pruebas 7 y 10, no se generan datos, debido a la intensidad de la luz proveniente de una linterna y la luz directa del sol.

Si se experimenta en un sitio abierto, los resultados obtenidos son nulos, pues el brillo excesivo hace que en los primeros procesos en el análisis de imágenes, esta información se pierda, y los filtros eliminen por completo todos los datos de la imagen.

Los resultados obtenidos también se pueden observar en la figura 47.

Figura 47. Resultados porcentuales del algoritmo de localización de la pupila

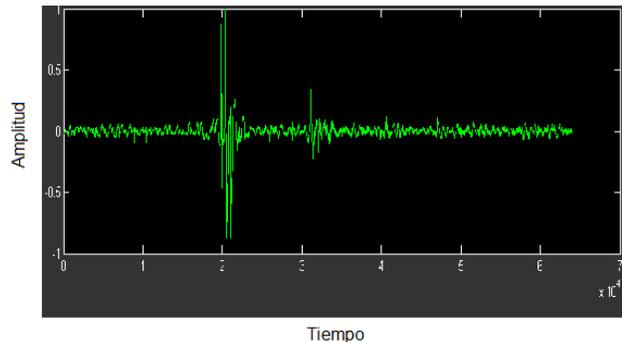


7.2 SEÑALES DE PARPADEO

La figura 48 muestra claramente el instante en que se produce el parpadeo que corresponde a señales de buena amplitud debido a que son señales producidas de manera voluntario dada por el usuario.

Esta imagen no se muestra en el programa principal, solo es para indicar el funcionamiento.

Figura 48. Imagen del parpadeo



Para activar o desactivar un dispositivo es necesario realizar algunos parpadeos, puesto que el software no se interrumpe al realizar dicha acción, solo espera en un determinado tiempo que esta señal se produzca.

7.3 SISTEMA USB

El prototipo obtenido es una placa de 12,7 x 3,8 cm en la que se incluye tanto la interfaz USB como la etapa de transmisión de radiofrecuencia y cuya imagen se muestra en el Anexo E.

El periférico se ha probado en varios equipos con distintos sistemas operativos y arquitecturas. El objetivo es comprobar que el periférico sea reconocido como un dispositivo USB y que todo el proceso de instalación y comunicación periférico-host sea correcto.

Los computadores donde se llevaron a cabo las pruebas son:

- HP con procesador AMD Turion X2 a 2.1 GHz, memoria RAM de 4 GB y sistema operativo Windows 7 Profesional 64 bits.
- DELL con procesador Intel Core2Duo a 2.1 GHz, memoria RAM de 3 GB y sistema operativo Windows 7 Premium 32 bits.

- PC Smart con procesador Intel Pentium IV a 2.6 GHz, memoria RAM de 256 MB y sistema operativo Windows XP Profesional 32 bits.

En todos ellos al conectar el periférico al puerto USB se ha detectado nuevo hardware y se ha mostrado el asistente para agregarlo. Se ha indicado la localización del fichero INF y la instalación del driver ha sido correcta, pudiendo ver en el Administrador de Dispositivos como aparecía el periférico. Se ha vuelto a repetir el proceso en todos ellos y, la segunda vez que se conectaba, lo reconocía de forma automática sin necesidad de indicar ningún dato.

La instalación se llevó a cabo siguiendo las indicaciones del manual de usuario del Anexo H.

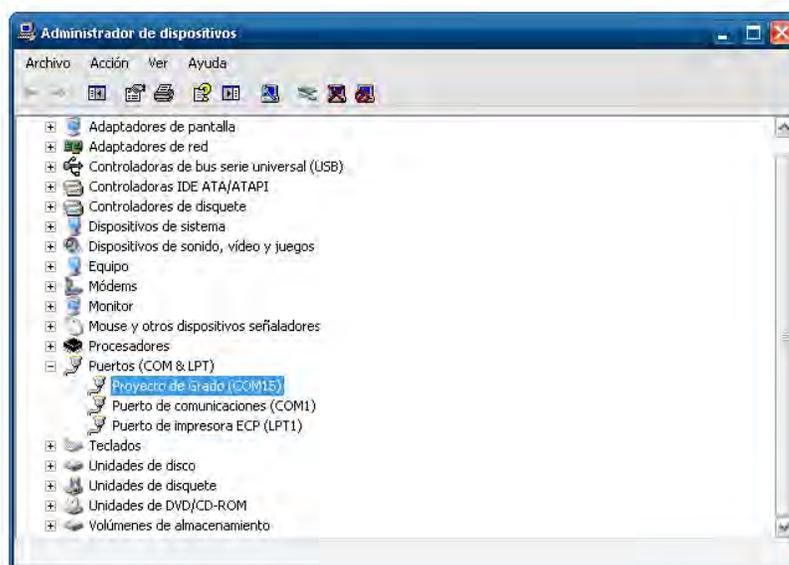
El mensaje de instalación exitosa en la barra de tareas (figura 49), solo se muestra en sistema operativo Windows XP.

Figura 49. Mensaje de instalación correcta del hardware.



La figura 50 muestra en el administrador de dispositivos la asignación aleatoria de un puerto COM virtual e indica que el sistema está conectado y listo para usarse.

Figura 50. Presentación del hardware USB en el administrador de dispositivos



7.4 MENÚ GRÁFICO

Al ejecutar la aplicación de Interfaz Hombre-Computador, la primera pantalla que aparece es la mostrada en la figura 51, la cual tiene un fondo negro, en el que se destacan en la parte central la imagen del primer dispositivo que corresponde al televisor, encima de esta, se localiza el nombre y estado actual del mismo, a la derecha se encuentra la imagen del siguiente elemento, el radio que es representado por un símbolo de música. La imagen de la lámpara solo aparece cuando se cambia el menú al siguiente elemento.

Los botones Iniciar y Detener cumplen la función de configurar, cargar y abrir todos los elementos que serán utilizados en la interfaz como la webcam y detener la ejecución del procesamiento de imágenes y del sistema de datos USB.

Figura 51. Pantalla inicial del menú



El cuadro gris en la parte inferior es el cuadro de referencia, donde aparece el ojo del usuario y con la ubicación exacta del mismo, se logra el correcto funcionamiento del procesamiento de imágenes, después de dar clic en el botón Iniciar. Algunas veces el software detecta imágenes que no corresponden al ojo y sin embargo genera datos correctos.

Cabe resaltar que si se presiona sobre el botones Iniciar sin tener conectado el dispositivo USB se genera un mensaje de error indicando la ausencia de este sistema.

Figura 52. Mensaje de hardware sin conexión



Adicionalmente se tiene un botón en la parte superior derecha denominado Acerca de, que muestra el título del menú, la universidad, los autores y director del proyecto (figura 53).

Figura 53. Pantalla de información del proyecto



Este software ha sido probado en las mismas maquinas en que se ensayó el sistema USB, pero existe un problema en cualquiera de los equipos que tenga una resolución diferente a 1280x800, ya que no muestra totalmente la interfaz. Este problema se resuelve si se configura a la resolución mencionada.

En general, la aplicación software funciona correctamente, pero se observó que para prender o apagar el sistema seleccionado en el menú, se debe realizar varios parpadeos para que el algoritmo lo reconozca.

7.5 CIRCUITOS DE RADIOFRECUENCIA

El prototipo obtenido es una placa de 10 x 5,7 cm que consta de la etapa de recepción de radiofrecuencia y la fase de potencia y cuya imagen se muestra en el Anexo G.

De antena, tanto en el receptor como en el transmisor, se utilizó un alambre de 17 cm. de longitud, pero también se observó que en el receptor no era necesario este elemento. Sin embargo el enlace funcionó muy bien, comunicándose incluso a través de paredes, y a una distancia de unos 10 metros aproximadamente.

La dirección de dispositivo se ingresa a través de los pines 1 a 8 en ambos chips. La hoja de datos indica que se pueden dejar estas patas al aire, pero con esta configuración no se logró establecer una comunicación, así que se conectó dichos pines a tierra. Entonces, el enlace descrito en este trabajo funcionó con una dirección igual a 0. Por consiguiente, con las pruebas realizadas se demostró que es mejor poner estos pines a un nivel y no dejarlas en el aire, sea a masa o sea a un valor de +5V, porque si esto no se hace el funcionamiento puede resultar irregular.

En el integrado decodificador HT12D, la señal VT significa Valid Transmission (Transmisión Válida), es decir, cada vez que esta señal va a un nivel alto es porque el código presente en la salida de datos es un dato válido para el dispositivo receptor.

Con respecto a la parte de dirección, si el dispositivo HT12D no tiene la misma dirección que viene en la palabra que ha recibido, no se produce esta señal VT.

Cuando se mantiene en bajo la entrada TE o Habilitación de la transmisión, el integrado codificador HT12E transmite constantemente, repitiendo el código una y otra vez. En estas condiciones, si se modifica el valor de las entradas de datos se observa que el valor va cambiando en el receptor. Así fue como se realizó la prueba.

Respecto a la alimentación, únicamente se probó el transmisor con 5V ya que esta es la que tiene el puerto USB, en cambio el circuito de recepción se probó con fuentes de 5V y 9V, sin alterar el óptimo funcionamiento en los dos casos, pero debido a la activación de los relés se dejó la de 9V puesto que manejan mayor potencia.

7.6 CIRCUITOS DE POTENCIA

La función de los relés utilizados en esta fase es cerrar o abrir contactos lo cual permite conectarlos en serie y así activar o desactivar los sistemas del televisor, el radio y la lámpara.

El correcto funcionamiento se logró en todos los sistemas, dentro de los cuales se encuentran los siguientes:

Un televisor SONY de 21" de 100 W.

Un televisor SONY LCD de 32" de 114 W.

Un televisor Phillips de 14" de 60 W.

Un televisor LG de 21" de 100 W.

Una grabadora SONY de 60 W.

Un radio SONY de 20 W.

Un bombillo Phillips de 100 W.

Un bombillo Phillips de 60 W.

Un bombillo ahorrador Phillips de 24W

Se observó que algunos equipos con sistema Stand By como el televisor LG de 21" no pueden ser accionados directamente, puesto que es necesario activarlos con control remoto. En tal caso la aplicación queda limitada a habilitar solo el sistema Stand By y no el encendido del mismo.

8. CONCLUSIONES

Respecto al objetivo general de este proyecto, se puede decir que se cumplió plenamente, puesto que se logró obtener un sistema en que el usuario interactúa con el computador para habilitar o deshabilitar dispositivos comunes, aunque el porcentaje de aciertos en el procesamiento de datos sea de un 60% dentro de diferentes entornos de iluminación.

Igualmente los objetivos específicos se cumplieron a cabalidad, el menú gráfico se visualizó en todos los computadores utilizados y debidamente configurados, el procesador de imágenes generó datos acordes a la posición de los ojos del usuario y la interfaz de salida de datos USB permitió activar correctamente los dispositivos utilizados.

En cuanto al software, se produce mejores resultados en el procesamiento de imágenes si las condiciones de iluminación son las apropiadas, es decir sin exceso de luz y sin ausencia de la misma.

Dentro de la aplicación software se debe tener en cuenta que en las imágenes adquiridas, el fondo en el rostro no altera el procesamiento óptimo, pero sí produce una mala detección el exceso de luz o la ausencia de esta. El sistema no funciona correctamente si la cámara no soporta la resolución 160x120 píxeles. Además, cabe resaltar que este software no es aplicable cuando se produce reflejos de luz provenientes de vidrios o gafas del usuario.

La parte que involucra la utilización del micrófono como sensor es más confiable que si se lo utilizara con el reconocimiento de imágenes, puesto que se evita los errores producidos por un parpadeo natural e involuntario.

La elección del bus USB y del microcontrolador PIC18F2550 ha influido tanto en el tamaño del periférico como en su costo. USB dispone de conectores muy sencillos, de tamaño reducido y es bastante económico.

La utilización de relés en la etapa de potencia, implica una forma económica de manejar aparatos que operan con voltajes altos, además de la sencillez y el óptimo funcionamiento que se logra en esta fase.

9. RECOMENDACIONES

Para correr el software, inicialmente otra persona debe ejecutar la aplicación, para instalarla, iniciarla y detenerla, luego el usuario es autónomo en la utilización del sistema. No es conveniente cerrar la aplicación sin antes detenerla con el respectivo botón, ya que se genera errores al intentar cerrar las funciones asociadas al puerto serie virtual.

No se puede controlar el encendido de todos los aparatos electrodomésticos, puesto que algunos disponen de sistemas Stand By que necesitan accionarse en el aparato o en el control remoto del mismo, dicho problema puede ser solucionado retirando la etapa de Stand By y conectando el sistema del TV directamente.

El desarrollo de este proyecto se puede ampliar con un mayor número de procesos, por ejemplo el menú, puede incluir más gráficos relacionados con un sinnúmero de electrodomésticos, y además se puede agregar submenús para interactuar de forma más amplia, como subir o bajar el volumen o incluso cambiar de canal, para el caso de un televisor.

También se podría implementar el sistema desarrollado en un hardware portátil, como un celular, que incluya una cámara necesaria para adquirir imágenes y procesarlas, y adicional a ello se podría utilizar el sistema Bluetooth que incorporar los modernos teléfonos móviles para activar sistemas sin necesidad de cables.

Otra de las aplicaciones en las que se podría profundizar, basándose en el algoritmo de procesamiento de imágenes, consistiría en el movimiento del mouse para activar botones, mediante el seguimiento de la pupila del ojo.

BIBLIOGRAFÍA

ACHARYA, T., Ray, A. K. Image processing: principles and applications. John Wiley & Sons. 2005.

ALEGRE, E., SÁNCHEZ, L., FERNÁNDEZ, R. Á. Procesamiento Digital de Imagen: fundamentos y prácticas con Matlab. Universidad de León. 2003

ARBOLEDA, Carolina, POSADA, Alejandro. Diseño y construcción de un prototipo de interfaz cerebro-computador para facilitar la comunicación de personas con discapacidad motora. Revista EIA. Escuela de Ingeniería de Antioquia, Medellín. Julio 2009. p. 105-115

AXELSON, Jan. USB complete, tercera edición. Lakeview Research. 2005. 572 p.
BORWICK, John, Micrófonos, Tecnologías y Aplicaciones. Escuela de Cine y Video, 1990.

BOYLESTAD, Robert. Electrónica: Teoría de Circuitos y dispositivos electrónicos. 8 Ed. México: Pearson Education, 2003.

CALVO, José Luis. Medida del movimiento del párpado mediante visión artificial. Tesis de grado. Madrid: Universidad pontificia Comillas. 2007. 195 p.

DAUGMAN, 'How Iris Recognition Works', IEEE Transactions on Circuits and Systems For Video Technology, Vol. 14, n 1, January 2004.

ENUMERACIÓN USB. Documentación HTML. Disponible en internet.
<http://www.unpocodelectronica.net.au.net/generador-de-inf-para-los-drivers-usb-de-microchip/>

FRANCO, John, GIRALDO, Eduardo. Reconocimiento de patrones de movimiento del sistema fisiológico brazo-antebrazo, a partir de señales electromiográficas. Tesis de grado. Pereira: Universidad tecnológica de Pereira, Facultad de ingeniería eléctrica. 2004. 54 p.

GARCIA BREIJO, Eduardo. Compilador C CCS y simulador PROTEUS para microcontroladores PIC. Barcelona, España. Ediciones Marcombo. 2008. 262 p.

GONZÁLEZ, R.C., Wintz, P. Procesamiento digital de imágenes. Addison-Wesley 1996.

HYDE, John. USB design by example. Intel University Press. 348 p.

IBRAHIM, Dogan. Advanced PIC Microcontroller Projects in C. Elsevier. 2008. 544 p.

IFEACHOR, E., Jervis, B. Digital Signal Processing: a practical approach, Addison-Wesley Longman Ltd, Edingburg Gate, Harlow, Inglaterra. 1993.

MUHAMMAD H. Rashid, Electrónica de Potencia, Circuitos, dispositivos y aplicaciones, Tercera Edición, Pearson Prentice Hall.

MÚSCULOS DE LA CARA. Documentación HTML. Disponible en Internet. http://es.wikipedia.org/wiki/M%C3%BAsculos_de_la_cara#M.C3.BAsculo_Orbicular_de_los_p.C3.A1rpados

MOYANO, Jonathan, PIC18F2550 y USB Desarrollo de aplicaciones. Documentación PDF, disponible en internet: <http://www.todopic.com.ar/>

NAVARRO, Daniel. Monitor EMG con conexión inalámbrica. Tesis de grado. Bellaterra: Universidad Autónoma de Barcelona. 2008. 72 p.

PROAKIS, John G y MANOLOKIS, Dimitris G., Tratamiento digital de señales: Principios, algoritmos y aplicaciones, 3ra edición, Pearson - Prentice Hall, 2005.

REALPE, Judy, ROMO, Harold, Análisis de Señales EMG Superficiales y su Aplicación en Control de Prótesis de Mano. Tesis de grado. Universidad del Cauca. 2007.

SANTOS Sierra, C. Sánchez Ávila, E. Marchiori, Iris Recognition: Segmentation enhancement by using Morphological Operators, Master Final Thesis, June, 2007. Amsterdam.

SHNEIDERMAN, Ben. Designing The user interface, Strategies for effective Human-computer interaction. Addison-wesley, 1998.

TARJETA DE SONIDO, Documentación HTML. Disponible en Internet http://es.wikipedia.org/wiki/tarjeta_de_sonido.htm

TORTORA, Derrickson. Principios de Anatomía y Fisiología, 11ª edición. 2006.

VALERIUS, Klaus. El libro de los músculos. ARS Médica. 2008. 419 págs.

ANEXOS

ANEXO A: Hoja de datos del PIC18F2550

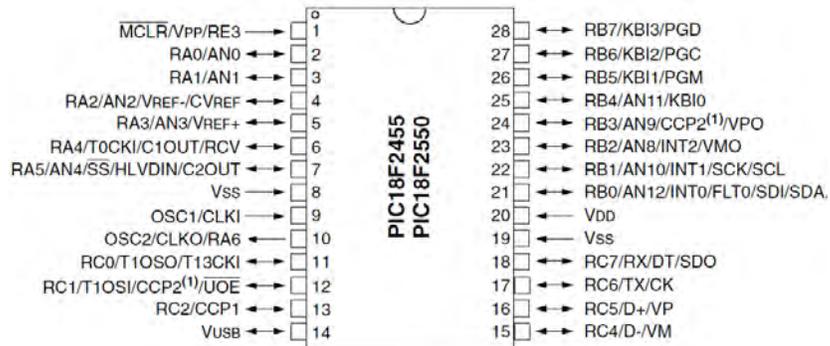
Interface USB 2.0 de alta velocidad 12Mbit/s

Typo de memoria: Flash, Memoria de programa: 32 Kb

CPU Speed (MIPS): 12

Memoria Ram: 2048 bytes, EEPROM: 256 bytes

Voltaje de operacion: 2 a 5.5 V



Absolute Maximum Ratings(†)

Ambient temperature under bias.....	-40°C to +85°C
Storage temperature	-65°C to +150°C
Voltage on any pin with respect to VSS (except VDD, MCLR and RA4)	-0.3V to (VDD + 0.3V)
Voltage on VDD with respect to VSS	-0.3V to +7.5V
Voltage on MCLR with respect to VSS (Note 2)	0V to +13.25V
Total power dissipation (Note 1)	1.0W
Maximum current out of VSS pin	300 mA
Maximum current into VDD pin	250 mA
Input clamp current, I _{IK} (V _I < 0 or V _I > VDD).....	±20 mA
Output clamp current, I _{OK} (V _O < 0 or V _O > VDD)	±20 mA
Maximum output current sunk by any I/O pin.....	25 mA
Maximum output current sourced by any I/O pin	25 mA
Maximum current sunk by all ports	200 mA
Maximum current sourced by all ports	200 mA

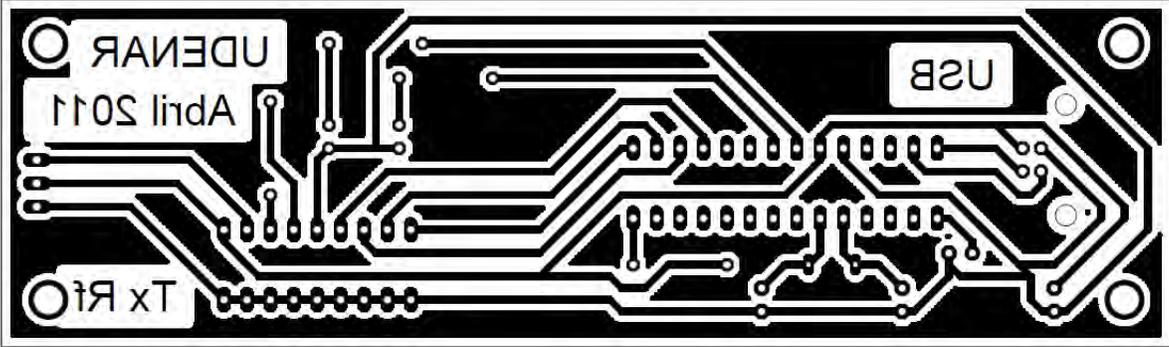
ANEXO B: Características eléctricas del circuito HT12E

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		V _{DD}	Conditions				
V _{DD}	Operating Voltage	—	—	2.4	5	12	V
I _{STB}	Standby Current	3V	Oscillator stops	—	0.1	1	μA
		12V		—	2	4	μA
I _{DD}	Operating Current	3V	No load, f _{OSC} =3kHz	—	40	80	μA
		12V		—	150	300	μA
I _{OUT}	Output Drive Current	5V	V _{OH} =0.9V _{DD} (Source)	-1	-1.6	—	mA
			V _{OL} =0.1V _{DD} (Sink)	1	1.6	—	mA
V _{IH}	"H" Input Voltage	—	—	0.8V _{DD}	—	V _{DD}	V
V _{IL}	"L" Input Voltage	—	—	0	—	0.2V _{DD}	V
f _{OSC}	Oscillator Frequency	5V	R _{OSC} =1.1MΩ	—	3	—	kHz
R _{TE}	$\overline{\text{TE}}$ Pull-high Resistance	5V	V _{TE} =0V	—	1.5	3	MΩ

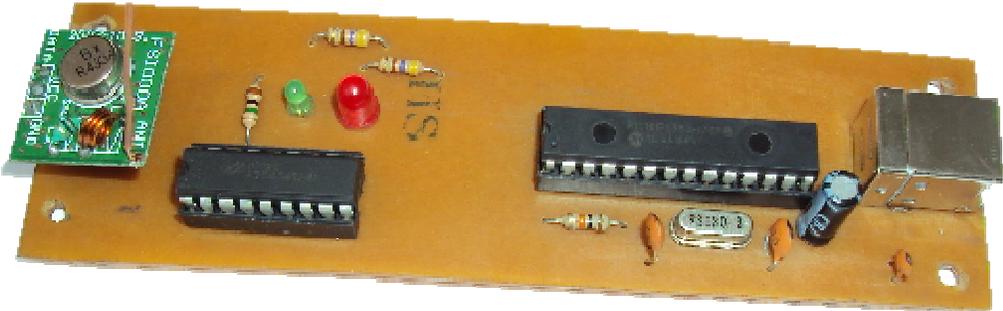
ANEXO C: Características eléctricas del circuito HT12D

Symbol	Parameter	Test Conditions		Min.	Typ.	Max.	Unit
		V _{DD}	Conditions				
V _{DD}	Operating Voltage	—	—	2.4	5	12	V
I _{STB}	Standby Current	5V	Oscillator stops	—	0.1	1	μA
		12V		—	2	4	μA
I _{DD}	Operating Current	5V	No load f _{OSC} =150kHz	—	200	400	μA
I _O	Data Output Source Current (D8~D11)	5V	V _{OH} =4.5V	-1	-1.6	—	mA
	Data Output Sink Current (D8~D11)	5V	V _{OL} =0.5V	1	1.6	—	mA
I _{VT}	VT Output Source Current	5V	V _{OH} =4.5V	-1	-1.6	—	mA
	VT Output Sink Current		V _{OL} =0.5V	1	1.6	—	mA
V _{IH}	"H" Input Voltage	5V	—	3.5	—	5	V
V _{IL}	"L" Input Voltage	5V	—	0	—	1	V
f _{OSC}	Oscillator Frequency	5V	R _{OSC} =51kΩ	—	150	—	kHz

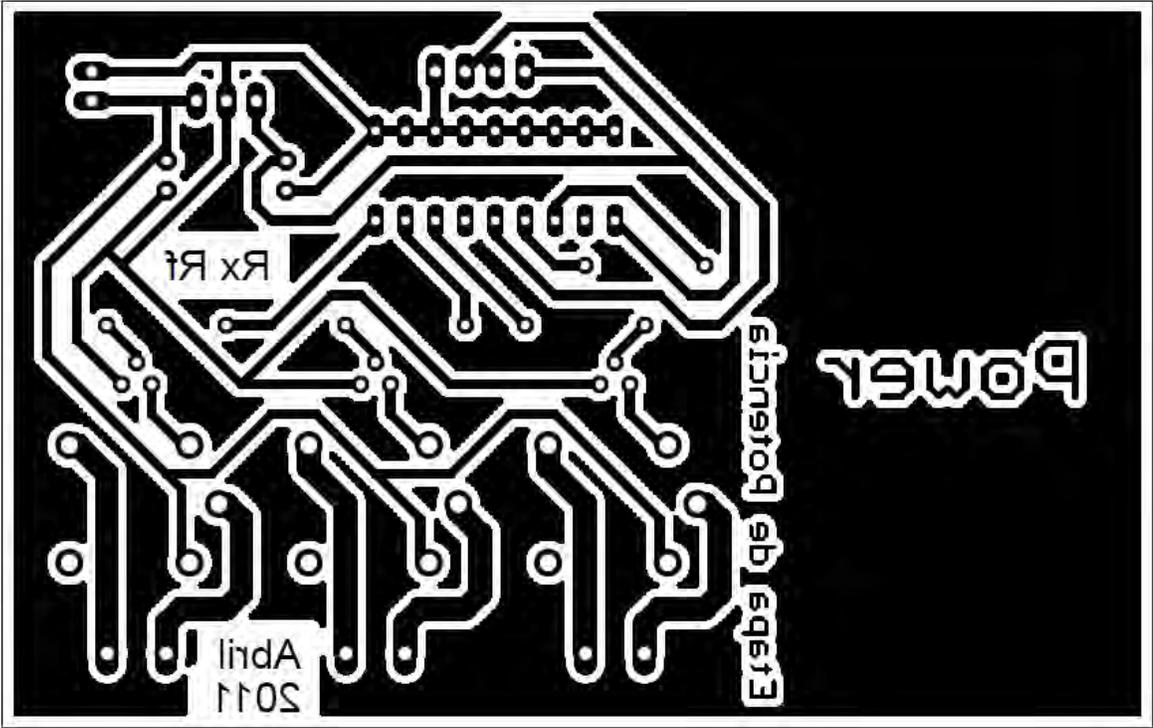
ANEXO D: Imagen del circuito impreso de la interfaz USB y la etapa de transmisión.



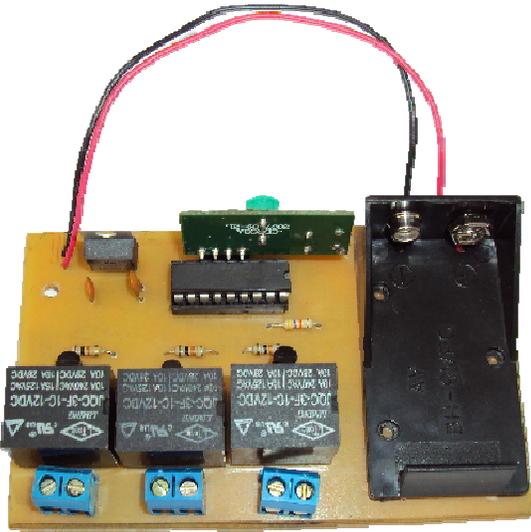
ANEXO E: Placa montada del USB y transmisor de RF



ANEXO F: Imagen del circuito impreso de la etapa de recepción y etapa de potencia.



ANEXO G: Placa montada del receptor de RF y potencia



ANEXO H. Manual de usuario.

Lea este manual antes de usar el producto.

INTRODUCCIÓN

Este manual ha sido realizado para garantizar el óptimo funcionamiento del sistema que permite la interacción entre un usuario con discapacidad motriz o del habla y elementos electrónicos externos.

Aquí se describe las instrucciones necesarias para la instalación y configuración de todos los elementos de sistema.

Antes de utilizar los dispositivos lea cuidadosamente este manual en su totalidad y manténgalo siempre a mano para posibles consultas.

ADVERTENCIA



PRECAUCION

Riesgo de choque eléctrico

No abrir los dispositivos



Coloque los sistemas USB, transmisión, recepción y potencia en un lugar estable y lejos de la humedad.



Para evitar descargas eléctricas, no abrir ninguno de los dispositivos ni manipular los circuitos electrónicos internos.

Use los equipos de potencia únicamente con CA de 110 V.



Cuando la batería contenida en la etapa de potencia se gaste, reemplácela por una de propósito general de 9V. Por ningún motivo utilice fuentes externas o pilas de menor o mayor voltaje.

REQUERIMIENTOS MINIMOS

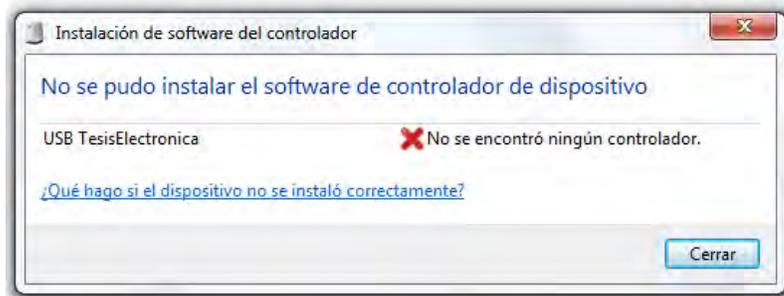
- Sistema operativo: Microsoft Windows XP, Windows 7 a 32 o 64 bits
- Dispositivos de entrada: Webcam que maneje resolución 640x480
Tarjeta de sonido
- Monitor que admita resolución de 1280x800
- Memoria RAM: 1 GB
- Espacio en disco: 500 MB

ACCESORIOS

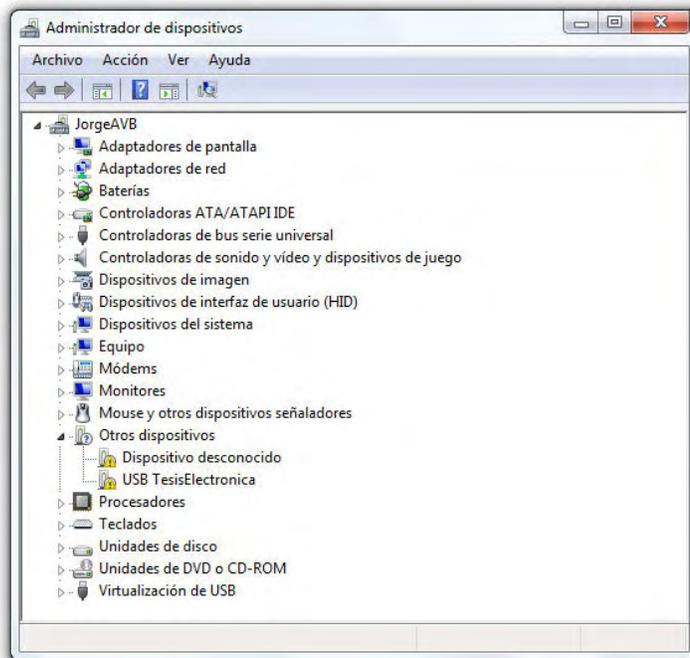
- Archivo instalador IHCsetup.exe
- Dispositivo USB y transmisor de radiofrecuencia TxRf+USB
- Dispositivo receptor de radiofrecuencia y etapa de potencia RxRf+Potencia
- Cable USB tipo A-B
- Sistema de adquisición de parpadeo (micrófono y plug).
- Pila de 9V
- Driver para conectar el sistema USB, mchpcdc.inf

INSTALACIÓN SISTEMA USB EN WINDOWS 7 Y VISTA (32 Y 64 BITS)

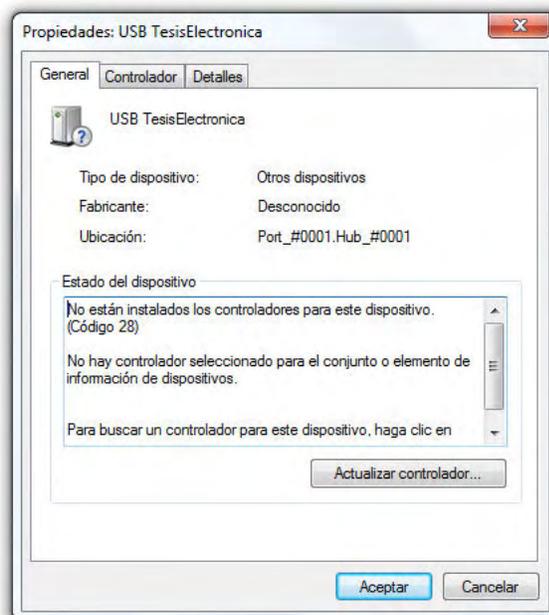
1. Conectar el cable al dispositivo USB (un LED rojo indicara la conexión correcta) y luego al computador, entonces Windows buscará los drivers, pero como no tiene ningún controlador asociado, muestra el siguiente mensaje:

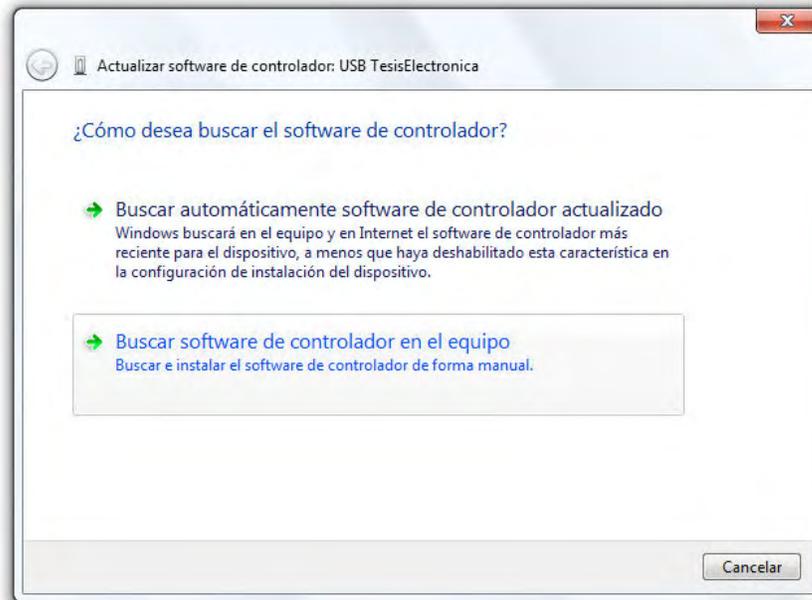


2. Hacer clic en la tecla Windows y escribir Administrador de dispositivos, se selecciona dicho programa y allí se muestra el elemento denominado "USB TesisElectronica".

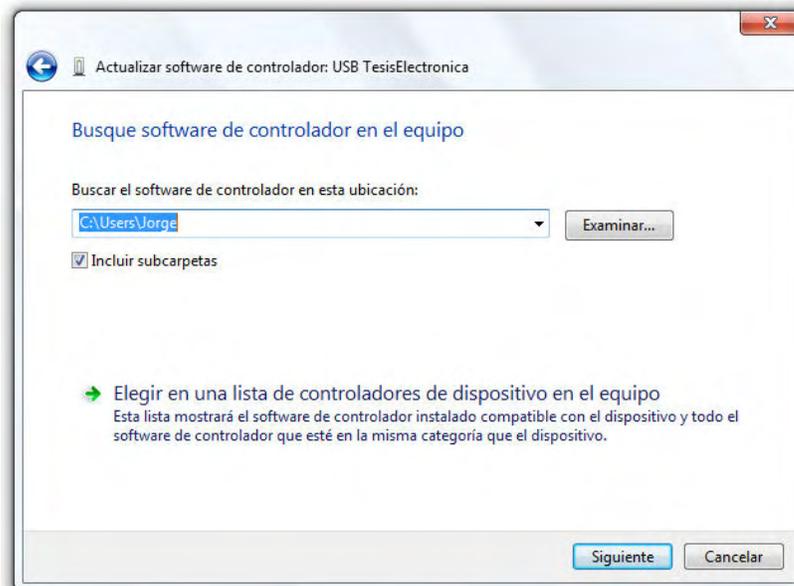


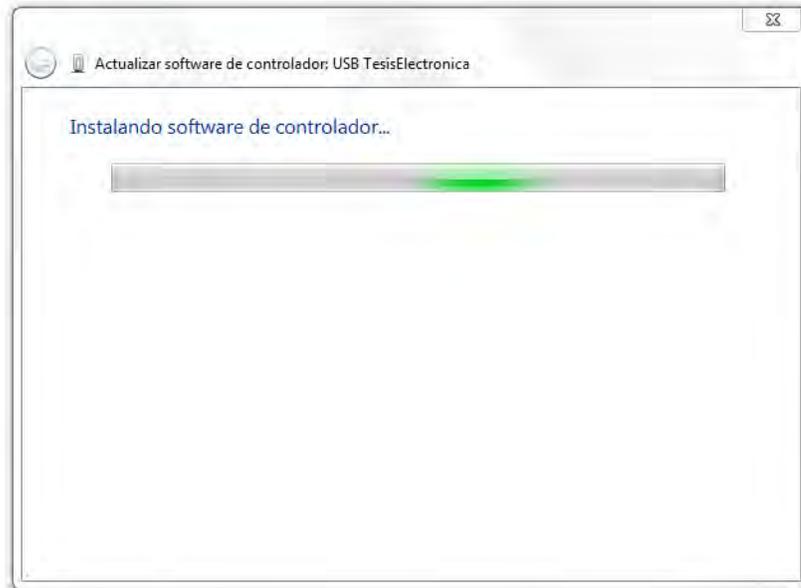
3. Hacer doble clic en el elemento USB TesisElectronica y clic en el botón Actualizar controlador y luego en Buscar Software de controlador en el equipo.





4. Buscar con el botón examinar la carpeta donde está el driver mchpcdc.inf suministrado, y luego hacer clic en siguiente y comenzara la instalación.

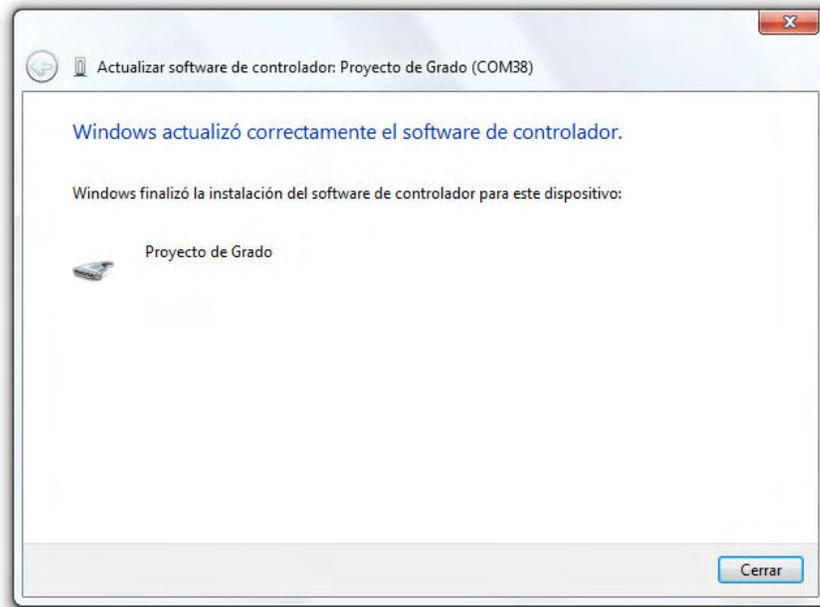




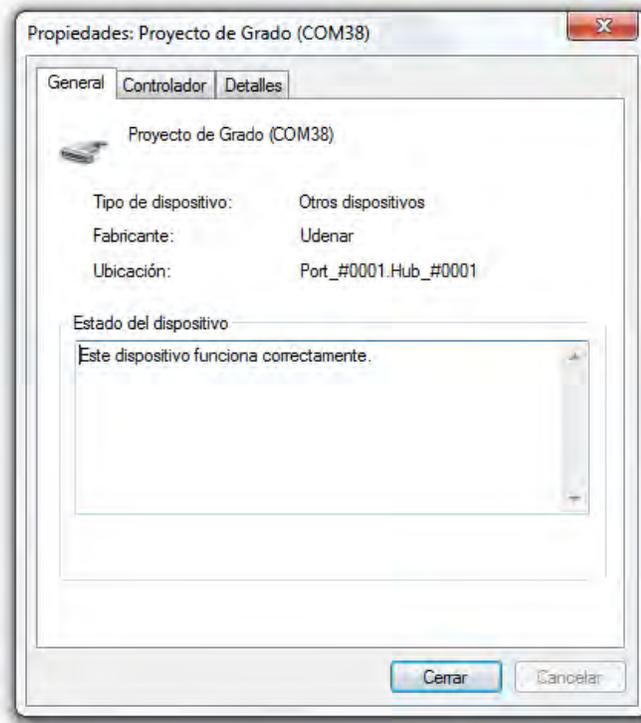
5. Se muestra un mensaje de seguridad de Windows, hacer clic en Instalar este software de controlador de todas formas.



6. Se muestra un mensaje de actualización correcta, clic en cerrar para terminar. En la parte superior de la pantalla aparece el número del puerto COM asignado aleatoriamente. Para configurarlo como puerto COM38 (predeterminado para este proyecto) consulte el tema **Configuración Puerto COM**.

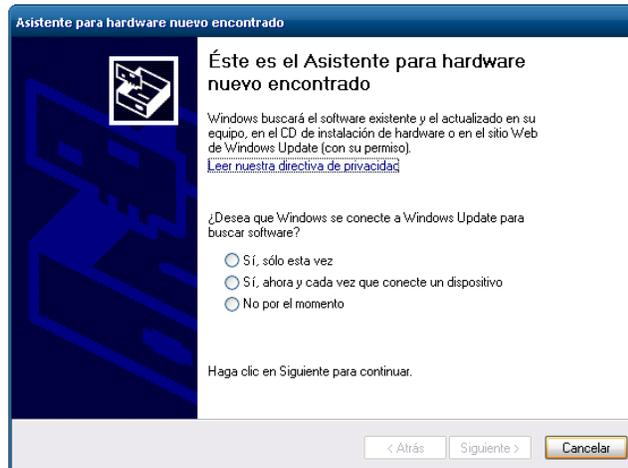


7. Abra nuevamente las propiedades del elemento y comprobará que el dispositivo ahora funciona correctamente.



INSTALACIÓN SISTEMA USB EN WINDOWS XP (32 BITS)

1. Conectar el cable al dispositivo USB (un LED rojo indicara la conexión correcta) y luego al computador, entonces aparecerá el asistente de instalación de hardware.

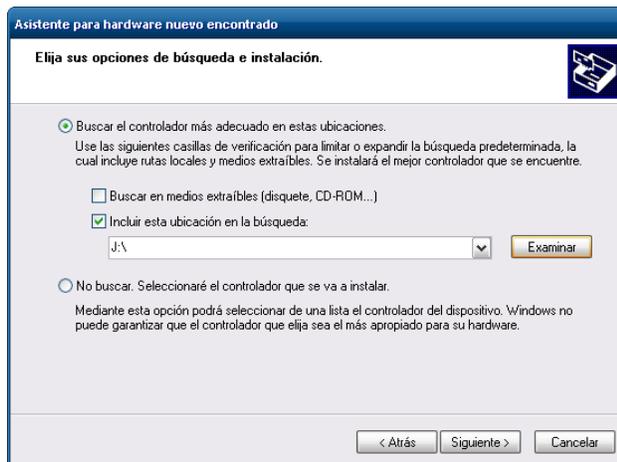


2. Como no es necesario buscar el controlador en internet, pulse en “No por el momento” y luego en siguiente.

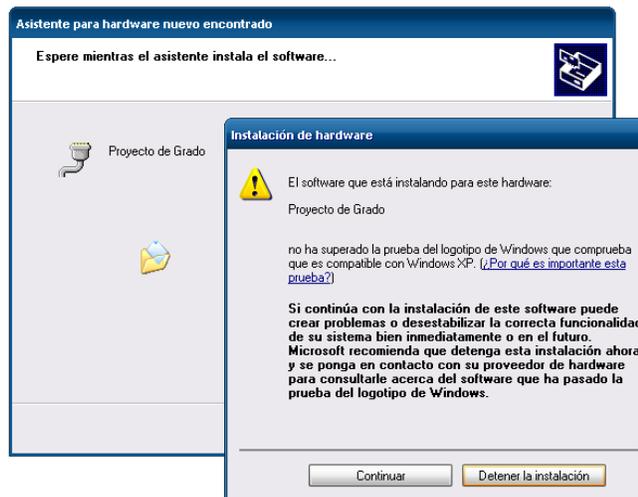
3. En la nueva ventana, pulse en “Instalar desde una lista o ubicación específica” y en siguiente.



4. Indique la ruta donde este el driver y de clic en siguiente.

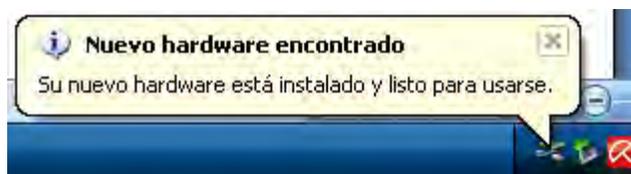


5. De clic en Continuar en la advertencia de Windows.



6. El asistente terminará la instalación.

7. El sistema está listo para usarse y se indicara en la barra de tareas.



CONFIGURACIÓN PUERTO COM

1. Conecte el dispositivo en caso de que este desconectado.
2. Presione la tecla Windows y escriba Administrador de dispositivos, localice el elemento USB TesisElectronica en la parte de puertos COM, abra las propiedades haciendo doble clic.
3. En la pestaña Configuración del puerto, presione sobre Opciones avanzadas, y en Número de puerto COM, cambie al COM38 y presione en Aceptar.
4. Para que los cambios tengan resultados, desconecte y conecte nuevamente el circuito USB.



INSTALACIÓN SETUP

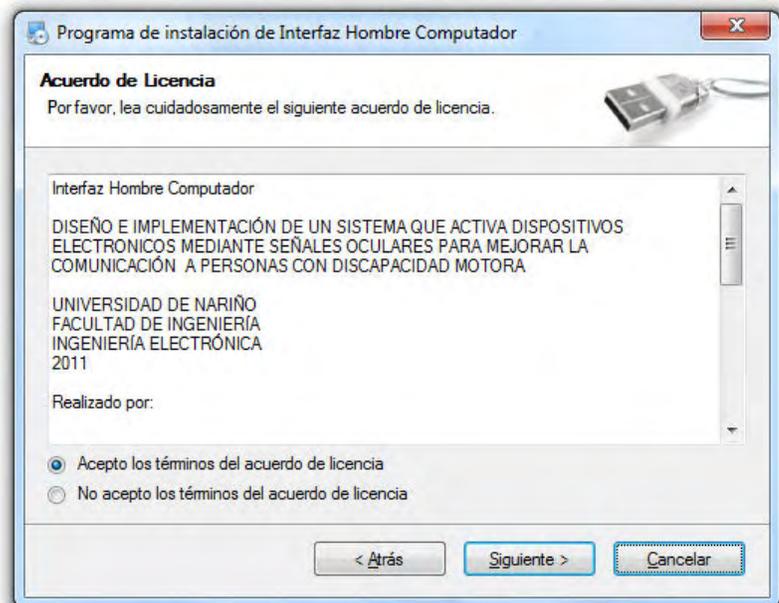
1. Ejecute el archivo IHCsetup.exe haciendo doble clic, (en Windows 7 acepte los privilegios de administrador).



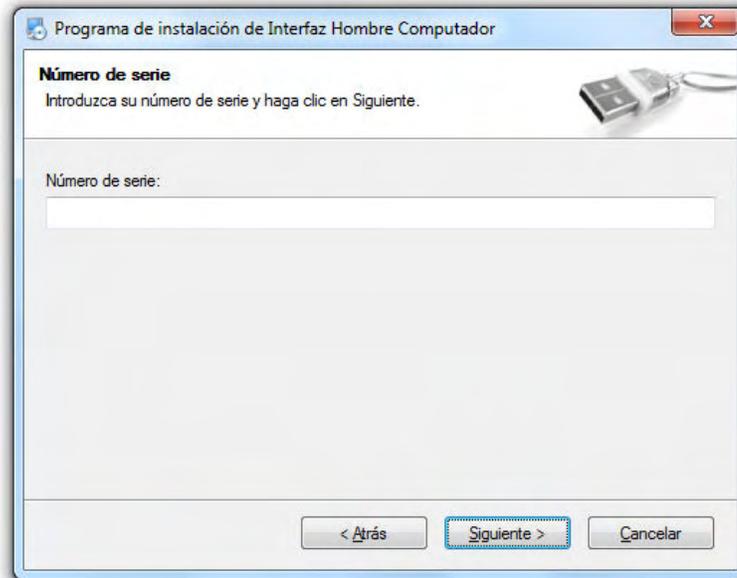
2. La primera pantalla es de Bienvenida, que además indica que debe cerrar todos los programas en ejecución, se pulsa sobre siguiente.



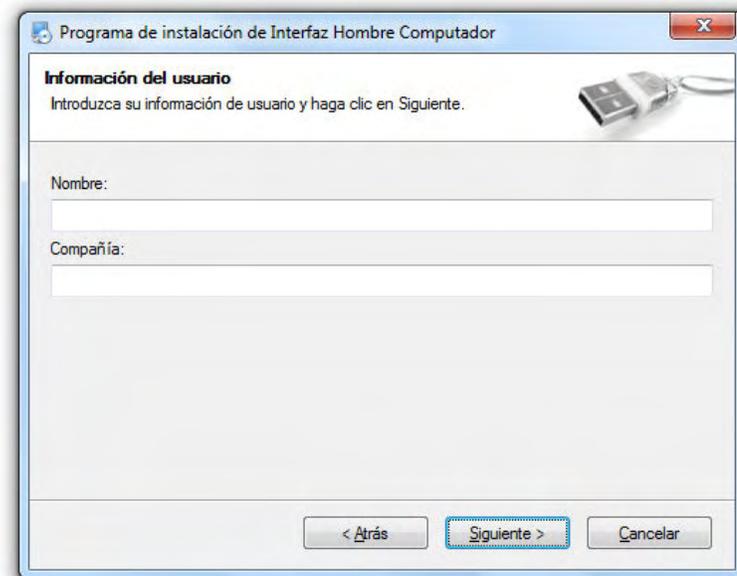
3. La siguiente pantalla muestra la licencia, se debe indicar la opción Acepto los términos de acuerdo de licencia y pulsar sobre siguiente.



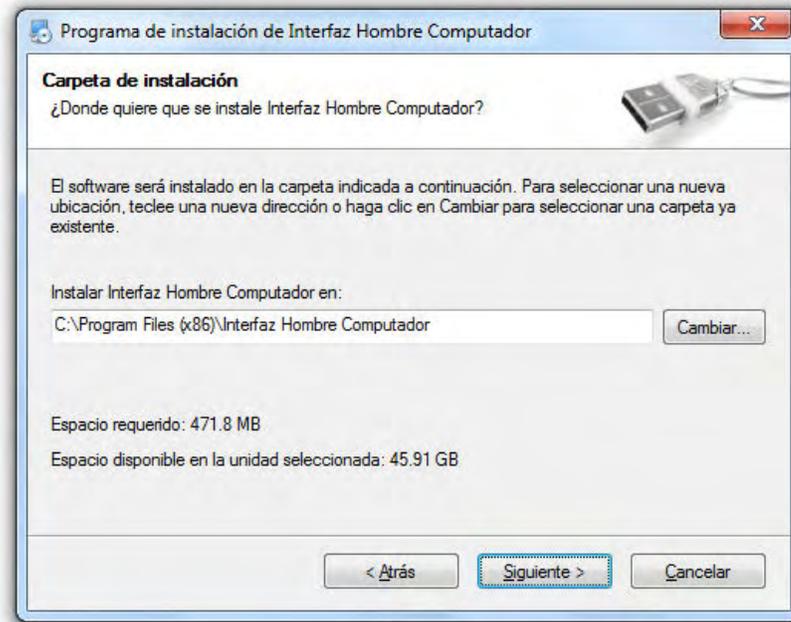
4. Otra pantalla pide se ingrese un número de serie, para poder instalar la aplicación, si no tiene un número, solicítelo al correo electrónico de los autores del proyecto, incluido en la licencia. Si el serial es correcto, al presionar sobre siguiente le indicará la nueva pantalla, de lo contrario tendrá que cancelar la instalación.



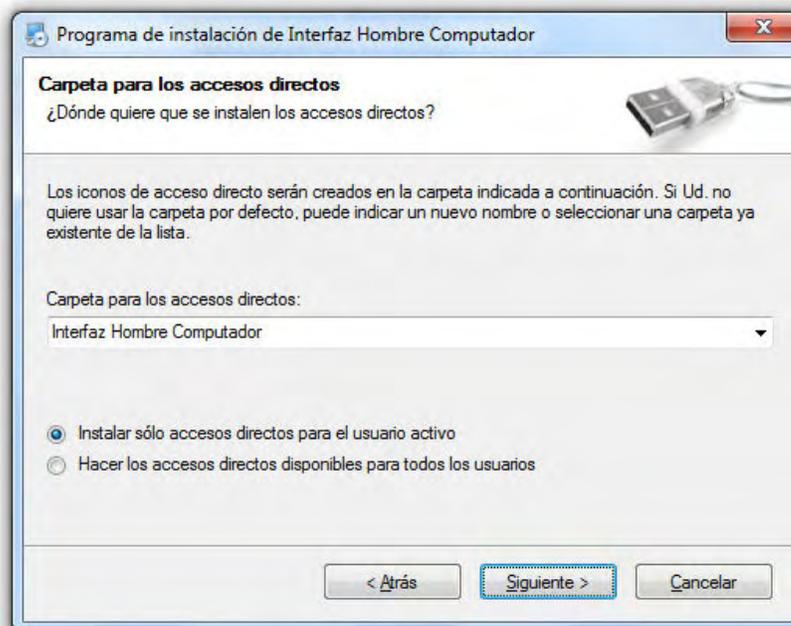
5. Ahora debe indicar la información de usuario y presionar sobre siguiente.



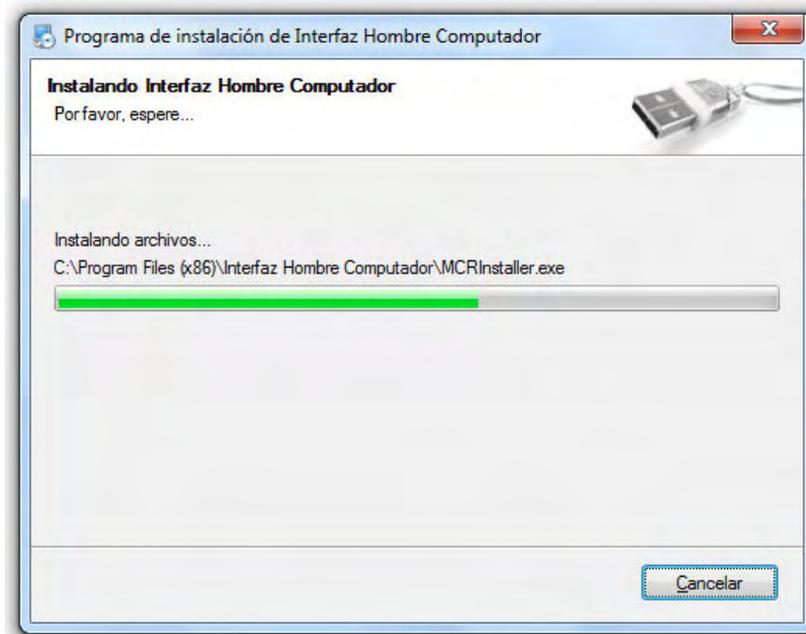
6. Indique la ruta de instalación, que por defecto es C:\Program Files\Interfaz Hombre Computador y pulse siguiente.



7. La siguiente pantalla permite indicar donde se crearán los accesos directos de la aplicación, seleccione la primera opción y pulse en siguiente.



8. Luego una pantalla le indica que está listo para instalar, pulse en siguiente y espere unos minutos.



9. El asistente le indica que necesita instalar un Runtime llamado MCR, si tiene instalada alguna versión de Matlab, presione en Cancelar, de lo contrario continúe con Aceptar.

10. Si Aceptó instalar el Runtime en la siguiente pantalla seleccione el idioma de instalación (Inglés) y proceda a instalar normalmente.



11. Finalmente la aplicación Interfaz Hombre Computador queda instalada, y se crea un acceso directo en el escritorio para mayor comodidad.

CONFIGURACIÓN DEL COMPUTADOR

Para que la aplicación se observe completamente se debe verificar y ajustar la resolución del monitor del computador.

1. En Windows 7, de clic derecho en el escritorio y seleccione Resolución de pantalla, en la opción Resolución verifique que sea 1280x800 y pulse en Aplicar y luego en Aceptar.
2. En Windows XP, de clic derecho en el escritorio y seleccione Propiedades, en la pestaña Configuración, cambie la resolución de pantalla a 1280x800 y pulse en Aplicar y luego en Aceptar.

Es posible que algunos computadores no manejen esta resolución, intente con alguna aproximada.

INSTALACIÓN DEL SISTEMA DE POTENCIA



1. Conecte el adaptador de 9V al sistema.
2. Conecte el cable de alimentación del dispositivo a un tomacorriente de 110 V.
3. Conecte los aparatos que serán activados en cada uno de los tomacorrientes marcados con su respectivo nombre.
4. Encienda el interruptor del aparato.

Es posible que algunos electrodomésticos no funcionen, solo se activaría su sistema Stand By.

CONEXIÓN DEL DISPOSITIVO USB Y RADIOFRECUENCIA

1. Conecte el cable USB tipo AB, un extremo al dispositivo y el otro al computador (se escuchara un sonido de conexión en el PC). No es necesario instalar el driver, puesto que lo reconoce automáticamente.
2. El circuito USB indica la conexión con el PC mediante la activación de un LED rojo. El LED verde solo se encenderá cuando la aplicación se ejecute y se llame al puerto y el LED rojo se apagará.



CONEXIÓN SISTEMA DE RECONOCIMIENTO DE PARPADEO

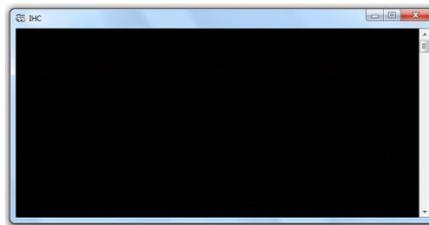
1. Desprende la cubierta protectora del micrófono y adhiéralo al ojo con el cual desea hacer el parpadeo para activar los aparatos.
2. Conecte el plug que une al sensor.

EJECUCIÓN DE LA APLICACIÓN

1. Ubique el computador en un espacio sin exceso de luz y sin ausencia de ella.
2. Abra la aplicación denominada IHC.exe que está en el escritorio.



3. Una pantalla ms-dos aparecerá durante unos segundos.



4. Luego aparecerá la pantalla principal pero esta permanece detenida.



5. Ahora se debe colocar la adaptación de la cámara a la persona con la discapacidad para operar los elementos del menú.

El usuario debe escoger una posición cómoda.

6. Otra persona puede correr la aplicación haciendo clic en Iniciar.

7. El usuario debe ajustar la posición de la adaptación de la cámara, de tal manera que en el recuadro de referencia aparezca claramente cualquiera de sus ojos.
8. Se escuchara un beep que indica que el procesamiento de imágenes se está ejecutando correctamente y se cambia de elemento en el menú.
9. Mire hacia la derecha para seleccionar el siguiente elemento, la imagen central cambiara del televisor al radio.
10. Si el menú no se desplaza o lo hace de forma automática, realice cambios de lugar junto al computador sin detener la aplicación, hasta observar que funcione correctamente.
10. Parpadee al escuchar el beep para activar o desactivar un elemento del menú, notara que el aparato se prendera o apagara y además escuchara un clic en el sistema de rf y potencia que indica la activación de unos de los relés.
11. Realice varias pruebas para que se familiarice con la aplicación de hardware y software.
12. Cuando termine presione Detener. Nunca cierre la aplicación directamente.

PROBLEMAS Y SOLUCIONES

Problema	Solución
Al correr la aplicación software aparece un mensaje error que indica la ausencia del componente mclmcrtr78.dll	Este problema se presenta por que el equipo no tiene instalada ninguna versión de Matlab y no se instaló el Runtime. Instale nuevamente la aplicación, y al finalizar instale el componente Runtime.
La pantalla de la aplicación no se muestra completamente.	Ajuste la resolución en las propiedades de la pantalla del equipo.
Al iniciar la aplicación IHC se presenta el error Hardware sin conexión.	Conecte el dispositivo USB e inicie la aplicación nuevamente. Si el error continúa, cambie el número de puerto a COM38 en las propiedades del dispositivo una vez conectado, luego desconéctelo y conéctelo otra vez.

<p>La conexión USB no funciona.</p>	<p>Instale el controlador inf suministrado.</p> <p>Si ningún de los dos LEDs están prendidos, verifique que el cable USB no este roto.</p>
<p>No se logra activar o desactivar los aparatos eléctricos.</p>	<p>Es probable que estén ubicados fuera de cobertura de los equipos de radiofrecuencia, acerque todos los dispositivos al circuito USB.</p> <p>Si el problema continua, escuche si los relés en la etapa de potencia suenan levemente cuando se acciona en el programa, si es así, verifique el estado del adaptador de 9V.</p>