



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**DISEÑO DE FRAME ADAPTABLE A PANTALLAS
PARA GENERACIÓN DE SUPERFICIES MULTITOUCH**

TESIS

**QUE PARA OBTENER EL TÍTULO DE
INGENIERO ELÉCTRICO ELECTRÓNICO**

PRESENTAN

**MARTÍNEZ MARTÍNEZ DAFNE GUADALUPE
QUINTANILLA ORDOÑEZ ISRAEL**

DIRECTOR DE TESIS

M. EN P. JESÚS RAMÍREZ ORTEGA



MÉXICO, D.F. CIUDAD UNIVERSITARIA 2015

JURADO ASIGNADO

PRESIDENTE:	Ing. Francisco J. Rodríguez Ramírez
VOCAL:	Ing. Jesús Ramírez Ortega
SECRETARIO:	Dr. Pablo R. Pérez Alcázar
1er. SUPLENTE:	Ing. Jesús J. Cortes Rosas
2do. SUPLENTE:	Dra. Fátima Moumtadi

SITIO DONDE SE DESARROLLÓ EL TEMA:

Centro de Ciencias Aplicadas y Desarrollo Tecnológico CCADET.
Universidad Nacional Autónoma de México.

ASESOR DEL TEMA:

Ing. Jesús Ramírez Ortega

SUSTENTANTES:

Martínez Martínez Dafne G.

Quintanilla Ordoñez Israel

Agradecimientos

A mis padres: **Victoria** y **Silvestre**, por haberme dado la vida y el privilegio de ser su hija. Les estaré infinitamente agradecida por todo el apoyo, los consejos, el impulso, el amor, los principios y los valores que me han dado. Sé que en ocasiones les eh sacado canas moradas, pero es parte de la vida. Gracias a ustedes eh culminado un ciclo más en mi vida y por fin logramos terminar la tesis. Los amo

A mi mamá **Vicky** por todo el apoyo, enseñanzas, alegrías, desvelos, regaños pero sobre todo por su inmenso cariño y amor. Nunca olvides que te admiro muchísimo por ser una mujer fuerte que ha sabido enfrentar todos los retos que te ha puesto la vida, sé que algunos son muy difíciles, pero no dudes de que yo siempre estaré a tu lado para apoyarte. Sabes que te quiero y te amo mucho. Gracias por ser mi mamá.

“El tiempo es un juez tan sabio, que no sentencia de inmediato, pero al final da la razón a quien la tiene”

A mi papá **Silvestre** quiero agradecerle todos esos esfuerzos por levantarse temprano y trabajar doble turno para salir adelante. Gracias por demostrarme que tengo muchas cosas bellas en la vida. Papá, por fin logramos obtener un título, ahora vienen muchas cosas buenas para nosotros y espero te quedes a mi lado para compartir contigo cada logro que vaya teniendo. Te quiero mucho.

“Piensa en todo aquello que tienes y no en lo que te falta” porque “Lo que hoy no se valora, en un futuro se lamenta”

A mi hermana **Itzel**, sabes que te quiero mucho y que te deseo lo mejor en tu futuro aprendizaje universitario. No olvides que para obtener cosas buenas hay que esforzarse el 200%, así que nunca te rindas de tus objetivos. Siempre habrá obstáculos en la vida, pero así es esto y hay que superarlos. Por cierto, ya no seas tan enojona que aun somos vecinas :)

“No hay ascensor para el éxito. Tienes que tomar las escaleras”

Al jovencito **Isra** Quintanilla, con el que comencé siendo compañera de generación, después nos volvimos amigos inseparables y ahora compañeros de tesis. Quiero agradecerle por su apoyo incondicional en todo momento, siempre ha sabido darme un abrazo de aliento cuando lo eh necesitado. Gracias por ser mi amigo, cómplice, confidente y novio al mismo tiempo. Pero principalmente gracias por invitarme a formar parte de este proyecto y haberme tenido paciencia. Juntos iniciamos este camino y juntos lo terminamos. Gracias por todo. Te amo.

“Llegar juntos es un comienzo; mantenerse juntos es progreso, trabajar juntos es éxito”

Al M en P **Jesús** Ramírez Ortega por habernos brindado la confianza y la paciencia de que este proyecto saldría adelante y por haberse tomado el tiempo de leer este trabajo. Gracias por todo el apoyo que nos dio en este proceso.

Al Ing. **Francisco** Rodríguez por confiar en nosotros y apoyarnos en todo esto desde el momento en que le pedimos ser nuestro sinodal.

A los sinodales: el Ing. **Pablo** Alcázar, la Dra. **Fátima** M y al Ing. **José** Cortes, por darse un tiempo para leer esta tesis y poder mejorarla con cada una de sus correcciones.

A mis amig@s: **Yunueth, Arely, Georgina, Diana, Roberto Nicolás y Francisco** que a pesar de no vernos seguido los estimo y los quiero mucho. Yunueth, sin duda mi amiga eterna, gracias por tu amistad por casi toda mi existencia. Arely, gracias por las pláticas intensas y buenísimas que hemos tenido. Georgina, gracias por ser mi amiga en esta facultad llena de hombres (aunque me duro poco el gusto☺) y por escucharme cuando más lo necesitaba. Diana, por haber sido una gran amiga y compañera de diversión en la prepa, fueron días tan divertidos e inolvidables. Nicolás, gracias por ser siempre un amigo incondicional y estar disponible para mí. Y Francisco, gracias por todas esas risas que vivimos juntos en la prepa, eres un amigo muy divertido.

Gracias a mis compañeros **Fabián, Benji, Yolo, Javier, Machorro, Maricruz, Ale**, etc. Por todos los momentos de locura que pasamos juntos. Sin su compañía la universidad no hubiera sido tan divertida e intensa.

Gracias a la **UNAM** y a la **Facultad de Ingeniería** por darme la oportunidad de haber sido alumna de profesores excelentes desde la preparatoria hasta la universidad. Gracias a todos aquellos maestros que alguna vez me dieron clase y compartieron mucho del conocimiento que tienen. Mil gracias a la Máxima casa de estudios por haberme dado un lugar en esta institución tan bella y por ser parte de mi vida, ahora ya soy de sangre azul y piel dorada.

Pero sobre todo gracias a **Dios** por bendecirme a diario con todas las cosas buenas que me rodean. Nunca te olvides de mí y de demostrarme que las cosas siempre pasan por algún motivo.

Dafne Martínez

Agradecimientos

Son muchas las personas especiales a las que quisiera agradecer su amistad, apoyo, ánimo y compañía en las diferentes etapas de mi vida.

Sin importar en dónde estén o si alguna vez llegan a leer estas dedicatorias quiero darles las gracias por formar parte de mí, por todo lo que me han brindado y por todas sus bendiciones.

Mamá, eres la mejor mamá del mundo, gracias por todo tu esfuerzo, tu apoyo y por la confianza que depositaste en mí. Gracias porque siempre has estado a mi lado. Te quiero mucho.

Papá, éste es un logro que quiero compartir contigo, gracias por ser mi papá y por creer. Quiero que sepas que ocupas un lugar especial en mi vida.

A mi hermana, eres la gran hermana que alguien pudiese tener, gracias.

A mis amigos, José Luis, Emanuel, Nicolás, Georgina que aunque son pocos, sé que puedo confiar en cada uno de ellos y no los cambiaría por nada.

A todos mis profes no sólo de la carrera sino de toda la vida, mil gracias porque de alguna manera forman parte de lo que ahora soy.

A Jesús y al CCADET por haber apoyado este proyecto y darme las herramientas para poder trabajar y seguir adelante.

Alethia, Mariana, Carmen, Elías, Jonathan donde el lugar de trabajo se hizo muy ameno y divertido, donde los días pasaron muy rápido gracias a su compañía.

A la persona que se ha vuelto parte fundamental en mi vida, a la que cada esfuerzo con ella vale la pena, a la que me hace reír y me hace enojarme a la vez, a la que me saca canas verdes pero me llena de felicidad, a esa persona que me cambió la vida, a mi novia y compañera de tesis Dafne. Muchas gracias. Te amo.

*A la UNAM la máxima casa de estudios y a la
Facultad de Ingeniería la mejor facultad de todas.*

Israel

ÍNDICE

Índice de figuras	IX
Resumen	XII
Introducción.....	XIV
Objetivos.....	XV
Capítulo 1. Conceptos básicos	
1.1. Superficies multitáctiles.....	1
1.2. Definición de marcos multitáctiles.....	2
1.3. Funcionamiento de los marcos actuales	
1.3.1. Estructura física.....	3
1.3.2. Funcionamiento.....	3
1.4. Luz infrarroja	
1.4.1. Led infrarrojo.....	17
1.4.2. Diodo láser.....	18
1.4.3. Fotodiodo.....	19
1.4.4. Fototransistor.....	20
1.5. Multiplexores y decodificadores (demultiplexores).....	21
1.6. Dispositivos de control.....	23
1.7. Arduino.....	28

Capítulo 2. Diseño de Marcos Multitáctiles

2.1.	Propuesta.....	35
2.2.	Mejoras del marco.....	35
2.3.	Ventajas.....	36
2.4.	Deficiencias.....	36
2.5.	Estructura física.....	37
2.6.	Materiales.....	41
2.7.	Circuito eléctrico.....	42
2.8.	Programación.....	47

Capítulo 3. Comparación con otros modelos del mercado

3.1.	Fabricantes de marcos multitáctiles.....	61
3.2.	Funcionamiento.....	64
3.3.	Área de proyección y puntos de toque.....	66
3.4.	Aplicaciones.....	67
3.5.	Precios.....	68

Capítulo 4. Resultados esperados.....77

Capítulo 5. Conclusiones.....79

Glosario.....81

Referencias.....83

Anexos.....87

ÍNDICE DE FIGURAS

Figura 1.1 Modelo multitáctil por infrarrojos.....	4
Figura 1.2 Modelo multitáctil por panel resistivo.....	5
Figura 1.3 Modelo multitáctil por onda acústica.....	6
Figura 1.4 Modelo multitáctil capacitivo.....	7
Figura 1.5 Galgas extensiométricas.....	8
Figura 1.6 Simulación de señal dispersiva.....	9
Figura 1.7 Modelo touch por pulsos acústicos.....	10
Figura 1.8 Modelo touch por infrarrojos y método FTIR.....	11
Figura 1.9 Modelo touch por infrarrojos y método DI.....	12
Figura 1.10 Modelo touch por infrarrojos y método DSI.	13
Figura 1.11 Modelo touch por infrarrojos y método LLP.....	14
Figura 1.12 Espectro electromagnético.....	15
Figura 1.13 Nivel de transmitancia en los glóbulos ópticos.....	16
Figura 1.14 Penetración de la radiación en la piel.....	17
Figura 1.15 Interior de un led.....	17
Figura 1.16 Diodo láser.....	18
Figura 1.17 Comparación del espectro de un led y un láser.....	18
Figura 1.18 Fotodiodo.....	20
Figura 1.19 Fototransistores.....	21

Figura 1.20 Diagrama de un multiplexor con compuertas lógicas.....	22
Figura 1.21 Decodificador.....	23
Figura 1.22 PIC16F628A.....	24
Figura 1.23 Microprocesador.....	25
Figura 1.24 Cyclone V GX Starter Kit (FPGA) MAX V Development Board (CPLD).	26
Figura 1.25 Raspberry Pi.....	27
Figura 1.26 Arduino UNO.....	28
Figura 1.27 Ventana del programador Arduino.....	29
Figura 2.1 Diagrama esquemático de recepción.....	38
Figura 2.2 Modelo final y circuitos impresos para el módulo de recepción.....	39
Figura 2.3 Diagrama esquemático de control.....	40
Figura 2.4 Modelo final y circuitos impresos para el módulo de control.....	41
Figura 2.5 Conexión de las resistencias en el circuito.....	43
Figura 2.6 Conexión de los multiplexores a los fotodiodos.....	43
Figura 2.7 Conexión de los multiplexores de control.....	44
Figura 2.8 Conexiones del marco multitáctil al Arduino.....	45
Figura 2.9 Esquema del marco multitáctil.....	46
Figura 3.1 Marco multitáctil de Music Computing.....	61
Figura 3.2 Marco multitáctil multipunto de Interactuando.....	62
Figura 3.3 Marco multitáctil multipunto de Peau Productions.....	63
Figura 3.4 Aula del Futuro.....	64

TABLAS

Tabla 1.1 Tabla de rangos de longitud de onda y sus respectivos términos.....	15
Tabla 1.2 Precauciones del manejo de láseres según su clase.....	19
Tabla 3.1 Tabla de precios de Music Computing.....	69
Tabla 3.2 Tabla de precios de Peau Productions.....	73
Tabla 3.3 Tabla de precios de nuestra propuesta.....	76

RESUMEN

El proyecto que se maneja en esta tesis describe el prototipo de un frame multitáctil utilizando láseres infrarrojos para la detección de los objetos. La tesis está dividida en 5 capítulos los cuales se describen a continuación.

En el capítulo 1 se mencionan y describen los conceptos básicos y especificaciones necesarias para que los lectores entren en contexto a los conceptos concernientes a las tecnologías interactivas táctiles. Se da una breve explicación de lo que son las superficies multitáctiles y los frames multitouch, también se explica el funcionamiento básico de las tecnologías actuales para la manipulación de los frames mencionando algunas de las ventajas y desventajas de cada uno de ellos. Se describirá brevemente el espectro de luz infrarroja, así como, los elementos que la pueden emitir y recibir y algunos de los parámetros que posee. Por último, se mencionan los elementos de control que se podrían utilizar para la detección de huellas en una superficie multitoque, junto con algunas de sus características y la explicación del por qué elegir el Arduino como el medio de control.

En el capítulo 2 se plantea el desarrollo del diseño a realizar para nuestro proyecto al igual que la tecnología que se usará. Se mencionan las mejoras que se harán con respecto a otros diseños comercializados actualmente, mencionando las ventajas y desventajas que se tienen con ellos. Se dan las características y especificaciones de la estructura física que tendrá el diseño, así como los materiales que se utilizan para la realización del circuito correspondiente. Se explica el funcionamiento de cada uno de los módulos que se diseñarán junto con su circuito y la función de cada uno de sus componentes. Por último, se explica cómo se lleva a cabo la comunicación USB que tendrá con la computadora y la manera en que ésta lo interpretará, al igual que la programación que tendrá el Arduino para poder hacer cada una de las funciones que se mencionaron anteriormente y la explicación del código que se utilizará.

En el capítulo 3 se mencionan los marcos que están actualmente en el mercado y compararemos algunas de sus características como el funcionamiento, aplicaciones y costos con nuestro diseño.

En el capítulo 4 se mencionan los resultados esperados en el diseño de este prototipo y la manera en la que se puede incorporar a un entorno educativo y finalmente en el capítulo 5 las conclusiones que se tendrá al aplicarlo al entorno educativo y sus respectivas observaciones.

INTRODUCCIÓN

La tecnología está dando grandes saltos en estos últimos años y muchas de las cosas que hace poco eran novedad, hoy, ya no lo son. Es por eso que es necesario estudiar esas utilidades y mejorarlas, no solo por el hecho de consumir un nuevo producto, al contrario, tomar el que tenemos y sacarle el máximo provecho.

Hoy en día se pueden encontrar nuevos equipos de cómputo, tabletas electrónicas, celulares (Smartphone), gadgets, etc. Cada uno con su propio estilo pero muchos de ellos tienen algo en común, son táctiles o touch. También es cierto que no hace mucho las cosas no eran táctiles y muchos consumidores compraron pantallas o computadoras que ahora creen obsoletas, pero es ahí donde podemos sacarle gran provecho a las cosas que consumimos sin necesidad de desecharlas.

Por otro lado, se comprueba que el entorno educativo actual está decayendo y se necesitan nuevas herramientas interactivas de enseñanza que acaparen la atención del estudiante y que ayuden al profesor en sus actividades docentes. Con estas nuevas herramientas se busca un trabajo más colaborativo aprovechando el carácter de sistemas multiusuario que consideran, ya que varios estudiantes participarían y no solo el estudiante que tenga el equipo (teclado y ratón) sea el único que comprenda los conceptos. Muchas de estas herramientas están diseñadas para la diversión, pero se pueden adaptar para una mejor educación siendo más interactivas y amigables.

Para lograr lo antes mencionado, proponemos diseñar un frame táctil que ayude a la interacción de los usuarios con el equipo que se tenga a la mano sin necesidad de renovar todos los componentes.

Método.

Analizar las tecnologías existentes de los diferentes dispositivos multitáctiles, enfocándonos principalmente en aquellos que utilicen una tecnología basada en infrarrojos, posteriormente diseñar un sistema que permita obtener la ubicación precisa de un objeto que se encuentre dentro del área de proyección sin interferir con la imagen proyectada y, finalmente, crear una comunicación entre este dispositivo y un equipo de cómputo para su manipulación.

OBJETIVO

Analizar el funcionamiento de los diferentes frames en el mercado y proponer un diseño que permita la interacción de usuarios con un entorno computacional (Windows) de manera multitáctil a través de pantallas de tecnología LCD, led o plasma. Por otra parte, se busca que los elementos del frame a desarrollar sean de fácil acceso y sobre todo, económicos para los usuarios.

Se pretende que este frame se utilice a futuro en las nuevas aulas de laboratorio de bachillerato, así como en diferentes áreas académicas de la UNAM.

Resultados esperados.

Tener un frame capaz de manipular con gran precisión el entorno Windows, que sea adaptable a diferentes tipos de pantallas, de materiales resistentes, duraderos, ligeros y que sea competitivo dentro de la economía actual.

CAPÍTULO 1

CONCEPTOS BÁSICOS

1.1. SUPERFICIES MULTITÁCTILES

Una superficie multitáctil es un área en la cual el usuario puede interactuar con una computadora sin necesidad de utilizar algún dispositivo como un teclado o ratón, solamente usando el toque de sus dedos.

Estas superficies se encuentran actualmente en desarrollo al igual que la tecnología que se usa en ellas.

Una de las características principales de estas superficies es el tamaño que estas tienen, ya que su principal objetivo es que varias personas sean capaces de interactuar con ella. Estas superficies pueden ir desde el tamaño de una hoja carta hasta paredes enteras de una habitación.

Actualmente, estas superficies son comercializadas y dirigidas mayoritariamente al entretenimiento y la diversión, ya que se encuentran en las discotecas en donde los DJ pueden manipularlas dando una mejor impresión y un mejor control de la música que está reproduciendo. También se están colocando como mesas en restaurantes donde los propios usuarios pueden pedir las órdenes a los comensales además de acceder a otras opciones dentro de ellas. Por otro lado y no tan avanzadamente se encuentran en el área de la educación donde grupos de niños y estudiantes se reúnen alrededor de estas superficies y logran interactuar y realizar algún trabajo de manera grupal y por ende ayudan a la convivencia y al trabajo en grupo, Cada una de estas tecnologías tienen sus pros y sus contras pero la mayoría de ellas son relativamente costosas. Las más usuales son la capacitiva, la resistiva y por supuesto la infrarroja.

La capacitiva, siendo la más comercial, acapara todo el mercado pero se considera la opción más cara para los usuarios. Una de las principales ventajas es que es más precisa, permite varios toques a la vez y es mejor la calidad de imagen. Por otro lado, la tecnología resistiva está siendo desplazada debido a que presenta pérdidas de calidad de imagen, sin embargo es menos propensa a la interferencia de polvo. Y, por último, la infrarroja que

aunque es una tecnología conocida, es poco popular en dispositivos móviles. Sin embargo se considera muy accesible y económica para los usuarios.

1.2. DEFINICIÓN DE MARCOS MULTITÁCTILES.

La palabra frame es una palabra inglesa que significa fotograma o cuadro, usada comúnmente para medir los fotogramas que pasan en un segundo y que producen la sensación de movimiento. La palabra frame se utiliza para referirnos al dispositivo en forma de rectángulo que se montará sobre y alrededor de una pantalla y que nos ayudará a crear nuestra superficie multitouch. Multitouch se refiere a una pantalla táctil que reconoce simultáneamente múltiples puntos de contacto, pero debido a que ambas palabras son inglesas, de ahora en adelante al referirnos al término frame y multitouch lo haremos como “marco” y “multitáctil”.

Existen actualmente diferentes diseños de estos marcos, al igual que tecnologías que lo hacen funcionar, y todos ellos son solamente interfaces entre el dispositivo y el sistema operativo en el cual funcionan. La gran mayoría funcionan bajo sistema Windows por su popularidad y la familiarización de la población. Otros sistemas que empiezan a utilizarse, por su simplicidad y porque actualmente se encuentran en otros dispositivos, son los basados en Android, el cual se encuentra en celulares, tabletas, algunas computadoras o como sticks para la transformación de televisores en Smart TV. Otros marcos trabajan con sistemas operativos basados en Linux.

Por otro lado, se encuentran las tecnologías que necesitan una presión para poder detectar un toque, en las cuales la tecnología de detección es en la propia superficie por lo que el marco no existe o es mínimo; el inconveniente de estas tecnologías es que se degradan con el uso y pueden dejar de funcionar, por ello se utilizan en lugares menos concurridos y con mayor vigilancia.

Estos marcos utilizan la conexión USB de las computadoras para poder interactuar con ellas gracias a los controladores de interfaz humana que pueden utilizarse. La mayoría de estos marcos traducen por sí mismos los toques que suceden dentro del área de proyección

y simplemente mandan las coordenadas al computador para decirle donde ocurrió un toque y que él decida qué acción ejecutar.

1.3. FUNCIONAMIENTO DE LOS MARCOS ACTUALES.

1.3.1. Estructura física

Las superficies que necesitan la interferencia de los objetos para poder captar los puntos de toque son manufacturados con biseles de acero o de aluminio, los cuales aportan rigidez al marco, ya que en su interior viene incorporada la tecnología que hace posible la detección de movimiento, y necesitan estar fijos para no dar falsas coordenadas ya que generalmente los sensores se encuentran en el marco. Esta estructura cada vez se está haciendo más ligera y delgada para su mejor transportación y ensamble, además de que logra que su incorporación a la pantalla sea imperceptible y promueve que los usuarios se acostumbren fácilmente a ella. También es usada principalmente en lugares donde existe mayor afluencia de personas ya que el área de proyección puede estar reforzada por acrílico y no interfiere con la detección de los toques.

1.3.2. Funcionamiento.

En el mercado existen diferentes métodos para lograr que una superficie sea táctil y cada uno de ellos tiene sus ventajas y desventajas. Algunas superficies son más costosas y otras requieren más espacio debido a su necesidad de componentes extras.

La mayoría de las superficies multitáctil utilizan sensores ópticos, lo cual quiere decir que necesitan un dispositivo que capture visualmente los movimientos, siendo generalmente una webcam modificada; además necesitan un área de proyección que no interfiera con la cámara, por lo general basado en proyectores, que ocupan un espacio extra.

También existen otras tecnologías que se acoplan directamente a la proyección, ya sea por proyectores o por pantallas, las cuales usan menor espacio pero son más costosas.

A continuación se explican brevemente las tecnologías más usadas dentro del área de las superficies multitáctil, dando las ventajas y desventajas que puedan tener.

Infrarrojos

Las pantallas táctiles de matriz infrarroja consisten de un arreglo de emisores y receptores infrarrojos colocados horizontal y verticalmente. Cada receptor tiene en el lado opuesto un emisor, de tal forma que al tocar con un objeto la pantalla se interrumpe un haz infrarrojo vertical y otro horizontal, tal como se muestra en la figura 1.1, permitiendo de esta forma localizar la posición exacta en que se realizó el contacto.

La ventaja de este tipo de pantallas es que son muy resistentes, ya que la matriz no está directamente en contacto con la pantalla y permite tener una distancia apropiada para colocar una protección entre estas dos, por lo que son utilizadas en exteriores y en lugares muy concurridos, donde las personas libremente pueden manipularla.

La desventaja de este método básicamente consiste en que puede detectar fácilmente cualquier obstrucción de un objeto no deseado y tener por lo tanto una coordenada errónea.

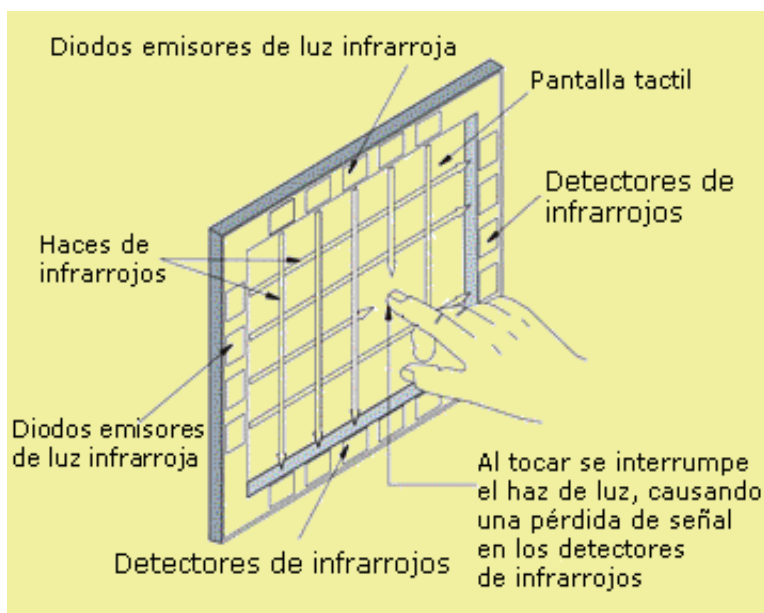


Figura 1.1 Modelo multitáctil por infrarrojos. Figura tomada de <http://touchscreensune.blogspot.mx/2007/07/c-como-funcionan.html>

Resistiva.

Una pantalla táctil resistiva está formada por varias capas. Dos de estas capas tienen finas camaras de material conductor y están separadas por un espacio mínimo. El objetivo principal de esta separación es que cuando algún objeto toque la superficie de la capa, las dos capas conductoras entren en contacto en un punto concreto y de esta manera se produzca un cambio en la corriente eléctrica que permita a un controlador calcular la posición del punto en el que se ha tocado la pantalla midiendo la resistencia. Algunas de estas pantallas pueden medir la presión que se ha ejercido sobre la misma. Un ejemplo de este tipo de superficies se muestra en la figura 1.2.

La ventaja de usar este tipo de pantallas es que son accesibles, además de que su diseño permite que las interferencias externas como el polvo, agua o algunos objetos no muy pesados no creen puntos erróneos en la pantalla.

La desventaja que tiene es que es propensa a que se rompa con facilidad con algún objeto afilado o con punta, ya que suelen ser muy delgadas, y por otro lado tienen una pérdida de brillo de la imagen original de aproximadamente el 25 %, por lo que a mayor resistencia menor será la nitidez.

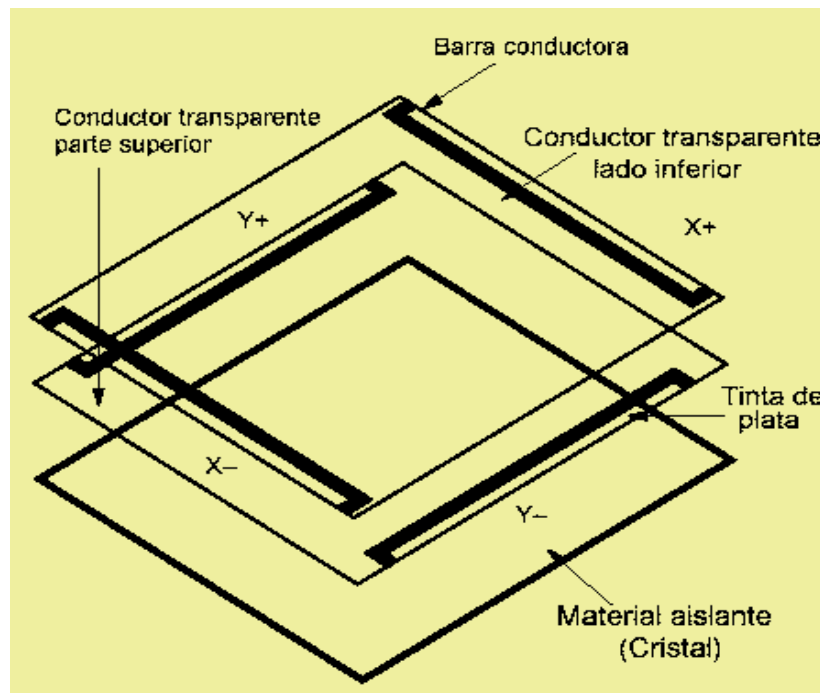


Figura 1.2 Modelo multitáctil por panel resistivo. Figura tomada de <http://touchscreensune.blogspot.mx/2007/07/c-como-funcionan.html>

Onda acústica superficial.

Las pantallas que poseen la tecnología de onda acústica superficial (SAW, Surface Acoustic Wave) utilizan ondas ultrasónicas que se transmiten sobre la pantalla táctil. Cuando la pantalla es tocada, una parte de la onda es absorbida, tal como se muestra en la figura 1.3, este cambio en las ondas de ultrasonido permite registrar la posición en la que se ha tocado la pantalla y enviarla al controlador para que pueda procesarla.

La ventaja de esta tecnología es que puede ser muy precisa a la hora de detectar un toque sobre la pantalla pero esta ventaja viene enlazada con la sensibilidad que tendrán los sensores para poder detectarlo y por consiguiente su costo.

La presencia de contaminantes u objetos interfieren fácilmente las ondas acústicas por lo que afectan su funcionamiento en exteriores

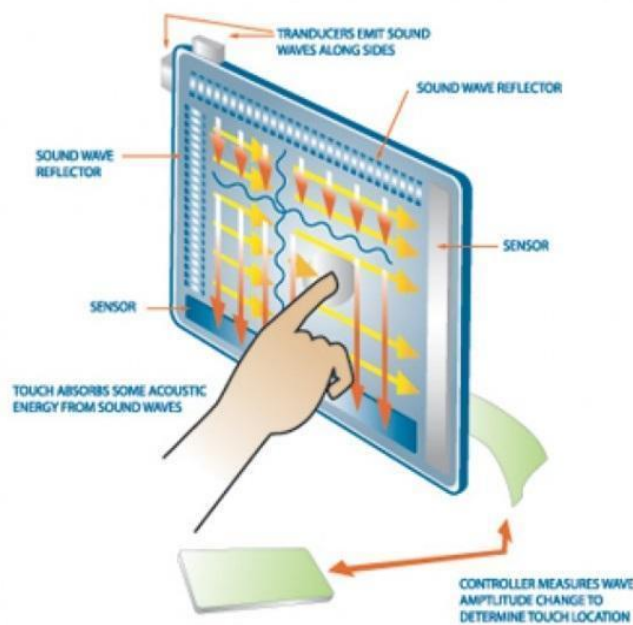


Figura 1.3 Modelo multitáctil por onda acústica. Figura tomada de <http://degreedix.com/gadgets/touchscreen-technology>

Capacitiva.

Las pantallas táctiles capacitivas tienen un panel de vidrio recubierto con una capa de material conductor transparente, lo que permite tener una capacitancia en la pantalla. En cada una de las esquinas de esta pantalla táctil se aplica una tensión, y cuando el usuario toca este panel se altera la capacitancia. Los sensores situados en cada esquina miden la variación capacitiva resultante con un campo de referencia y envían la información acerca de este evento al controlador para su procesamiento; logrando que se identifique la coordenada de toque, este proceso se puede observar en la figura 1.4.

La ventaja de estas pantallas es que solo pueden ser manipuladas por objetos que tengan capacidad de conducir la electricidad, de lo contrario los sensores no detectarían ninguna variación sobre la superficie. Otra ventaja de esta tecnología es su precisión, en la determinación del punto de toque, y la nitidez de la imagen real. Estas 2 últimas ventajas vienen enlazadas con su principal desventaja; su alto costo de fabricación, lo que da como resultado que muy pocos puedan adquirir un marco con este tipo de tecnología.

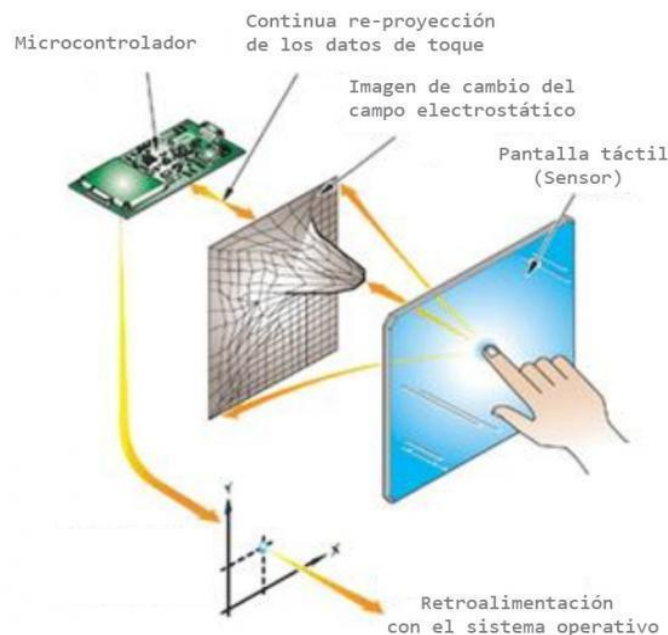


Figura 1.4 Modelo multitáctil capacitivo. Figura tomada de <http://www.sky-technology.eu/Projected-Capacitive-Technology-explained>

Galgas extensiométricas

Las pantallas con galgas extensiométricas poseen una estructura elástica de forma que se pueden utilizar para determinar la posición en que ha sido tocada la pantalla a partir de las deformaciones producidas en las mismas. Este tipo de tecnología también puede medir profundidad y presión sobre el eje en la que se toca. En la figura 1.5 se muestra la estructura interna de las galgas extensiométricas.

Por la resistencia que poseen son usadas en exteriores y en lugares comúnmente manipulables por lo que esta resistencia es la ventaja principal de estas pantallas.

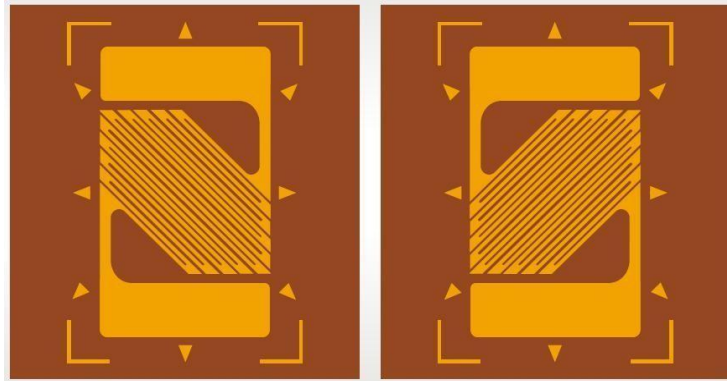


Figura 1.5 Galgas extensiométricas. Figura tomada de

<http://www.directindustry.com/prod/bcm-sensor-technologies-bvba/shear-strain-gauges-56103-659543.html>

Señal dispersiva.

Este tipo de superficies tienen sensores en sus contornos los cuales les permiten medir la energía mecánica del cristal. El procesador que manipula los algoritmos e interpreta las señales de estos sensores permite obtener la información del punto exacto en el que ha sido tocada la pantalla. Un ejemplo de simulación de señal dispersa se puede apreciar en la figura 1.6.

La ventaja de este tipo de superficies es que son muy resistentes a los objetos ya que el cristal no contiene directamente los componentes de medición por lo que se pueden obtener

cristales más duros, como no posee mecanismos en su superficie logra también una mayor nitidez de la imagen de la pantalla.

La desventaja de esos sensores es que no pueden detectar el toque de un objeto si éste se ha quedado tocando la pantalla y por lo tanto no existe algún movimiento que identificar.

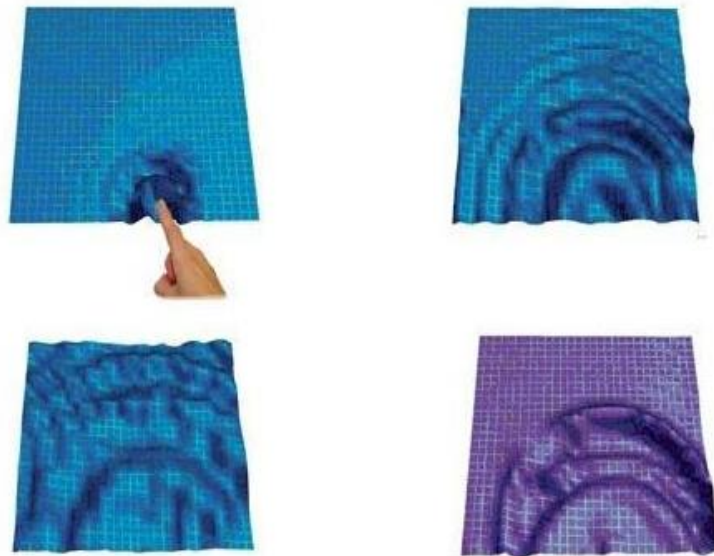


Figura 1.6 Simulación de señal dispersiva. Figura tomada de <http://multimedia.3m.com/mws/mediawebserver?mwsId=66666UF6EVsSyXTtmxTXoxfaEVtQEVs6EVs6EVs6E666666--&fn=DST%20Tech%20Profile.pdf>

Pulso Acústico.

Estos tipos de superficies utilizan cuatro transductores piezoeléctricos situados en cada lado de la pantalla para convertir la energía mecánica del contacto en una señal electrónica. La estructura de una superficie por pulsos acústicos se puede apreciar en la figura 1.7. Esta señal es posteriormente convertida en una onda de sonido, la cual es comparada con el perfil de sonido preexistente para cada posición en la pantalla.

La ventaja de estas superficies es que no necesitan ningún mecanismo sobre ellas y es posible colocar cualquier tipo de cristal, logrando así una mayor durabilidad de la superficie. Otra ventaja, es la buena nitidez que tienen las imágenes, a pesar del cristal que se coloque.

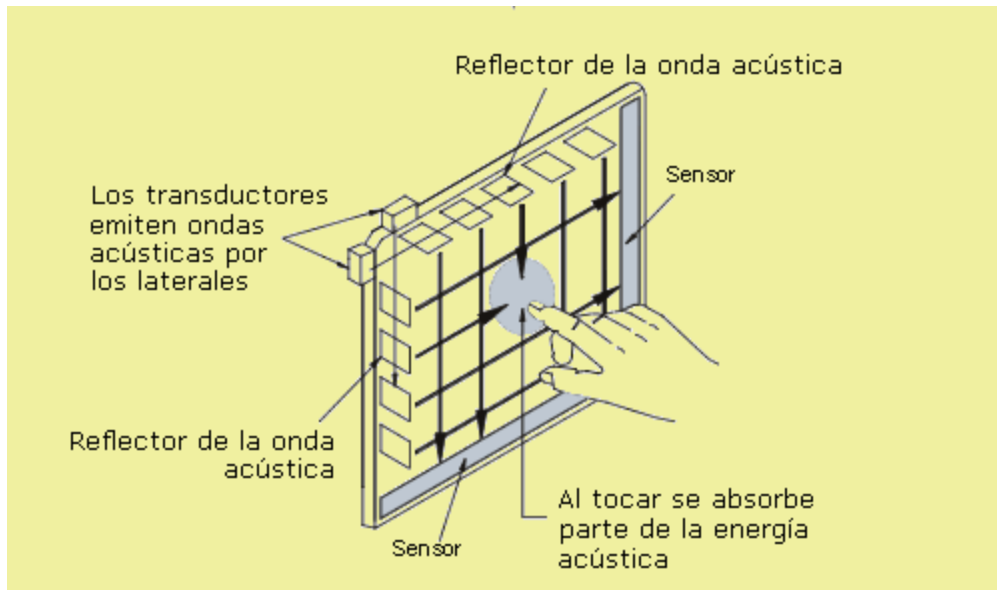


Figura 1.7 Modelo touch por pulsos acústicos. Figura tomada de <http://touchscreensune.blogspot.mx/2007/07/c-como-funcionan.html>

Ópticos.

Existen otras técnicas relativamente nuevas basadas en sensores ópticos siendo las principales: reflexión interna total frustrada (FTIR), iluminación difusa (DI), plano de luz láser (LLP), plano de luz LED (LED-LP) y finalmente la superficie de iluminación difusa (DSI).

FTIR.

El principio establece que las ondas electromagnéticas transmitidas por el interior de un material son completamente reflejadas si el material tiene un índice de refracción superior a los materiales situados en su frontera y el ángulo de incidencia es superior a un cierto ángulo crítico. Sin embargo otro material situado en la frontera puede frustrar esta reflexión interna, dejando escapar la luz. Las superficies que utilizan este método por tanto constan generalmente de un panel acrílico con un marco de led infrarrojo alrededor, que inyectan luz dentro. Al ser tocado el acrílico, la luz es reflejada en el punto de contacto del dedo debido a su mayor índice de refracción. Una cámara sensible al espectro infrarrojo situada

al otro lado del panel puede detectar estas reflexiones, tal como se muestra en la figura 1.8. Tratando estas imágenes se pueden conocer los puntos de contacto. Al ser la superficie transparente se puede usar un proyector detrás del panel para obtener una pantalla multitáctil.

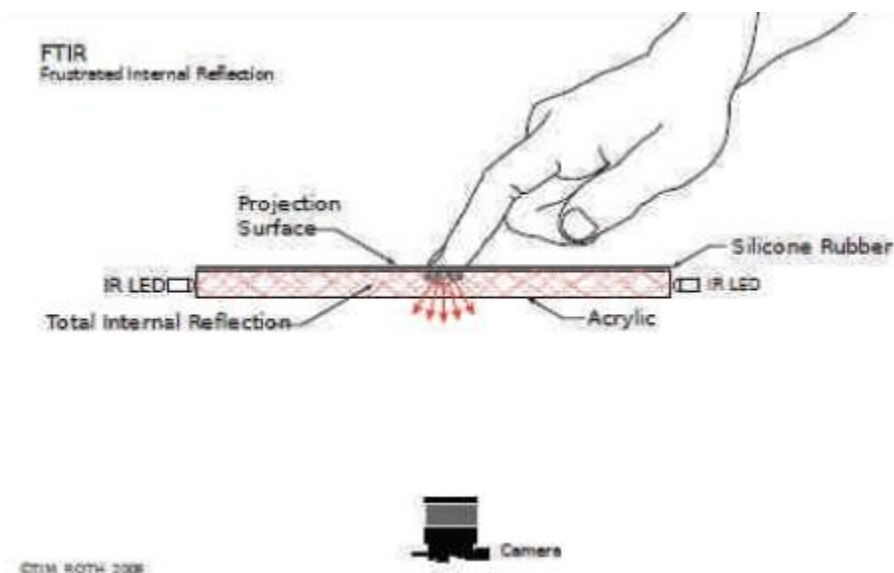


Figura 1.8 Modelo touch por infrarrojos y método FTIR. Figura tomada de http://nuicode.com/attachments/download/115/Multi-Touch_Technologies_v1.01.pdf

DI.

Como se aprecia en la figura 1.9, la configuración es similar a la de FTIR, pero esta vez la luz infrarroja se distribuye por dentro de una mesa cerrada (le afecta la iluminación ambiental) y se lanza desde lámparas de infrarrojos. El objetivo es tener una iluminación difusa en la superficie. Este sistema permite detectar objetos por su forma y admite capas transparentes de protección por encima de la superficie. Al recibir un contacto, la cámara lo percibe porque refleja la luz infrarroja. Sin embargo, un objeto situado en un plano superior al de la superficie es difuminado y por lo tanto no leído.

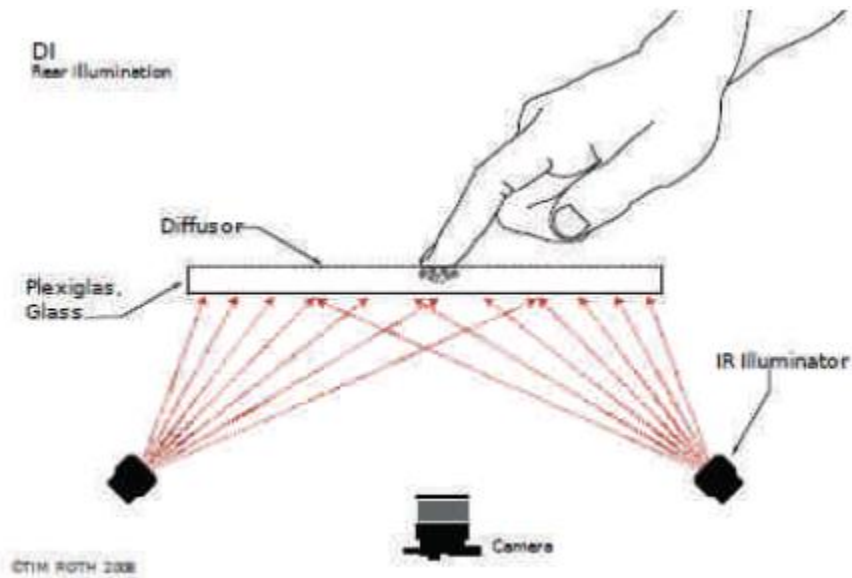


Figura 1.9 Modelo touch por infrarrojos y método DI. Figura tomada de http://nuicode.com/attachments/download/115/Multi-Touch_Technologies_v1.01.pdf

DSI.

El inventor de esta técnica derivada de las dos anteriores es Tim Roth. Es similar a FTIR porque lleva luz infrarroja por dentro del acrílico, mediante la colocación de led en los bordes, y porque no se necesita tapar la mesa; pero, al tener menos contraste, sufre más con las condiciones lumínicas ambientales. La diferencia principal respecto a FTIR es que el acrílico lleva por dentro unas pequeñas partículas que actúan como espejos y expanden la luz por todo el acrílico (incluso por arriba y debajo de él), tal como se observa en la figura 1.10. Esto resulta en una iluminación realmente difusa en la superficie (el gran reto al construir una mesa DI) y además permite la lectura de fiduciales (no se necesitan siliconas u otros materiales encima del acrílico).

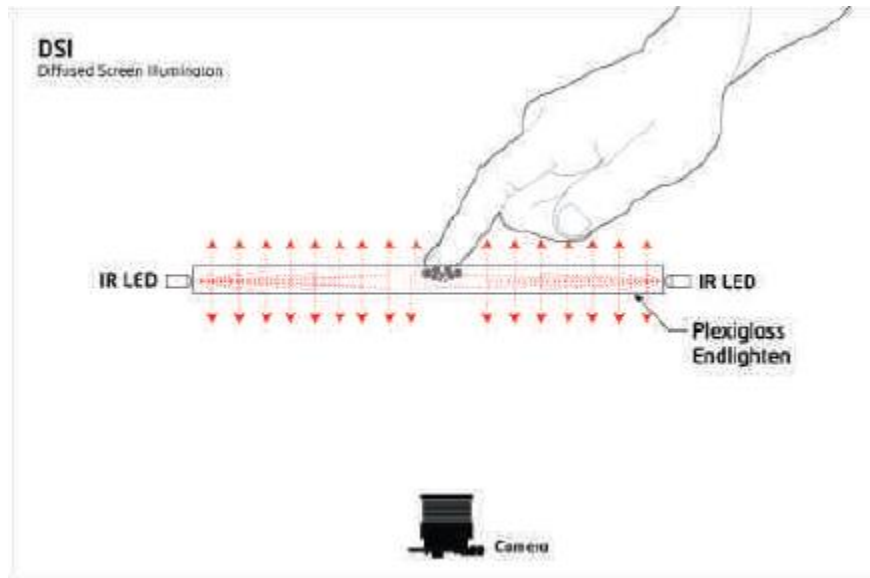


Figura 1.10 Modelo touch por infrarrojos y método DSI. Figura tomada de http://nuicode.com/attachments/download/115/Multi-Touch_Technologies_v1.01.pdf

Dos o más sensores son situados alrededor de la pantalla, habitualmente en las esquinas y en todas ellas se utiliza una cámara de detección infrarroja la cual capta la interacción de los dedos con la superficie por medio de una sombra y crea una retroalimentación visual que es la proyección en esta superficie, el inconveniente de este método es que se necesita un espacio mayor para la colocación de la cámara y poder así captar todo el área visible de la superficie

LLP.

Para usar la técnica LLP, plano de luz laser, tenemos que usar una lente generadora de línea, las cuales existen de 90 y 120°. Estas lentes generadoras sirven para crear un plano de luz de 90 o 120° sobre la superficie, el cual debe ser aproximadamente de 1 mm de grosor. Cuando el dedo toque la superficie se registrará como un punto de toque, esto se puede apreciar en la figura 1.11. Los láseres infrarrojos son una forma fácil y por lo general económica de crear una configuración multitouch utilizando el método LLP. La mayoría de las configuraciones usan de 2 a 4 láseres, colocados en las esquinas de la superficie.

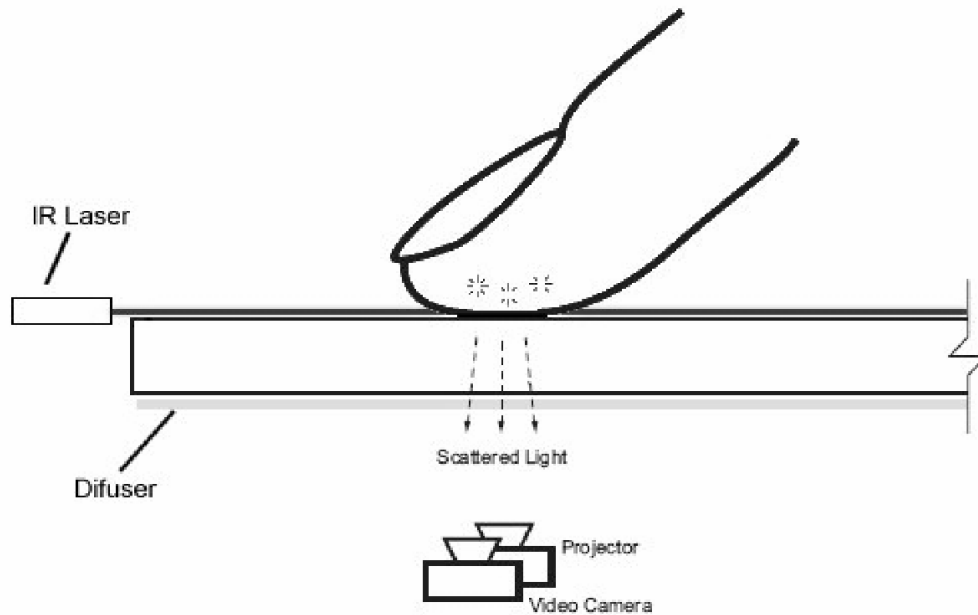


Figura 1.11 Modelo touch por infrarrojos y método LLP. Figura tomada de http://nuicode.com/attachments/download/115/Multi-Touch_Technologies_v1.01.pdf

1.4. LUZ INFRARROJA.

La radiación infrarroja también es conocida como radiación térmica y es emitida por todos los cuerpos calientes, desde los focos incandescentes hasta fabricas fundidoras de metal.

La longitud de onda de la luz infrarroja se encuentra comprendida entre las microondas y la luz visible (780nm y 1mm). La comisión Internacional de Iluminación divide esta banda en tres categorías: IRA que comprende de los 780nm a 1.4um, IRB que comprende de 1.4um a 3um y por ultimo IRC que comprende de los 3um a 1mm.

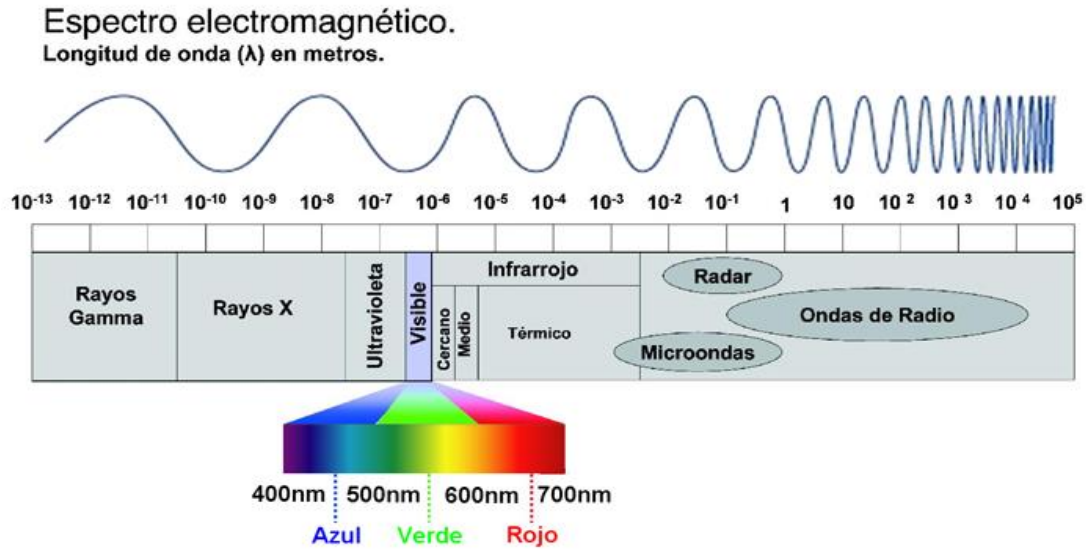


Figura 1.12 Espectro electromagnético. Figura tomada de <http://fisica.laguia2000.com/energia/escala-de-las-radiaciones-electromagneticas>

Dentro del espectro electromagnético, las distintas longitudes de onda tienen energías considerablemente diferentes para ocasionar efectos biológicos, por lo cual, el ultravioleta y el infrarrojo pueden subdividirse aún más. Tabla 1.1

Subdivisiones del ultravioleta y el infrarrojo

Longitud de onda	Banda	Termino
100nm a 280nm	UVC	Ultravioleta lejano
280nm a 315nm	UVB	Ultravioleta medio
315nm a 280nm	UVA	Ultravioleta cercano
400nm a 700nm	LUZ	Visible
700nm a 1400nm	IRA	Infrarrojo cercano
1.4um a 3um	IRB	Infrarrojo medio
3um a 100um	IRC	Infrarrojo lejano

Tabla 1.1 Tabla de rangos de longitud de onda y sus respectivos términos. Tabla de autoría propia.

Estos intervalos están de acuerdo con lo definidos por la Comisión Internacional de Iluminación (CIE). Además, si bien no existe un límite claro entre longitudes de onda visible y UV, los límites de uso general se dan en esta tabla.

Las radiaciones se pueden clasificar en radiaciones ionizantes y no ionizantes.

Las radiaciones ionizantes poseen un alto poder energético, teniendo la capacidad para penetrar en la materia y desprender electrones de los átomos que la constituyen, provocando una ionización. Las radiaciones ionizantes engloban a los rayos X, rayos γ , partículas α , partículas β y neutrones.

Las radiaciones no ionizantes, no producen ionización de la materia, aunque pueden causar otros efectos térmicos o fotoquímicos. Las frecuencias de las radiaciones no ionizantes son iguales o menores de 3000 THz y sus energías son iguales o menores a 12.4 eV

La exposición a la radiación infrarroja puede deberse a diversas fuentes naturales o artificiales. Esta radiación puede estar constituida por una sola longitud de onda o emisión espectral, como lo hace un láser, o tener una amplia gama de longitudes de onda, como la producida por algunos diodos led.

Generalmente la radiación óptica no llega a gran profundidad del tejido biológico por lo que la mayor absorción es en la piel y en los ojos, donde puede provocar enrojecimiento y hasta quemaduras. La radiación que se presenta en ojos y en piel se puede observar en las figuras 1.13 y 1.14 respectivamente.

Transmitancia Ocular

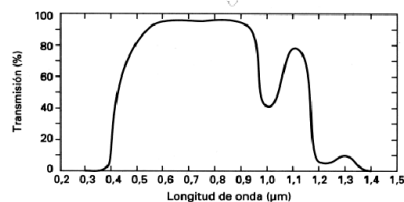


Figura 4-6. Transmitancia espectral global a través de la córnea, el humor acuoso, el cristalino y el humor vítreo.

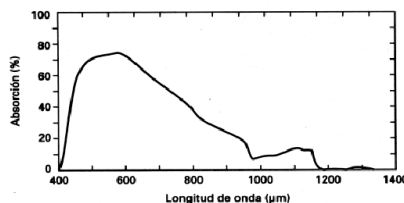
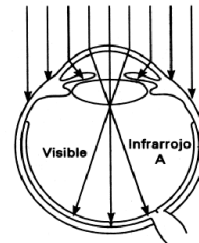
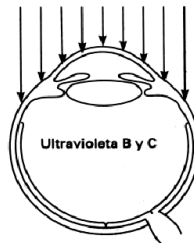


Figura 4-7. Absorción espectral de la retina y la coroides.

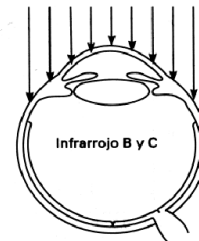
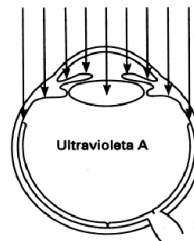


Figura 1.13 Nivel de transmitancia en los glóbulos ópticos. Figura tomada de <http://www.uv.es/artigas/Opt%20Oftal%20II/Tema%20V-Resumen.pdf>

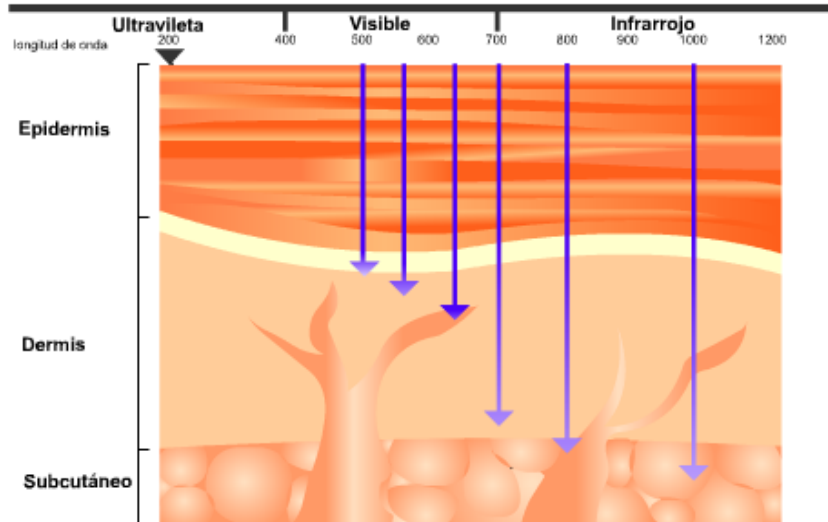


Figura 1.14 Penetración de la radiación en la piel. Figura tomada de http://nupen.com.br/Revista_esp/fund_fisicos4_3.php

1.4.1. Led infrarrojo.

Un led infrarrojo es un diodo semiconductor en el cual, cuando se encuentra en polarización en directa, los electrones pueden recombinarse con los huecos liberando energía en forma de fotones, a este intercambio de electrones se le denomina electroluminiscencia y el color que emite este diodo es proporcional a la banda de energía que tiene el semiconductor. En la figura 1.15 se puede apreciar como es física e internamente un led.

La longitud de onda que emiten estos diodos va desde los 700nm hasta los 100um.

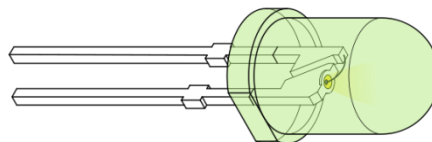


Figura 1.15 Interior de un led. Figura tomada de <http://tectonicablog.com/?p=57360>

1.4.2. Diodo láser.

La palabra LASER proviene de las siglas en inglés: Light Amplification by Stimulated Emission of Radiation que significa: Amplificación de luz por Emisión estimulada de radiación. El principio de funcionamiento de un diodo laser está basado en una emisión estimulada, donde un fotón permite inducir a un electrón que se encuentra en un estado excitado a un estado en reposo, esta estimulación provoca que los demás electrones pasen a este estado con la misma sincronía, liberando energía en forma de iluminación. En la figura 1.16 se muestra un diodo laser y en la figura 1.17 se muestran los espectros emitidos por un led y un láser, observando que el espectro del led abarca una mayor longitud de onda y el láser una banda muy estrecha.



Figura 1.16 Diodo láser. Figura tomada de

<http://es.farnell.com/laser-components/2008369/diodo-laser-655nm-1-version/dp/1272661>

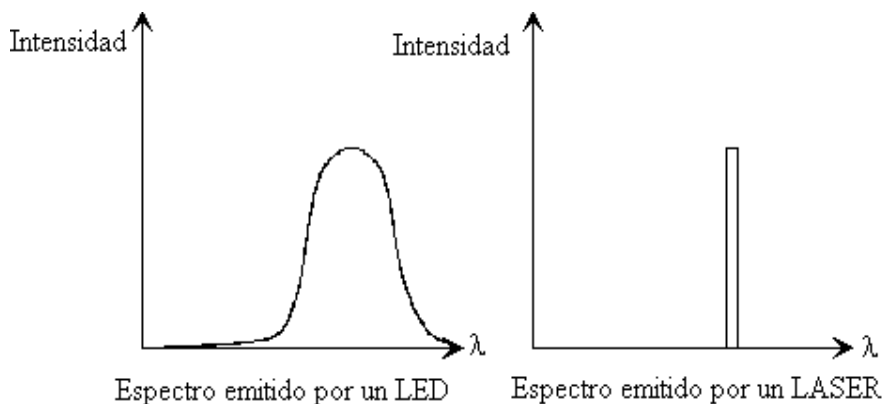


Figura 1.17 Comparación del espectro de un led y un láser. Gráfica tomada de

<http://www.electronicafacil.net/tutoriales/Diodo-Laser.php>

Cada láser lleva de forma permanente y en lugar visible una o más etiquetas de aviso, según la clase o grupo de riesgo al que pertenezca, por lo que cada equipo llevará etiquetas rectangulares con frases de advertencia que permitirán al usuario conocer el potencial de riesgo al que se expone y una señal triangular con el símbolo de peligro por radiación láser. Los símbolos antes mencionados se muestran en la tabla 1.2 al igual que los diferentes tipos de clase que existen para los láseres.

	CLASE 1	PRODUCTO LASER CLASE 1
	CLASE 2	RADIACION LASER. NO MANTENGA LA VISTA EN EL HAZ. PRODUCTO LASER CLASE 2.
	CLASE 3A	RADIACION LASER. NO MANTENGA LA VISTA EN EL HAZ NI LO MIRE DIRECTAMENTE CON INSTRUMENTOS OPTICOS. PRODUCTO LASER CLASE 3A.
	CLASE 3B	RADIACION LASER. EVITE LA EXPOSICION AL HAZ. PRODUCTO LASER CLASE 3B.
	CLASE 4	RADIACION LASER. EVITE LA EXPOSICION OCULAR O LA PIEL A RADIACIONES DIRECTAS O DIFUSAS. PRODUCTO LASER CLASE 4.
	ABERTURA LASER	EVITAR LA EXPOSICION. SE EMITE RADIACION LASER POR ESTA ABERTURA.
	PANELES DE ENCLAVAMIENTO	PRECAUCION, RADIACION LASER EN CASO DE ABERTURA Y DESACTIVACION DE BLOQUEOS DE SEGURIDAD.

Tabla 1.2 Precauciones del manejo de láseres según su clase. Tabla tomada del artículo http://www.insht.es/InshtWeb/Contenidos/Documentacion/FichasTecnicas/NTP/Ficheros/201a300/ntp_261.pdf

1.4.3. Fotodiodo.

Un fotodiodo es básicamente un diodo semiconductor con una unión p-n activa, el cual trabaja con una polarización invertida. Este dispositivo conduce una corriente eléctrica proporcional a la cantidad de luz que lo ilumina (iluminancia). Esta corriente eléctrica fluye en sentido opuesto a la flecha del diodo y se llama corriente de fuga. Estos fotodiodos poseen una respuesta lineal y una velocidad de respuesta que se encuentra en el orden de los nanosegundos. En la figura 1.18 se muestran algunos fotodiodos que existen en el mercado.



Figura 1.18 Fotodiodo. Figura tomada de <http://www.digikey.com.mx/product-highlights/mx/es/vishay-tefd4300-pin-photodiodes/1248>

La mayoría de los fotodiodos vienen encapsulados con una lente que concentra la cantidad de luz que le incide, permitiéndole así que su respuesta sea más precisa.

La diferencia principal entre los fotodiodos y las fotorresistencias es que estos responden a los cambios de iluminación y oscuridad con una velocidad mayor.

1.4.4. Fototransistor.

El fototransistor es básicamente la unión de dos componentes, el de un transistor común y el de un fotodiodo conectado entre la base y el colector, con el cátodo al colector y el ánodo a la base, tal como se muestra en la figura 1.19. Un fototransistor es sensible a la luz, normalmente a la luz infrarroja. La luz que entra en el fototransistor genera una corriente portadora en la base y activa el transistor. Como un dispositivo activo, entonces obtendrá una ganancia en la salida, lo cual no ocurre en el fotodiodo.



Figura 1.19 Fototransistores. Figuras tomadas de
http://www.tme.eu/es/katalog/fototransistores_112306/
<http://www.electronicasi.com/enseanzas/electronica-elemental/aprender-electronica-practicando/aprende-practicando-interruptor-crepuscular/>

1.5. MULTIPLEXORES Y DECODIFICADORES (DEMULTIPLEXORES).

Los multiplexores son circuitos combinatoriales con n entradas, una salida y están dotados de entradas de control capaces de seleccionar una, y sólo una, de las entradas de datos para permitir su transmisión desde la entrada seleccionada a la salida que es única.

La función de un multiplexor consiste en seleccionar una línea entre un número n de líneas de entrada y transmitir este dato a un canal de información único, por lo que un multiplexor es equivalente a tener un conmutador de varias entradas y una salida. En la figura 1.20 se muestra el diagrama de un multiplexor de 8 entradas y una salida conformado por compuertas lógicas.

Los diseños de los multiplexores se realizan como cualquier sistema combinatorio. Un multiplexor actúa como un conversor paralelo-serie.

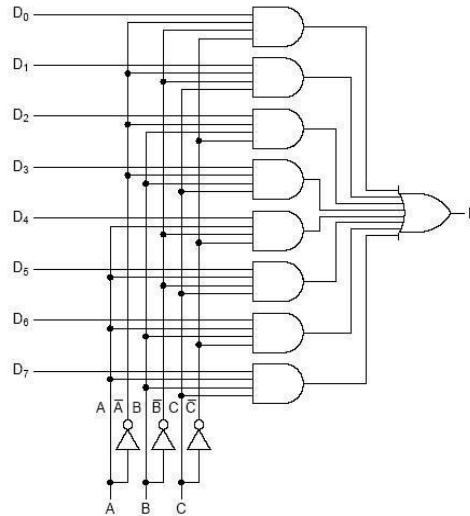


Figura 1.20 Diagrama de un multiplexor con compuertas lógicas. Figura tomada de http://et2-2011-3g.blogspot.mx/2011_06_01_archive.html

El concepto de demultiplexor es similar al de multiplexor, viendo las entradas de datos como salidas y la salida como entrada. En un multiplexor hay varias entradas de datos y sólo una de ellas se saca por el canal de salida. En los demultiplexores hay un único canal de entrada que se saca por una de las múltiples salidas.

El demultiplexor es un dispositivo que puede recibir a través de un medio de transmisión compartido una señal compleja, multiplexada, y separar las distintas señales integrantes de la misma encaminándolas a las salidas correspondientes.

El demultiplexor es un circuito combinacional que puede utilizarse en muchos casos como decodificador y adoptar cualquiera de las funciones que un decodificador realiza. En la figura 1.21 se muestra el diagrama de un demultiplexor de 1 entrada y 4 salidas conformado por compuertas lógicas así como algunas imágenes de decodificadores en el mercado.

Un demultiplexor actúa como un conversor serie-paralelo.

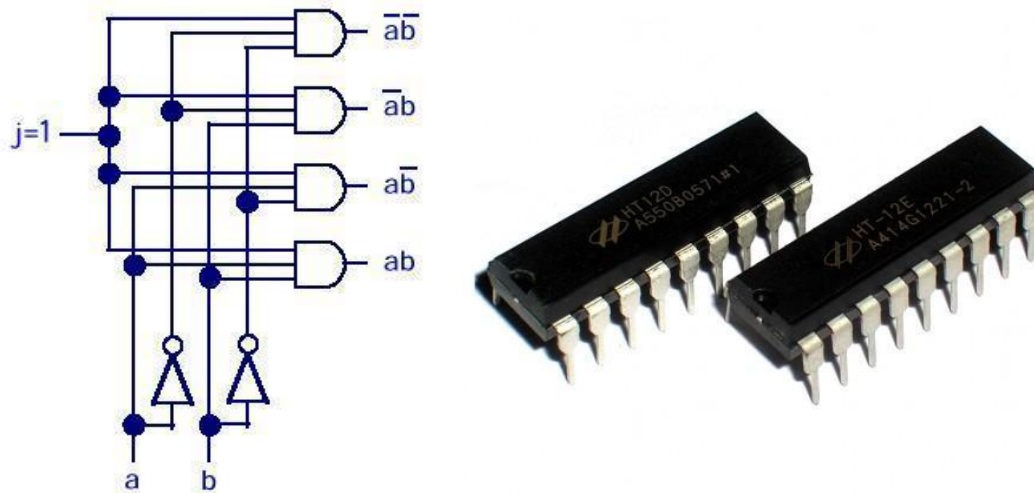


Figura 1.21 Decodificador. Figuras tomadas de <https://sites.google.com/site/electronicadigitaluvfime/3-5-multiplexores-y-demultiplexores-aplicaciones>
<http://www.geekfactory.mx/tienda/componentes-electronicos/ht12d-decodificador-de-control-remoto/>

1.6. DISPOSITIVOS DE CONTROL.

Existen en el mercado diferentes dispositivos los cuales nos pueden ayudar y fungir como el dispositivo de control para nuestro marco. Cada uno de estos dispositivos tiene sus ventajas y desventajas. A continuación se da una breve explicación de cada uno de ellos.

PIC.

Un dispositivo muy común es el PIC, el cual es un circuito integrado desarrollado por microchip Technology Inc, que pertenece al grupo de microcontroladores, ya que integran en el mismo dispositivo los circuitos necesarios para poder realizar un circuito programable.

Los PIC pueden ser de tecnología TTL o CMOS y todos ellos integran en su interior una CPU, una memoria PROM, una memoria RAM, múltiples líneas de entrada I/O, además de dispositivos auxiliares como lo pueden ser generadores de reloj, contadores, etc., según sea el modelo del PIC.

Los modelos más pequeños son fácilmente identificables por su sigla PIC12CXX los cuales poseen 8 pines, hasta llegar a los 40 pines con su sigla PIC17CXX. Se puede apreciar en la figura 1.22 un PIC de 18 pines con sus siglas PIC16F628A.

El PIC es un dispositivo muy versátil y de bajo costo, existiendo en el mercado diferentes gamas de ellos y con diferentes capacidades para poder adaptarse a las exigencias de los proyectos. Un inconveniente de ellos es que se necesita un circuito programador que permita el buen funcionamiento de este chip, además de diseñarlo a las necesidades de nuestro proyecto.

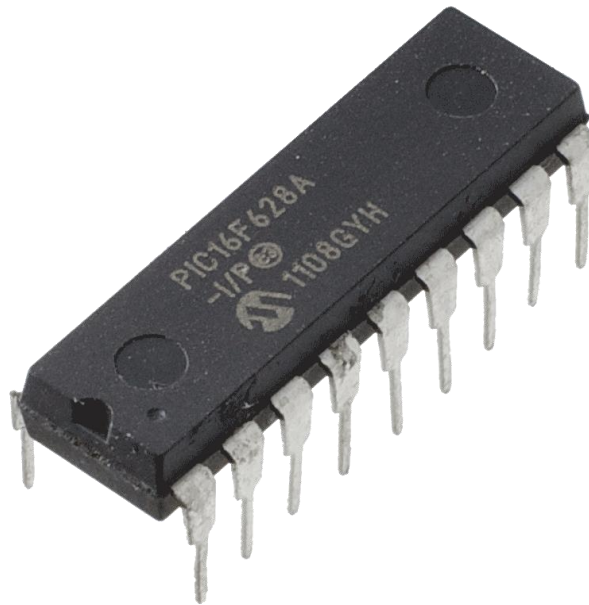


Figura 1.22 PIC16F628A. Figura tomada de <http://www.tandyonline.co.uk/components/semiconductors/microcontrollers/microchip-pic16f628a.html>

Microprocesador.

El microprocesador es un circuito integrado programable, que está conformado por diferentes bloques los cuales tienen una tarea específica. Posee 3 bloques principales: una unidad lógica de procesamiento CPU, una memoria y periféricos de entrada y salida, tal como se muestra en la figura 1.23.

En los microprocesadores, la memoria ROM está diseñada para que no contenga datos iniciales, por lo que es necesario generar un programa o proceso y cargarla en esta memoria; el cual deberá ser escrito en lenguaje ensamblador o algún otro lenguaje capaz de ser ejecutado por el microprocesador.

El microprocesador es un dispositivo igualmente versátil, como el PIC, aunque su costo es más elevado. La desventaja de este es que necesita un circuito que lo alimente para poder funcionar y que se comunique con nuestra PC; de igual manera se necesita un programador ya sea en el mismo circuito a diseñar o en otro, lo cual implica mayor costo.

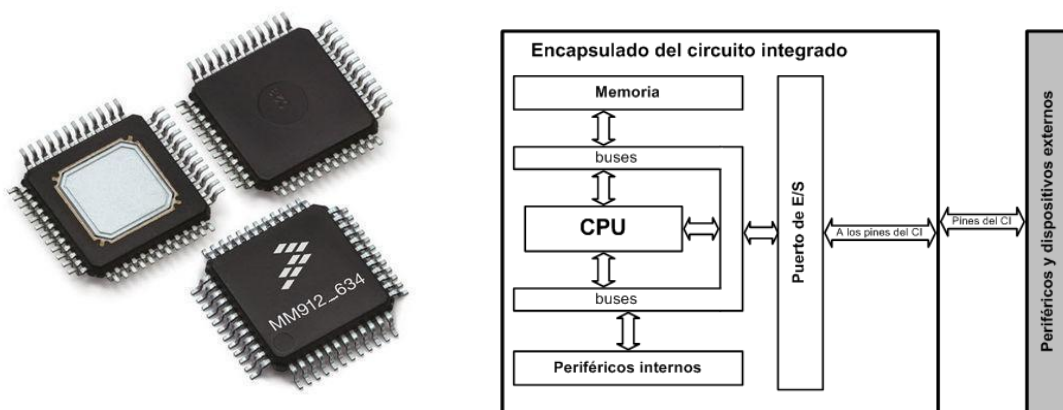


Figura 1.23 Microprocesador. Figuras tomadas de

<http://www.directindustry.es/prod/freescale/microcontroladores-senal-mixta-27531-465879.html>

<http://smrtauste.wikispaces.com/2.+EL+MICROPROCESADOR.+ESTRUCTURA+Y+FUNCIONAMIENTO.+CLASIFICACIONES+DE+LOS+MICROPROCESADORES>

CPLD's y FPGA's.

Los CPLDs y los FPGAs son dispositivos lógicos programables, cuya principal característica es la configuración de su memoria no volátil, poseen una gran densidad de compuertas, la diferencia de estos dos es que los FPGAs poseen una mayor densidad que los primeros.

Los CPLDs poseen varios PLDs de tipo PAL en su interior los cuales permiten implementar ecuaciones lógicas de gran complejidad con una mayor eficiencia gracias a su tipo de arquitectura (arquitectura de matriz de suma de productos y arquitectura look-up table o tabla de datos). La programación de estos es generalmente en lenguajes de descripción de hardware como el VHDL o Verilog.

Los principales fabricantes son Altera con su serie MAX, Atmel con su serie ATF, AMD con su familia Match, Cypress, Lattice y Xilinx.

Estos son otros dispositivos, figura 1.24, que por su diseño nos pueden servir en nuestros propósitos, sin embargo una de las desventajas que se tiene con ellos es que son relativamente costosos y están sobrados para nuestros propósitos.



Figura 1.24 Cyclone V GX Starter Kit (FPGA) MAX V Development Board (CPLD).

Figura tomada de

<http://www.altera.com/products/devkits/altera/kit-terasic-cyclone-v-gx-starter.html>

<http://www.altera.com/products/devkits/altera/kit-max-v.html>

Raspberry Pi.

Raspberry Pi es un dispositivo con los componentes de una placa de computadora (SBC), el cual fue desarrollado en Reino Unido y lanzado el 29 de febrero de 2012 por la fundación Raspberry Pi; en su diseño, como se muestra en la figura 1.25, se engloban diferentes módulos como el procesador ARM11 de 700Mhz, una GPU con memoria, dedicada a video, de 512MB, la cual transmite por su salida HDMI, tiene conectividad a red, además de soportar dispositivos USB, no tiene una unidad de disco duro ya que utiliza una tarjeta SD para ello.

Este dispositivo es un buen candidato para nuestro proyecto, sin embargo, es necesario instalarle un sistema operativo para su correcto funcionamiento, limitándonos en los desarrollos que se tienen.



Figura 1.25 Raspberry Pi. Figura tomada de http://de.wikipedia.org/wiki/Raspberry_Pi

Arduino.

El Arduino es un conjunto de software y hardware de código abierto, cuyo hardware es una placa con entradas y salidas que pueden ser tanto analógicas como digitales. La pieza central del Arduino es el microprocesador ATmega que en cada modelo puede variar al igual que el número de entradas y salidas; su entorno de desarrollo implementa el lenguaje de programación Processing/Wiring el cual diseña un software llamado ARDUINO para poder programar el dispositivo. En la figura 1.26 se puede apreciar los componentes de un Arduino UNO.

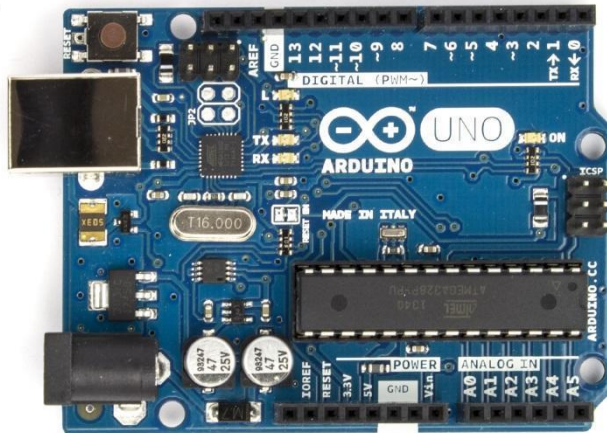


Figura 1.26 Arduino UNO. Figura tomada de <http://arduino.cc/en/Main/arduinoBoardUno>

Este proyecto fue concebido en Italia en el 2005 por David Cuartielles, ingeniero electrónico y docente de la Universidad de Mälmo (Suecia) y Massimo Banzi, italiano, diseñador y desarrollador Web.

El Arduino a utilizar en el proyecto es el Arduino UNO, el cual posee un micro controlador ATmega328, con 14 pines que pueden ser de entrada o salida y 6 de ellos pueden usarse como salidas de PWM, también posee 6 entradas analógicas. El voltaje al cual trabaja el Arduino se encuentra en el rango de los 3.3 Volts hasta los 12 Volts, que pueden suministrarse por la entrada Jack incorporada o conectarse directamente por USB donde se alimentara a si mismo

1.7. ARDUINO.

Arduino puede tomar información del entorno a través de sus pines de entrada, de toda una gama de sensores, y puede afectar aquello que le rodea controlando luces, motores y otros actuadores.

El microcontrolador en la placa Arduino se programa mediante el lenguaje de programación Arduino (basado en Wiring) y el entorno de desarrollo Arduino (basado en Processing). Los proyectos hechos con Arduino pueden ejecutarse sin necesidad de conectar a un ordenador, si bien tienen la posibilidad de hacerlo y comunicar con diferentes

tipos de software (p.ej. Flash, Processing, MaxMSP). En la figura 1.27 se muestra la ventana de programación de Arduino.

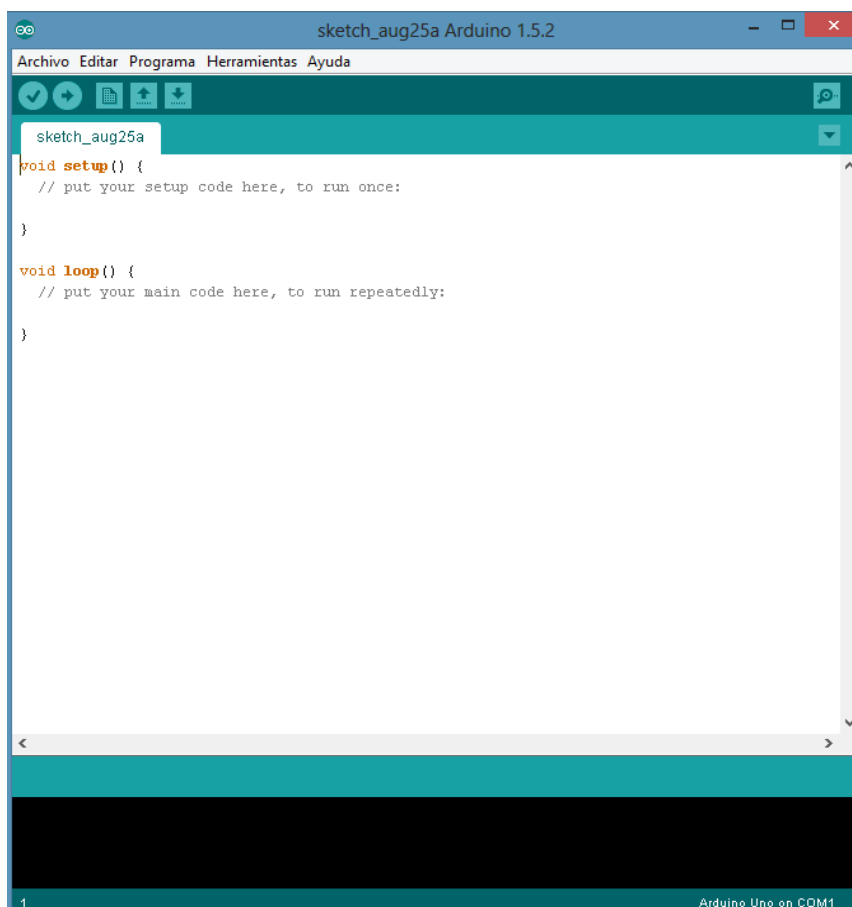


Figura 1.27 Ventana del programador Arduino. Imagen de autoría propia.

Las placas pueden ser hechas a mano o compradas montadas de fábrica; el software puede ser descargado de forma gratuita. Los ficheros de diseño de referencia (CAD) están disponibles bajo una licencia abierta y pueden ser adaptados a diferentes necesidades.

Conceptos básicos de Arduino

Se le llama SKETCH al nombre del programa que se monta al Arduino y que se ejecuta. Este sketch puede contener comentarios llamados Blink los cuales generalmente mencionan lo que realiza el programa y comenta algunos procesos, todos estos comentarios estarán entre /* y */ o si es un comentario de una sola línea podrá llevar simplemente e el comienzo // ya que todo lo que se encuentre entre estos símbolos serán ignorados por el Arduino cuando ejecute el SKETCH.

Al programar podemos utilizar variables las cuales son lugares donde se almacenan datos, estos poseen un nombre el cual puede estar enlazado a un pin de entrada o salida del Arduino, estas variables tienen tipo y valor, generalmente se utilizan para darle un nombre a una variable que se utilizara varias veces en el código y poderla modificar fácilmente.

Existen también funciones las cuales pueden estar escritas en cualquier punto del sketch, estas pueden ser llamadas en cualquier momento del programa pasando los valores que esta necesite para funcionar y posteriormente regresar al código principal al ser llamada. El código o subrutina de estas funciones estarán contenidas entre `{ }`. Existen funciones primordiales dentro del lenguaje de Arduino, uno de ellos es `pinMode` el cual puede configurar un pin del Arduino como una entrada o una salida colocándole como parámetro `input` o `output`

Ej. `pinMode(7, OUTPUT);`

En este ejemplo se le pide al Arduino coloque la salida en el pin 9.

Otra función es `digitalWrite`, esta función puede enviar un valor a un determinado pin, si se le envía `LOW` mandará 0 volts o tierra y si envía `HIGH` mandará 5 Volts.

Ej. `digitalWrite(7, HIGH);`

En esta ocasión indica al Arduino mandar un voltaje alto o 5 volts al pin 7.

`Delay` es otra función muy común, esta permitirá al Arduino esperar determinados milisegundos antes de continuar el proceso.

Ej. `delay(1000);`

Esta línea permitirá que el Arduino espere 1000 milisegundos o 1 segundo.

Existen dos funciones primordiales en un `SKETCH` de Arduino estas son `SETUP()` y `LOOP()`, `SETUP()` es llamado una vez en cuando se ejecuta el sketch y generalmente es

para inicializar todos los pines a utilizar y poder determinar variables, Por otro lado LOOP() es una función recursiva y generalmente es donde se contiene el código principal del sketch.

Arduino permite manejar los siguientes tipos de datos:

- Byte. Almacena un valor numérico de 8 bits. Tienen un rango de 0-255.
- Int. Almacena un valor entero de 16 bits con un rango de 32,767 a -32,768.
- Long. Valor entero almacenado en 32 bits con un rango de 2, 147, 483,647 a -2, 147, 483,648.
- Float. Tipo coma flotante almacenado en 32 bits con un rango de 3.4028235E+38 a -3.4028235E+38.
- Arrays Se trata de una colección de valores que pueden ser accedidos con un número de Índice (el primer valor del índice es 0). Ejemplos de utilización:

- Definición y asignación. `int myArray[] = {value0, value1, value2...}`

- Definición. `int myArray[5]; // declara un array de 6 enteros`

- Asignación del cuarto componente. `myArray[3] = 10;`

- Recuperar el cuarto componente y asignarlo a x. `x = myArray[3];`

Operadores aritméticos

Empleando variables, valores constantes o componentes de un array pueden realizarse operaciones aritméticas y se puede utilizar el operador cast para conversión de tipos.

Ej. `Int a =(int)3.5;` Además pueden hacerse las siguientes asignaciones:

`x ++`. Lo mismo que `x = x + 1`.

`x --`. Lo mismo que `x = x - 1`.

`x += y`. Lo mismo que `x = x + y`.

`x -= y`. Lo mismo que `x = x - y`.

`x *= y`. Lo mismo que `x = x * y`.

`x /= y`. Lo mismo que `x = x / y`.

Para su utilización en sentencias condicionales u otras funciones Arduino permite utilizar los siguientes operadores de comparación:

$x == y$. x es igual a y.

$x != y$. x no es igual a y.

$x < y$, $x > y$, $x <= y$, $x >= y$.

Los siguientes operadores lógicos:

Y lógico: $\text{if}(x > 0 \ \&\& \ x < 5)$. Cierto si las dos expresiones lo son.

O lógico: $\text{if}(x > 0 \ || \ y > 0)$. Cierto si alguna expresión lo es.

NO lógico: $\text{if}(!x > 0)$. Cierto si la expresión es falsa.

El lenguaje de Arduino presenta las siguientes constantes predefinidas:

TRUE / FALSE.

HIGH/LOW. Estas constantes definen los niveles de los pines como HIGH o LOW y son empleados cuando se leen o escriben en las entradas o salidas digitales. HIGH se define como el nivel lógico 1 (ON) o 5 V. LOW es el nivel lógico 0, OFF, o 0 V.

INPUT/OUTPUT. Constantes empleadas con la función `pinMode()` para definir el tipo de un pin digital usado como entrada INPUT o salida OUTPUT.

Sentencias condicionales

El lenguaje de Arduino permite realizar sentencias condicionales `if`, `if... else`, `for`, `while`, `do... while`. Su utilización es similar a las funciones correspondientes en C.

¿Por qué usamos Arduino para este proyecto?

En el ambiente tecnológico, el Arduino es considerado una plataforma accesible a toda la comunidad. A continuación se mencionan algunas razones por las cuales se eligió este controlador en lugar de los antes mencionados.

- Una de las principales razones es que Arduino es de código abierto. Esto significa que los usuarios pueden descargar las actualizaciones de forma gratuita sin la necesidad de adquirir una licencia, la cual, en muchas ocasiones, es muy cara. Tal es el ejemplo de los FPGA de la familia Spartan diseñados por la empresa XILINX o la familia Stratix diseñados por ALTERA o tal vez los microcontroladores de la serie ColdFire diseñados por Freescale, donde algunos de sus módulos tienen un costo extra para ser utilizados, por mencionar algunos casos.
- Su hardware y software se adaptan fácilmente a las necesidades de los usuarios.
- Su programación es fácil de realizar y es muy similar a programar en C. Además para aquellas personas que se les dificulte su uso, en la página de internet de Arduino se presentan ejemplos para comprender mejor el lenguaje de programación. La programación de PIC's y microcontroladores es más compleja, realizándose en varios lenguajes y programas, donde el código más eficiente se diseña en ensamblador siendo este más complicado.
- No se requiere de un circuito programador para hacer funcionar el Arduino. Este se conecta a la computadora y una vez finalizado el programa se compila y se "carga" al Arduino, caso contrario a los PIC y los microprocesadores, ya que los primeros requieren de un circuito programable y un circuito secundario para su funcionamiento, en cuanto a los CPLD's Y FPGA's es muy parecida su programación pero salen sobrados para nuestro diseño.
- Arduino se ha vuelto muy popular en el ramo tecnológico ya que es usado por emprendedores individuales hasta grandes empresas. El Raspberry Pi también es muy conocido, sin embargo, sus características sobrepasan nuestras necesidades al igual que su precio.
- En promedio las placas de circuitos son caras, por lo que en las universidades se compran pocas y estas tienen que ser compartidas entre numerosos grupos de

personas. Dando como resultado el aprendizaje de pocos y el desinterés de muchos. La finalidad es que las escuelas puedan comprar más dispositivos como estos para que el aprendizaje sea equitativo.

- Y como en muchas ocasiones, el estudiante tiene que ampliar su conocimiento fuera del aula, este puede comprarse su propio Arduino ya que es económico (comparado con otros controladores) y su precio aproximado anda rondando los \$400.00, precio similar a lo que costaría un libro de texto o una calculadora científica. Y si comparáramos el precio de una Raspberry Pi con un Arduino, el Raspberry Pi cuesta más del doble de lo que cuesta una placa de Arduino UNO.

Así que como pudimos ver, el Arduino es un microcontrolador muy accesible en todos los sentidos ya que cada usuario lo puede modificar a las necesidades de su proyecto y lo mejor, sin gastar mucho dinero. Lo cual apoya bastante, económicamente hablando, para aquellos estudiantes que quieran ampliar sus conocimientos y realizar futuros desarrollos.

CAPÍTULO 2

DISEÑO DE FRAMES MULTITOUCH

2.1. PROPUESTA.

EL modelo a seguir para nuestro marco es del tipo plano infrarrojo, su funcionamiento se basa en la interferencia que tiene un objeto con un haz infrarrojo procedente de un láser, este haz es producido por un lente de apertura a 120° el cual expandirá un haz infrarrojo a un plano que cubrirá la pantalla de nuestro monitor, en el contorno de la pantalla existirán fotodiodos que captarán la intensidad de luz que reciban, al existir una variación estos diodos la captaran y en consecuencia variará el voltaje de salida. Estas variaciones estarán registradas y transmitidas por los multiplexores.

Se propone construir un dispositivo que sea capaz de conectarse a otros dispositivos de su mismo tipo para su expansión siempre y cuando genere un área cuadrada para la proyección a manipular. Estos dispositivos deberán de contener el circuito básico para la recepción del láser infrarrojo y el medio de transmisión hacia el Arduino.

El Arduino será un elemento externo al cual llegarán todas las conexiones de los dispositivos anteriores, por el momento se utilizará de manera externa con lo que nos permitirá una mayor manipulación de este, posteriormente se puede reemplazar por un modelo más pequeño ya sea el Arduino mini o el Arduino nano, los cuales podrán incorporarse en alguno de los módulos ya que sus dimensiones son pequeñas y su funcionalidad es la misma

2.2. MEJORAS DEL MARCO.

Las mejoras que se le harán a nuestro marco en comparación a los que existen en el mercado actualmente le darán mayor capacidad de área de expansión con un costo menor, sin necesidad de adquirir un equipo nuevo.

La capacidad de expandirse se lograra al construir un modelo básico para cada segmento el cual será capaz de acoplarse a otro segmento que se añada posteriormente.

Todos los materiales que se utilizarán para este proyecto serán de fácil acceso y comúnmente usados en el ramo de la electrónica por lo que nos permitirá desarrollar más el proyecto y poder expandirlo sin un costo mayor.

2.3. VENTAJAS.

Las ventajas que se pueden percibir es el uso del Arduino como medio de comunicación, como es un dispositivo de código libre y sin restricciones dará lugar a que diferentes programadores puedan hacer uso de este recurso y puedan mejorarlo posteriormente.

El diseño que se pretende implementar permitirá que el mismo dispositivo pueda expandirse para lograr una mayor área de interacción, cosa contraria a los equipos actuales que se obtienen de un solo tamaño y se necesita otro dispositivo si es que se quiere un área mayor.

Se pretende por otro lado que los elementos que conformaran el circuito sean de fácil adquisición y no sean elementos especializados que genere un coste mayor ya que el principal objetivo de construir este diseño es que sea accesible para diferentes áreas.

2.4. DEFICIENCIAS.

Una desventaja que se tiene, es el uso de sensores infrarrojos en los cuales si existe una interferencia de longitud de onda igual al rango que manejan estos sensores, podría dar falsos puntos. El uso de sensores de un rango menor, los cuales reducen la interferencia ambiental y logran tener una mejor recepción, mejoran pero no eliminan al 100% el problema.

Si el área de interacción es muy grande es posible que el puerto USB no pueda alimentar a todo el circuito por medio del Arduino así que posiblemente podría utilizarse otra entrada USB o una fuente externa para su correcto funcionamiento.

2.5. ESTRUCTURA FÍSICA.

El marco estará conformado por módulos más pequeños que se acoplarán para poder expandir el área de cobertura. Estos módulos son de aproximadamente 70 mm de largo por 20 mm de ancho con una profundidad no mayor a 10mm y son los que alojarán los fotodiodos y estarán ubicados en el contorno del área de proyección.

Para este proyecto se necesitarán 2 módulos de control y cada uno podrá administrar a 8 módulos de recepción aunque en nuestro caso solo serán 7 ya que se mantendrá una relación aproximada de 16:9 lograda con la colocación de 14 módulos.

Ambos módulos, el de recepción y el de control, serán diseñados en placa fenólica de doble capa para optimizar el área de conducción y no colocarle componentes extras como puentes o cables, los diseños fueron creados en el programa Livewire y PCB Wizard de New Wave Concepts, cada modelo se describirá a continuación.

El modelo de recepción se diseña inicialmente en livewire donde es posible crear un diagrama esquemático con las conexiones necesarias para la realización del circuito, el siguiente diagrama de la figura 2.1, representa cada uno de los componentes que incorporara nuestro módulo. El programa incorpora componentes genéricos los cuales se pueden acoplar a los circuitos integrados según el número de pines que contenga, ya que el programa no contiene todos los circuitos integrados del mercado, también podemos incorporar elementos que son básicos en diagramas.

En este diagrama usaremos un modelo de circuito integrado de 16 pines el cual nos ayudará a representar el multiplexor 8 a 1 que utilizaremos y ayudándonos de la hoja de especificaciones del fabricante realizaremos las conexiones necesarias. En cada una de las salidas y/o entradas de nuestro circuito se conectará una resistencia de 10 kohm y un fotodiodo desde Vcc a tierra respectivamente, los pines de alimentación y de control están conectados a una terminal de 5 pines los cuales nos ayudarán a enlazar cada uno de los módulos ya que comparten los mismos pines, caso contrario con el pin común o salida ya que esta se conectara por separado.

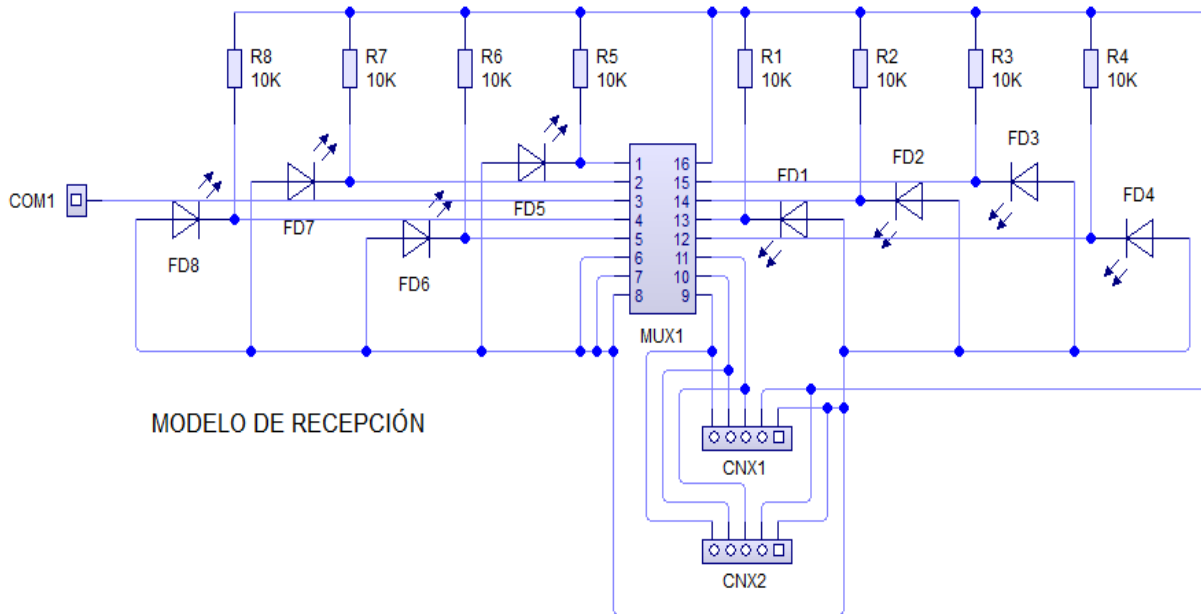


Figura 2.1 Diagrama esquemático de recepción. Imagen de autoría propia

Una vez realizado el diagrama en Livewire podemos convertirlo a un circuito impreso con una función propia del programa, pero debido a que el acomodo propuesto no cumplió con nuestras necesidades, nos dimos a la tarea de modificarlo, quedando un acomodo como se muestra en la primer imagen de la figura 2.2 (de izquierda a derecha). Donde se observa que los fotodiodos están en línea quedando en la orilla de nuestro circuito, los conectores se colocan a los lados para acoplar cada uno de los módulos y las resistencias se colocan de manera vertical para disminuir el espacio a utilizar. El diseño del circuito se realiza de tal forma que se usen ambas caras de la placa fenólica con la finalidad de no usar cables extras para la unión de componentes.

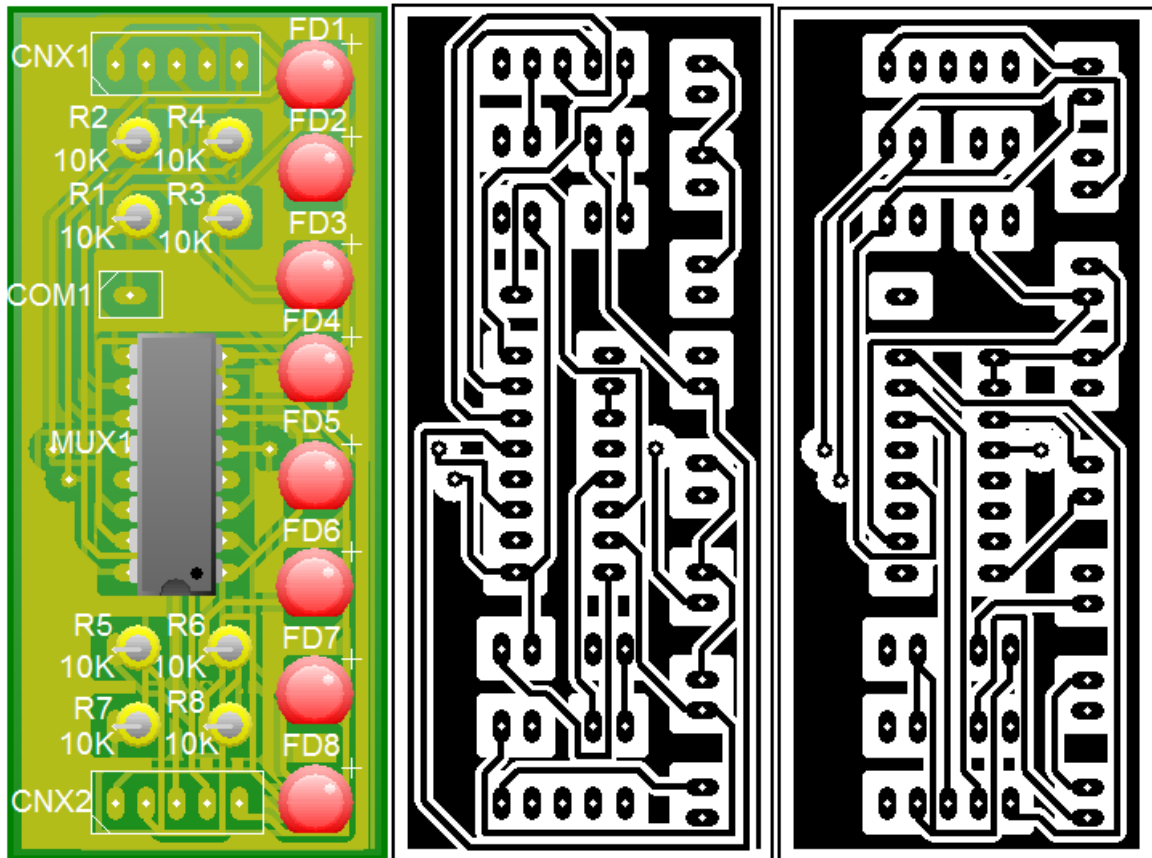


Figura 2.2 Modelo final y circuitos impresos para el módulo de recepción. Imagen de autoría propia.

La primera imagen de la figura 2.2 (observando de izquierda a derecha), muestra el diseño final con los componentes instalados, la segunda imagen es el diseño del circuito impreso del lado contrario de los componentes y la tercera imagen es el diseño del circuito impreso del mismo lado de los componentes.

El modelo de control es básicamente el mismo que el de los receptores solo que en este no intervienen los fotodiodos ni las resistencias, solamente se colocará una terminal de 8 salidas al multiplexor para la conexión de los pines comunes de los módulos de recepción. También contará con otros dos módulos de conexión: uno de control y otro de alimentación y por último un pin de salida que será el que se conecte al Arduino, teniendo como resultado el diagrama de conexiones que se muestra en la figura 2.3.

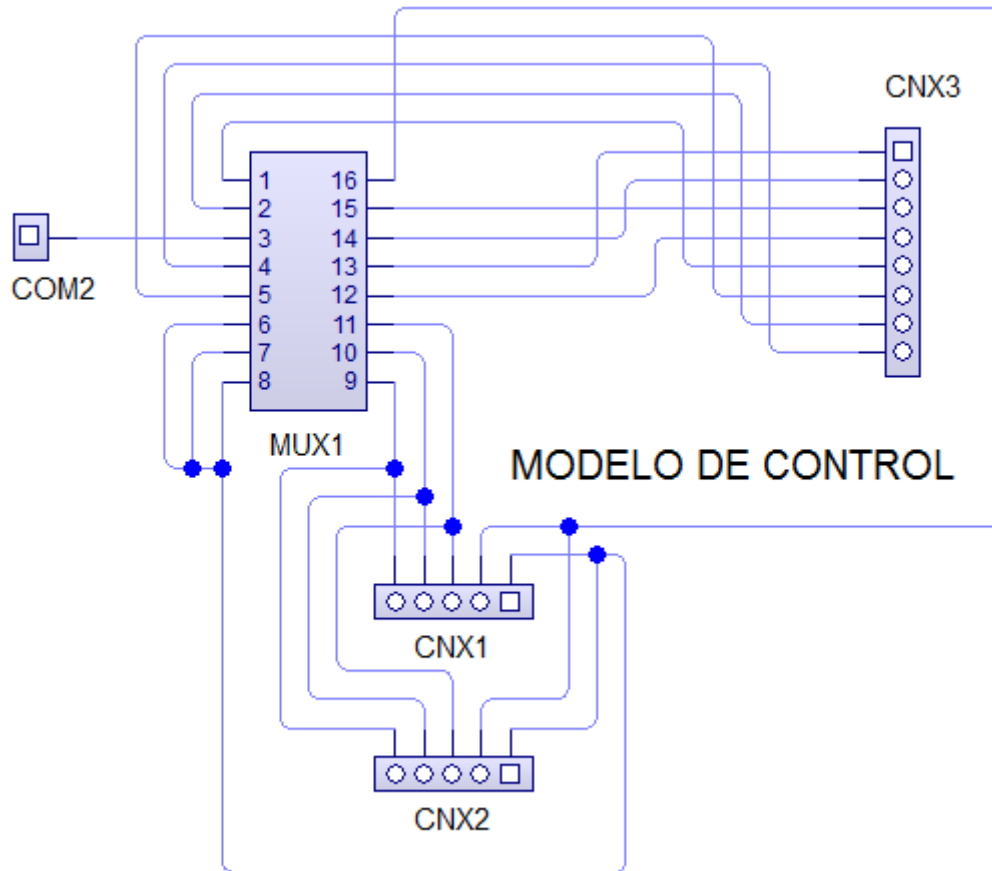


Figura 2.3 Diagrama esquemático de control. Imagen de autoría propia

Una vez terminado el diagrama procedemos a realizar el circuito impreso con la misma herramienta antes mencionada y reacomodando los componentes para lograr un mejor aprovechamiento de la placa. En la primer imagen de la figura 2.4 (observando de izquierda a derecha), las conexiones del multiplexor se encuentran en un lado del circuito y las conexiones de alimentación y de control se encuentran en los laterales para poder expandir el circuito en un futuro y finalmente nuestro pin común lo colocamos en una esquina de nuestro circuito para tener fácil acceso a él y poder conectarlo a nuestro Arduino.

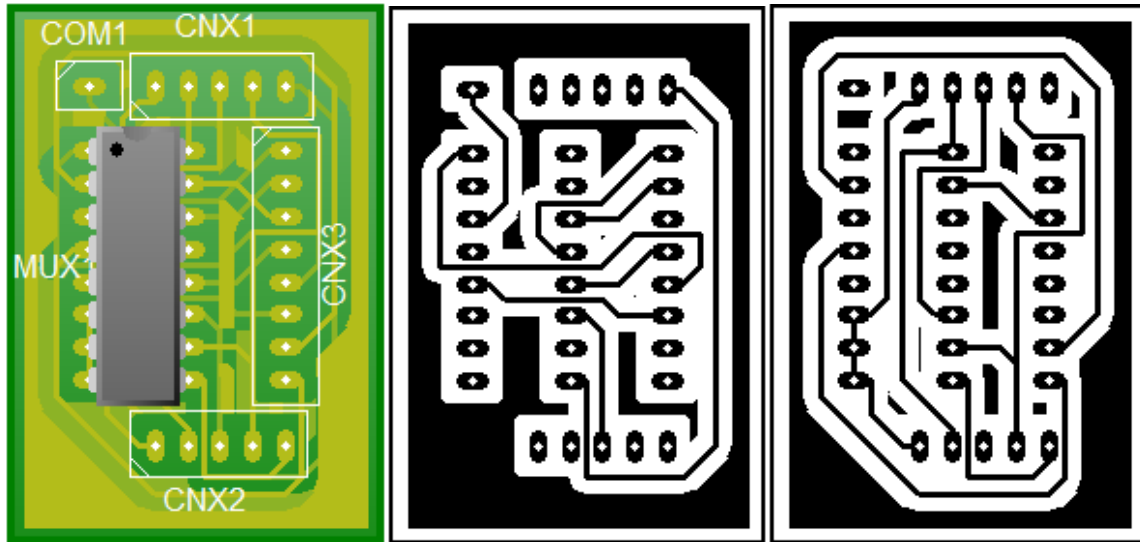


Figura 2.4 Modelo final y circuitos impresos para el módulo de control. Imagen de autoría propia

La primera imagen de la figura 2.4 (de izquierda a derecha), es el diseño final con los componentes instalados, la segunda imagen es el diseño del circuito impreso del lado contrario de los componentes y la tercera imagen es el diseño del circuito impreso del mismo lado de los componentes.

2.6. MATERIALES.

Para este diseño se necesitarán 16 multiplexores analógicos (HCF4051BEY), 110 fototransistores (PT331C), 110 resistencias de 10Kohm, 2 transistores NPN a 50 V (BC337-25), 2 láseres infrarrojos de 35mW con una longitud de onda de 980nm, 2 lentes generadores de línea de 120° y un dispositivo Arduino.

El multiplexor a utilizar es el HCF4051BEY de STMicroelectronics, este circuito integrado ofrece una baja resistencia y una mínima corriente de fuga, tiene la capacidad de trabajar analógica o digitalmente, ya que tiene un amplio rango de funcionamiento y de operación. El voltaje al cual trabaja se encuentra en el rango de -0.5 Volts hasta los 22 Volts. Este multiplexor tiene 8 canales de entrada y uno de salida, con una entrada de inhibición y tres

pinos de control binario. Este circuito integrado ofrece un bajo consumo de energía cuando está en reposo y una mínima interferencia de ruido en cada uno de sus canales.

Las resistencias serán de 10kohm y serán de ¼ de watt. Para un diseño posterior las resistencias podrán ser de montaje superficial.

Se utilizarán transistores BC337-25, los cuales permitirán la activación de los láseres. Estos transistores son del tipo NPN, y de un empaquetado TO-92, el voltaje máximo que permite entre el colector y la base es de 50V el voltaje entre el colector y el emisor es de 45V y el de emisor-base es de 5V, la corriente del colector es del orden de los 500mA.

Se usaran 2 láseres infrarrojos de 35mW con una longitud de onda de 980nm, a los cuales se les instalara 1 lente generadora de línea de 120° a cada uno y así crear un plano infrarrojo. Y para la detección del plano infrarrojo usaremos fototransistores PT331C con diámetro de 5mm, los cuales tienen una rápida respuesta en el tiempo y una alta sensibilidad a la luz normal e infrarroja.

2.7. CIRCUITO ELÉCTRICO.

El circuito eléctrico contará con diferentes módulos para su mejor diseño. El primer módulo contendrá los fotodiodos que fungirán como los receptores del plano infrarrojo. Estos estarán conectados a unos multiplexores los cuales transmitirán la detección de un toque dentro del área de proyección. Cada uno de los fotodiodos tendrá en el cátodo una resistencia, del orden de kohms, conectada a 5 volts, la cual ayudará a no generar un corto circuito; por consiguiente el ánodo estará conectado directamente a tierra común de todo el circuito, tal como se muestra en la figura 2.5.

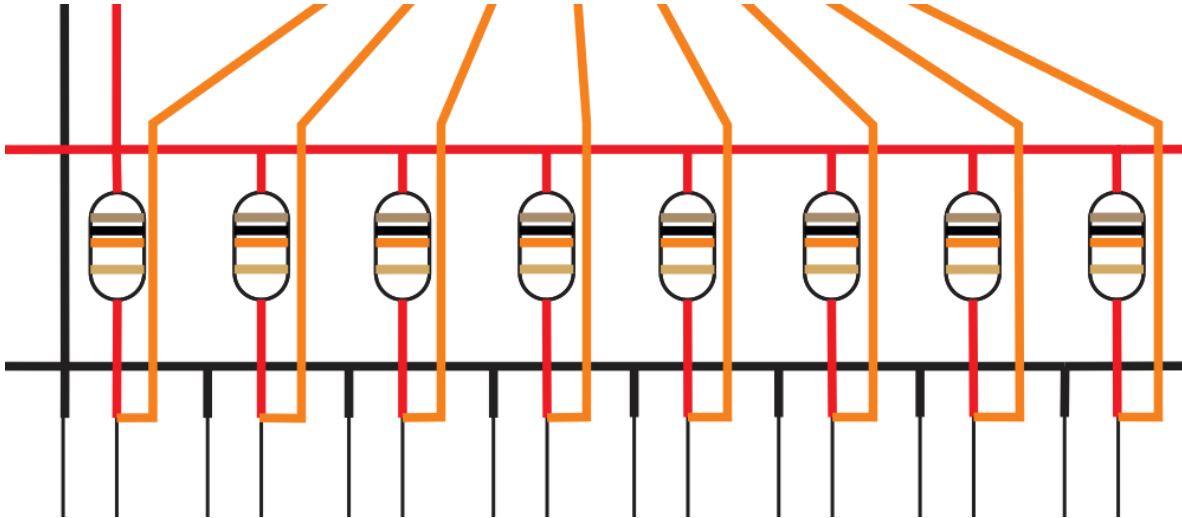


Figura 2.5 Conexión de las resistencias en el circuito. Imagen de autoría propia

El segundo módulo constará de los multiplexores principales, de 8 a 1 entradas/salidas, que estarán alimentados igualmente a la fuente de 5 volts, cada una de sus entradas corresponderá a cada fotodiodo, las entradas de control de cada uno de estos multiplexores estarán conectadas en común a las salidas de 3 puertos del Arduino. El pin común estará conectado a otro multiplexor que hará el censado de los multiplexores principales para que puedan abarcar toda el área de toque sin que se traslapen las señales, lo anterior se puede apreciar en la figura 2.6.

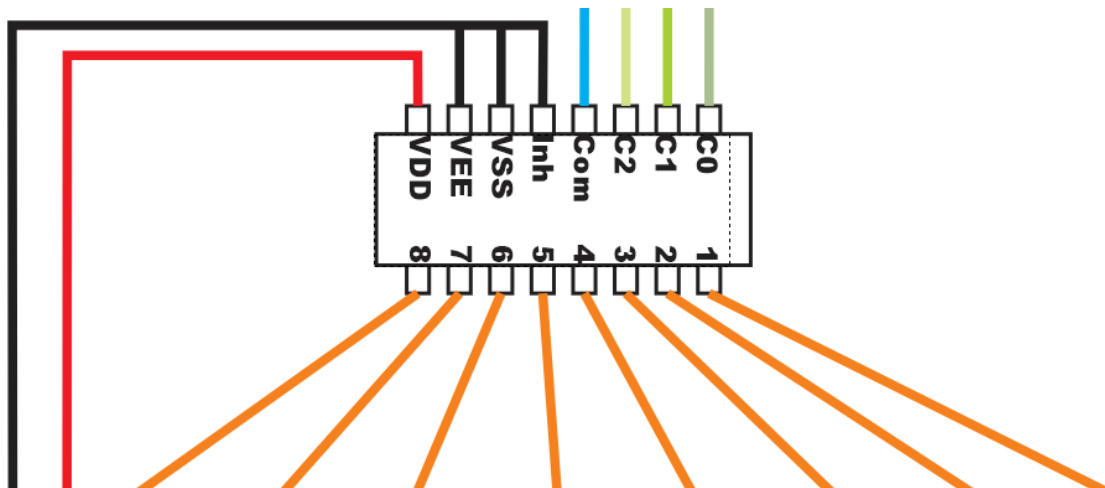


Figura 2.6 Conexión de los multiplexores a los fotodiodos. Imagen de autoría propia

El tercer módulo constará de dos multiplexores secundarios que son los que dará corrimiento de censado a los multiplexores principales. Cada uno de los pines de los multiplexores principales estarán conectados a las salidas de estos obteniendo como resultado que se habilite solo uno de los multiplexores principales y así determinar de manera más precisa, la coordenada de toque. Los pines de control de estos multiplexores secundarios estarán conectados entre sí y a su vez a otros tres pines del Arduino. En esta ocasión los pines comunes a estos multiplexores estarán conectados directamente al Arduino el cual, según la programación, habilitará cada uno permitiendo captar si se realizó un toque dentro del área lateral derecho o lateral izquierdo. Las conexiones del multiplexor de control se pueden apreciar en la figura 2.7. De igual manera la alimentación de estos multiplexores será de 5 volts en común con todo el circuito.

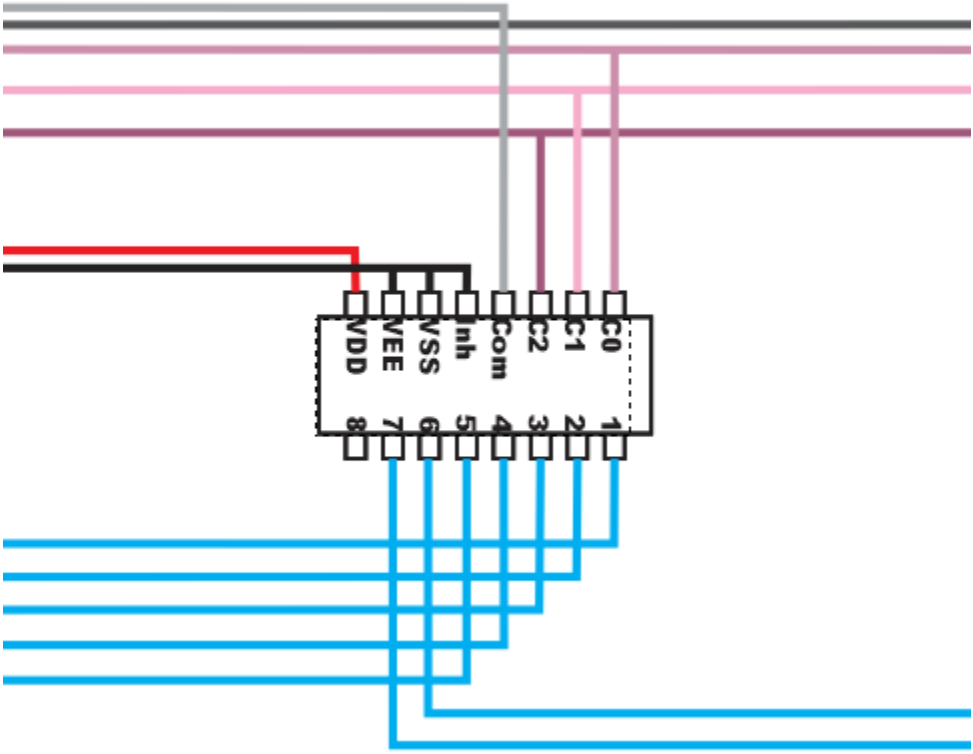


Figura 2.7 Conexión de los multiplexores de control. Imagen de autoría propia

El cuarto módulo tendrá 2 botones de control: uno que mandara la señal al Arduino para comenzar y el otro que mandara a encender los láseres para su alineación correcta. Estos botones serán dos switch (push bottom) que estarán conectados a tierra y al momento de presionarlos le mandaran la señal al Arduino para comenzar o alinear láseres, según sea el caso ; también existirán dos transistores que serán habilitados por medio del Arduino y son los que le darán potencia a los dos láseres que utilizaremos en esquinas inferiores de nuestro marco, estos transistores en la base tendrán una resistencia de 1 kohm conectada a un pin del Arduino cada uno, el colector estará conectado a tierra y el emisor estará colocado directamente a la parte negativa de los láseres.

Por último, el quinto módulo constará del Arduino Uno, el cual estará conectado por USB a una PC y deberá alimentar a todo el circuito eléctrico. En la figura 2.8 se muestran las conexiones hacia el Arduino y en la figura 2.9 se muestra el esquema final de cómo quedaría el marco multitáctil con todos los módulos unidos.

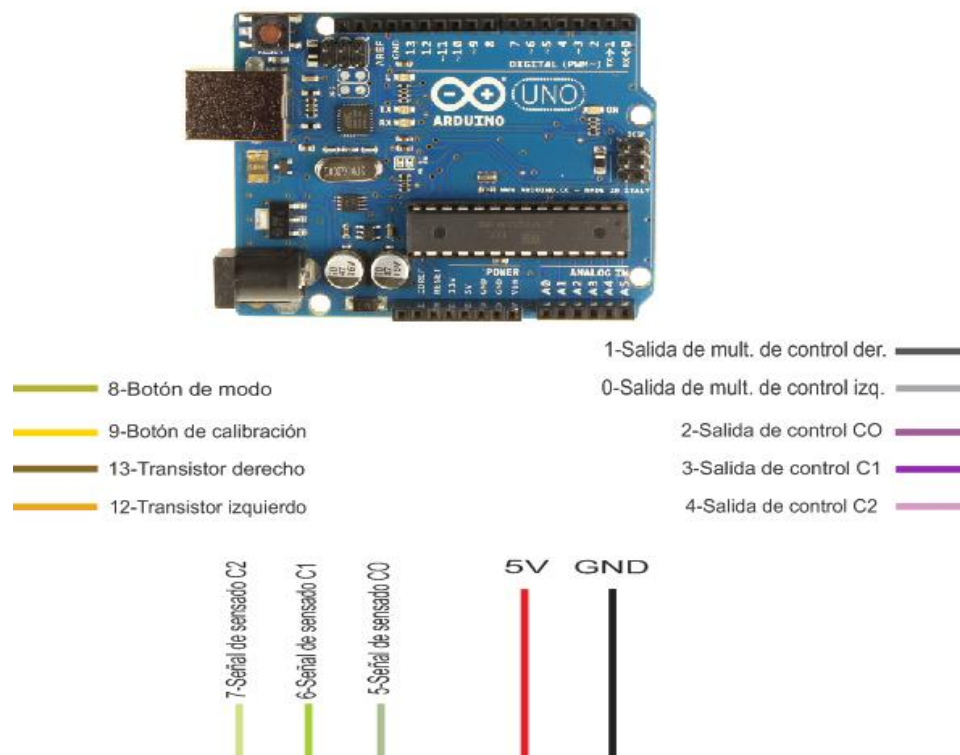


Figura 2.8 Conexiones del marco multitáctil al Arduino. Figura tomada de <http://arduino.cc/en/Main/arduinoBoardUno> e Imagen de auditoría propia.

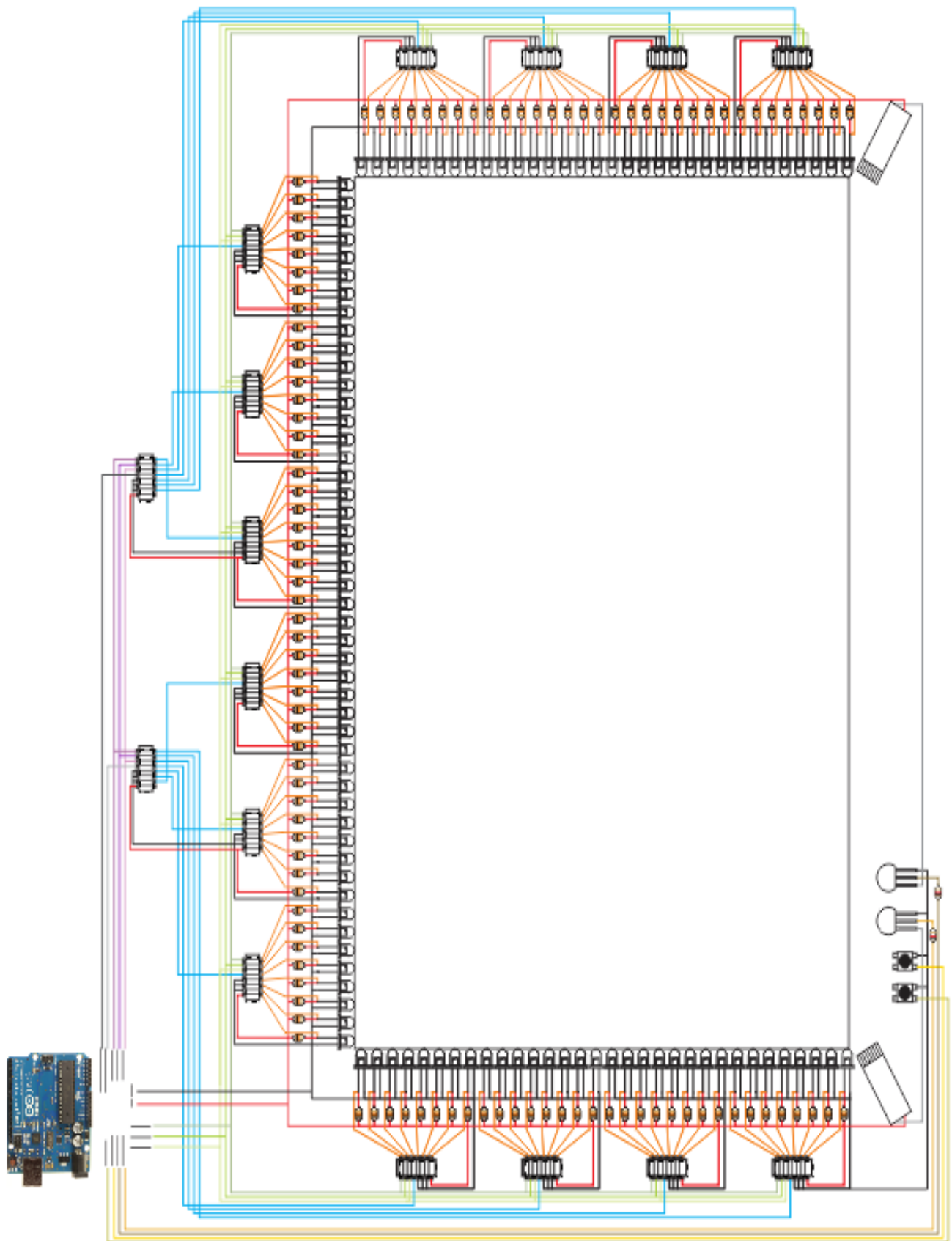


Figura 2.9 Esquema del marco multitáctil. Imagen de auditoría propia.

2.8. PROGRAMACIÓN.

El lenguaje de programación y el entorno de desarrollo del Arduino está basado en Processing, es posible poder programarlo en otros lenguajes mediante software que sirvan de traductores como: Minibloq, que mediante iconos puedes crear la programación y ver el código en C++; Modkit que permite programar otros dispositivos como Seeduino y LillyPad en un estilo de bloques, entre otros.

Es posible utilizar el Arduino para desarrollar aplicaciones interactivas, ya que puede comunicarse de manera serie con la pc, algunas aplicaciones capaces de manejar los datos del puerto serie y por lo tanto del Arduino son Java, Flash, Python, Matlab, Visual Basic, entre otros.

El software a utilizar para la programación de nuestro Arduino será “Arduino 1.0.5 R2” que es propio de la empresa y basado en Processing; este software utiliza un lenguaje de programación muy parecido a C++.

El dispositivo de control, en este caso el Arduino, es el que controlará cada uno de los multiplexores, y estos a su vez le enviarán las variaciones de voltaje de los fotodiodos. El Arduino los interpretará e interactuará con la PC y mandará una coordenada de ubicación dentro del espacio de proyección.

A continuación, se mostrara el código con el que se pretende que nuestro marco pueda interpretar los puntos de toque que ocurran. Este código está diseñado a base de funciones, las cuales tienen un proceso específico y serán funciones claves en la determinación del punto de toque.

La primera parte del código es donde se definen cada uno de los puertos del Arduino que se utilizarán, dándoles un nombre en especial para poder ubicarlos y recordarlos de manera sencilla. También se crean arreglos de datos para llamarlos fácilmente al momento de programar y variables que se utilizarán durante el programa.

```

/*PROGRAMA PARA ARDUINO.
CÓDIGO PARA EL CONTROL DE MULTIPLEXORES ANALÓGICO PARA LA CAPTURA DE SOMB
RAS INFRARROJAS
Y DETECCIÓN DE COORDENADAS EN UN DISEÑO DE FRAME MULTITOUCH */

#define AnalogicoUno 0 //SUSTITUYE AnalogicoUno
EN EL CODIGO PARA INDICAR EL PIN 0 ANALOGICO DEL ARDUINO
#define AnalogicoDos 1 //SUSTITUYE AnalogicoDos
EN EL CODIGO PARA INDICAR EL PIN 1 ANALOGICO DEL ARDUINO

int Pc0 = 2; //PINES DE CONTROL PARA
LOS MODULOS DE CONTROL
int Pc1 = 3;
int Pc2 = 4;
int Ps0 = 5; //PINES DE CONTROL PARA
MODULOS DE SENSORES
int Ps1 = 6;
int Ps2 = 7;
int Cali = 9; //PIN DE CALIBRACION
int LaserI = 12; //LASER IZQUIERDO
int LaserD = 13; //LASER DERECHO

int DirSens[] = {
    000, 001, 010, 011, 100, 101, 110, 111}; //DIRECCIONDE DE CONTROL
PRIMARIO (8)
int DirCont[] = {
    000, 001, 010, 011, 100, 101, 110}; //DIRECCIONES DE CONTROL
SECUNDARIO (7)

int SensorNum; //CONTADOR DEL NUMERO DE
SENSOR
int UmbralLasI[111]; //VALORES PARA EL UMBROL
DE SOMBRA DEL LASER IZQUIERDO
int UmbralLasD[79]; //VALORES PARA EL UMBROL
DE SOMBRA DEL LASER DERECHO
int ValorSombraI = NULL; //VALOR NULO PARA LA
SOMBRA LASER IZQUIERDA
int ValorSombraD = NULL; //VALOR NULO PARA LA
SOMBRA LASER DERECHA

float InicioDeX = NULL; //INICIO DEL TOQUE EN X
float InicioDeY = NULL; //INICIO DEL TOQUE EN Y
float FinDeX = NULL; //FIN DEL TOQUE EN X
float FinDeY = NULL; //FIN DEL TOQUE EN Y

int BanMod = 0; //0=NORMAL, 1=CALIBRACION
DE LOS LASERES
int LasNum = 0; //0=IZQUIERDA , 1=DERECHA

```

La función principal se llamada “setup”, la cual se ejecutará una sola vez al encender el Arduino, por lo tanto los procesos fundamentales y funciones iniciales se ejecutan aquí. La función setup será la encargada de inicializar el modo serial a 9600 baudios, que es la

velocidad estándar de comunicación, para crear la interacción que necesitamos con el ordenador. También se inicializan cada uno de los puertos que vayamos a utilizar en el Arduino, ya sea como un tipo de salida o entrada y poderla inicializar con algún valor. Y finalmente se realizan las funciones primordiales para la ejecución de algún programa.

```
void setup() {
  Serial.begin(9600);

  pinMode(Pc0, OUTPUT); //INICIALIZACION DE LOS
  PINES DE CONTROL SECUNDARIOS
  pinMode(Pc1, OUTPUT);
  pinMode(Pc2, OUTPUT);
  pinMode(Ps0, OUTPUT); //INICIALIZACION DE LOS PINES
  DE CONTROL PRIMARIOS
  pinMode(Ps1, OUTPUT);
  pinMode(Ps2, OUTPUT);
  pinMode(LaserD, OUTPUT); //INICIALIZACION D ELOS
  LASERES
  pinMode(LaserI, OUTPUT);
  pinMode(Cali, INPUT); //ENTRADA DE BOTON DE
  CALIBRACION

  ValorautoS('I'); //PRIMERA INICIALIZACION DE
  LA SOMBRA DEL LASER IZQUIERDO
  ValorautoS('D'); //PRIMERA INICIALIZACION DE
  LA SOMBRA DEL LASER DERECHO
}
```

La siguiente función es “loop”, esta función es esencial para nuestros programas ya que es un ciclo que se repetirá cuando termine el código contenido en él. Es un ciclo infinito que aprovecharemos para detectar cada uno de los puntos de toque. Esta parte del código es la que llamara a todas las funciones encargadas de la detección de los toques y verificara si el modo de calibración esta en operación.

```
void loop() {
  if(digitalRead(Cali) == LOW) { //VERIFICACION DEL
  MODO DE CALIBRACION //1=CALIBRACION
    BanMod = 0;
  }
  else{
    BanMod = 1;
  }
  if(BanMod == 1) { //INICIO MODO NORMAL
    Escaneo('I'); //ESCANEOS DE SOMBRAS
    Escaneo('D');
  }
}
```

```

        if((ValorSombraI != NULL) && (ValorSombraD != NULL)){ //SI EXISTE
SOMBRAS
        Coordenadas(); //BUSCA LAS
COORDENADAS XY

        if((InicioDeX != NULL) && (FinDeX != NULL)){ //SI LAS
COORDENADAS SON DIFERENTES DE 0
        if((InicioDeX != FinDeX) && (InicioDeY != FinDeY)){ //SI EL
INICIO FUE DIFERENTE AL FINAL
        MueveCursor(); //MUEVE
EL CURSOR
        delay(20); ////ESPERA
20 MILISEGUNDOS
        }
        }
        }
        else{ //SI NO
EXISTEN SOMBRAS
        InicioDeX = NULL; //COLOCA LOS
ESTADOS INICIALES NULOS
        InicioDeY = NULL;
        FinDeX = NULL; //COLOCA LOS
ESTADOS FINALES NULOS
        FinDeY = NULL;
        }
        }
        else{ //MODO DE
CALIBRACION DE LASERES
        CalibracionLaser(); //CALIBRAC
ION DE LASERES
        }
        delay(20); //ESPERA 20
MILISEGUNDOS
    }
}

```

La función “ValorautoS”, se ejecutará una sola vez por cada láser al encender el Arduino. La función principal es obtener el valor máximo de voltaje detectado en cada uno de los fotodiodos receptores, es decir, el voltaje de la entrada analógica de Arduino proveniente de los multiplexores y el receptor led y se guardará en un arreglo. Esta comparación se obtendrá del láser derecho e izquierdo y para cada uno tendremos un valor inicial del fotodiodo. El láser izquierdo alumbrará del fotodiodo 32 al 111, por lo tanto su valor inicial será 32 y el láser derecho alumbrará del fotodiodo 0 al 80 por lo tanto su valor inicial será 0.

```

void ValorautoS(char laser){
    if(laser == 'I'){ //LASER IZQUIERDO
        digitalWrite(LaserI, HIGH); //ENCIENDE EL LASER
IZQUIERDO
    }
}

```

```

    SensorNum = 32; //COLOCA COMO
CONTADOR INICIAL 32
}
else{ //CASO CONTRARIO EL LASER
DERECHO
    digitalWrite(LaserD, HIGH); //ENCIENDE EL LASER
DERECHO
    SensorNum = 0; //COLOCA EL CONTADOR
INICIAL EN 0
}

//===== PARTE IZQUIERDA
for(int a=0; a<7; a++){ //INICIA EL CICLO DE LOS
MODULOS DE CONTROL
    if((laser == 'I') && (a <= 3)){ //SE OMITE EL CICLO
MIENTRAS SE CUMPLA LA CONDICION
        continue; //SE SALTA LOS COMANDOS
Y SIGUE CON LA SIGUINETE INTERACCION
    }
    int FilaCont = DirCont[a]; //ASIGNA LA DIRECCION
DE CONTROL A VARIABLES SEPARADAS
    int p0 = FilaCont & 0x01; //GUARDA EL PRIMER
VALOR
    int p1 = (FilaCont>>1) & 0x01; //GUARDA EL SEGUNDO
VALOR CON UN CORRIMIENTO
    int p2 = (FilaCont>>2) & 0x01; //GUARDA EL TERCER
VALOR CON OTRO CORRIMINETO
    digitalWrite(Pc0, p0); //ASIGNA EL PRIMER VALOR
AL PIN DE CONTROL 0
    digitalWrite(Pc1, p1); //ASIGNA EL SEGUNDO
VALOR AL PIN DE CONTROL 1
    digitalWrite(Pc2, p2); //ASIGNA EL TERCER VALOR
AL PIN DE CONTROL 2

    for(int s=0; s<8; s++){ //INICIA EL CICLO DE LOS
MODULOS DE SENSADO
        int FilaSens = DirSens[s]; //ASIGNA LA DIRECCION
DE CONTROL DE LOS SENSORES A VARIABLES SEPARADAS
        int s0 = FilaSens & 0x01; //GUARDA EL PRIMER
VALOR
        int s1 = (FilaSens>>1) & 0x01; //GUARDA EL SEGUNDO
VALOR CON UN CORRIMIENTO
        int s2 = (FilaSens>>2) & 0x01; //GUARDA EL TERCER
VALOR CON OTRO CORRIMINETO
        digitalWrite(Ps0, s0); //ASIGNA EL PRIMER
VALOR AL PIN DE CONTROL DEL SENSOR 0
        digitalWrite(Ps1, s1); //ASIGNA EL SEGUNDO
VALOR AL PIN DE CONTROL DEL SENSOR 1
        digitalWrite(Ps2, s2); //ASIGNA EL TERCER
VALOR AL PIN DE CONTROL DEL SENSOR 2

        int tMax = 0; //INICIALIZA UNA
VARIABLE TEMPORAL EN 0
        delay(3); //RETARDO DE 3 MS
        int tVal = analogRead(AnalogicoUno); //INICIALIZA UNA
VARIABLE TEMPORAL CON EL VALOR ANALOGICO DEL PIN 0

```

```

        tMax = max(tVal,tMax); //GUARDA EL VALOR
MAXIMO ENTRE EL VALOR LEIDO Y EL ANTERIOR
        if(laser == 'I'){ //VERIFICA SI LO QUE
ESTE A CONTINUACION SERA PARA EL LADO IZQUIERDO
            UmbralLasI[SensorNum] = CalculoDeVentana(tMax); //GUARDA EL VALOR
DEL CALCULO DE LA SOMBRA SEGUN EL VALOR GUARDADO
        }
        else{ //SI NO ES PARA EL
LADO IZQUIERDO SERA PARA EL DERECHO
            UmbralLasD[SensorNum] = CalculoDeVentana(tMax); //GUARDA EL
VALOR DEL CALCULO DE LA SOMBRA SEGUN EL VALOR GUARDADO
        }

        SensorNum++; // AUMENTA EN 1
EL CONTADOR
    }
}
//===== PARTE IZQUIERDA TERMINA

//===== PARTE DERECHA
for(int b=0; b<7; b++){ //INICIA EL CICLO DE
LOS MODULOS DE CONTROL
    if((laser == 'D') && (b >= 3)){ //SE OMITE EL CICLO
MIENTRAS SE CUMPLA LA CONDICION
        break; //SE SALTA LOS
COMANDOS Y TERMINA EL CICLO
    }
    int FilaCont = DirCont[b]; //ASIGNA LA DIRECCION
DE CONTROL A VARIABLES SEPARADAS
    int p0 = FilaCont & 0x01; //GUARDA EL PRIMER
VALOR
    int p1 = (FilaCont>>1) & 0x01; //GUARDA EL SEGUNDO
VALOR CON UN CORRIMIENTO
    int p2 = (FilaCont>>2) & 0x01; //GUARDA EL TERCER
VALOR CON OTRO CORRIMINETO
    digitalWrite(Pc0, p0); //ASIGNA EL PRIMER
VALOR AL PIN DE CONTROL 0
    digitalWrite(Pc1, p1); //ASIGNA EL SEGUNDO
VALOR AL PIN DE CONTROL 1
    digitalWrite(Pc2, p2); //ASIGNA EL TERCER
VALOR AL PIN DE CONTROL 2

    for(int r=0; r<8; r++){ //INICIA EL CICLO DE
LOS MODULOS DE SENSADO
        int FilaSens = DirSens[r]; //ASIGNA LA
DIRECCION DE CONTROL DE LOS SENSORES A VARIABLES SEPARADAS
        int s0 = FilaSens & 0x01; //GUARDA EL
PRIMER VALOR
        int s1 = (FilaSens>>1) & 0x01; //GUARDA EL
SEGUNDO VALOR CON UN CORRIMIENTO
        int s2 = (FilaSens>>2) & 0x01; //GUARDA EL
TERCER VALOR CON OTRO CORRIMINETO
        digitalWrite(Ps0, s0); //ASIGNA EL PRIMER
VALOR AL PIN DE CONTROL DEL SENSOR 0
        digitalWrite(Ps1, s1); //ASIGNA EL SEGUNDO
VALOR AL PIN DE CONTROL DEL SENSOR 1

```

```

    digitalWrite(Ps2, s2); //ASIGNA EL TERCER
VALOR AL PIN DE CONTROL DEL SENSOR 2

    int tMax = 0; //INICIALIZA UNA
VARIABLE TEMPORAL EN 0
    delay(3); //RETARDO DE 3 MS
    int tVal = analogRead(AnalogicoDos); //INICIALIZA
UNA VARIABLE TEMPORAL CON EL VALOR ANALOGICO DEL PIN 1
    tMax = max(tVal,tMax); //GUARDA EL VALOR
MAXIMO ENTRE EL VALOR LEIDO Y EL ANTERIOR
    if(laser == 'I'){ //VERIFICA SI LO
QUE ESTE A CONTINUACION SERA PARA EL LADO IZQUIERDO
        UmbralLasI[SensorNum] = CalculoDeVentana(tMax); //GUARDA EL
VALOR DEL CALCULO DE LA SOMBRA SEGUN EL VALOR GUARDADO
    }
    else{ //SI NO ES PARA EL
LADO IZQUIERDO SERA PARA EL DERECHO
        UmbralLasD[SensorNum] = CalculoDeVentana(tMax); //GUARDA EL
VALOR DEL CALCULO DE LA SOMBRA SEGUN EL VALOR GUARDADO
    }

    SensorNum++; // AUMENTA EN
1 EL CONTADOR
}
}
//===== PARTE DERECHA TERMINA

    if(laser == 'I'){ //SI ES EL LADO
IZQUIERDO
        digitalWrite(LaserI, LOW); //APAGA EL LASER
IZQUIERDO
    }
    else{ //POR EL CONTRARIO
ES DERECHO
        digitalWrite(LaserD, LOW); //APAGA EL LASER
DERECHO
    }
}
}

```

El “CálculoDeVentana” es la función que calcula el valor promedio obtenido por cada uno de los sensores con el valor máximo que pueden tener, esta función es útil para obtener un rango de precisión en la detección de la sombra que generen, es decir, si se compara el valor inicial con el detectado y se presenta una variación de valores, podemos decir que existe interferencia.

```

int CalculoDeVentana(int ValS){
    int SValor = ((1024 - ValS)/2) + ValS; //PROMEDIO (1024+ValS)/2
    return SValor;
}

```

La función “CalibraciónLáser” nos ayudará a checar manualmente que nuestros láseres estén alineados permitiendo que cada láser encienda por determinado tiempo para poder alinearlo correctamente.

```
void CalibracionLaser(){
    digitalWrite(LaserI, HIGH);           //ENCIENDE EL LASER
IZQUIERDO
    delay(5000);                          //ESPERA 5 SEGUNDOS
    digitalWrite(LaserI, LOW);           //APAGA EL LASER
IZQUIERDO
    delay(1000);                          //ESPERA 1 SEGUNDOS
    digitalWrite(LaserD, HIGH);         //ENCIENDE EL LASER
DERECHO
    delay(5000);                          //ESPERA 5 SEGUNDOS
    digitalWrite(LaserD, LOW);         //APAGA EL LASER DERECHO
}
```

La función “Escaneo” recibe un caracter, D o I, el cual indicara de qué lado iniciara el escaneo, de la parte derecha o de la parte izquierda de nuestro marco. Lo primero que hará será encender el láser de donde se va a escanear, posteriormente con unos ciclos “for” encadenados logrará cambiar las direcciones de los multiplexores de control a cada uno de los multiplexores de censado, de igual manera podrá cambiar la dirección de los multiplexores de censado para poder recolectar la información de cada uno de los fotodiodos, esta información será de manera analógica y se guardará en una variable que se comparara con la información guardada del inicio del programa. De haber sido diferente la información, significara que hubo un toque o movimiento en nuestro marco, por lo que se procederá a guardar el número de fotodiodo donde se presentó dicha variación y posteriormente se analizara este resultado

```
void Escaneo(char laser){
    if(laser == 'I'){                     //LASER IZQUIERDO
        digitalWrite(LaserI, HIGH);       //ENCIENDE EL LASER IZQUIERDO
        SensorNum = 32;                   //COLOCA COMO CONTADOR INICIAL 32
        ValorSombraI = NULL;              //INICIALIZA LA SOMBRE DE LOS LASERES
    }
    else{                                  //LASER DERECHO
        digitalWrite(LaserD, HIGH);       //ENCIENDE EL LASER DERECHO
        SensorNum = 0;                    //COLOCA COMO CONTADOR INICIAL 0
    }
}
```



```

    ValorSombraD = NULL;           //INICIALIZA LA SOMBRA DE LOS LASERES
EN 0
}

//===== PARTE IZQUIERDA DEL
FRAME CON LASER DERECHO E IZQUIERDO
for(int a=0; a<7; a++){           //INICIA EL
CICLO DE LOS MODULOS DE CONTROL
    if(((ValorSombraI == NULL) && (laser == 'I')) || ((ValorSombraD ==
NULL) && (laser == 'R'))){ //SI SE CUMPLE SIGUE
        if((laser == 'I') && (a <= 3)){           //SE OMITE EL
CICLO MIENTRAS SE CUMPLA LA CONDICION
            continue;                             //SE SALTA LOS
COMANDOS Y SIGUE CON LA SIGUINETE INTERACCION
        }
        int FilaCont = DirCont[a];               //ASIGNA LA
DIRECCION DE CONTROL A VARIABLES SEPARADAS
        int p0 = FilaCont & 0x01;                 //GUARDA EL
PRIMER VALOR
        int p1 = (FilaCont>>1) & 0x01;           //GUARDA EL
SEGUNDO VALOR CON UN CORRIMIENTO
        int p2 = (FilaCont>>2) & 0x01;           //GUARDA EL
TERCER VALOR CON OTRO CORRIMINETO
        digitalWrite(Pc0, p0);                   //ASIGNA EL
PRIMER VALOR AL PIN DE CONTROL 0
        digitalWrite(Pc1, p1);                   //ASIGNA EL
SEGUNDO VALOR AL PIN DE CONTROL 1
        digitalWrite(Pc2, p2);                   //ASIGNA EL
TERCER VALOR AL PIN DE CONTROL 2

        for(int s=0; s<8; s++){                 //INICIA EL
CICLO DE LOS MODULOS DE SENSADO
            int FilaSens = DirSens[s];           //ASIGNA LA
DIRECCION DE CONTROL DE LOS SENSORES A VARIABLES SEPARADAS
            int s0 = FilaSens & 0x01;             //GUARDA EL
PRIMER VALOR
            int s1 = (FilaSens>>1) & 0x01;       //GUARDA EL
SEGUNDO VALOR CON UN CORRIMIENTO
            int s2 = (FilaSens>>2) & 0x01;       //GUARDA EL
TERCER VALOR CON OTRO CORRIMINETO
            digitalWrite(Ps0, s0);               //ASIGNA EL
PRIMER VALOR AL PIN DE CONTROL DEL SENSOR 0
            digitalWrite(Ps1, s1);               //ASIGNA EL
SEGUNDO VALOR AL PIN DE CONTROL DEL SENSOR 1
            digitalWrite(Ps2, s2);               //ASIGNA EL
TERCER VALOR AL PIN DE CONTROL DEL SENSOR 2

            int tVal = analogRead(AnalogicoUno); //INICIALIZA
UNA VARIABLE TEMPORAL CON EL VALOR ANALOGICO DEL PIN 0
            if(laser == 'I'){                     //VERIFICA SI LO
QUE ESTE A CONTINUACION SERA PARA EL LADO IZQUIERDO
                if(tVal > UmbralLasI[SensorNum]){ //SI EL VALOR
TOMADO ES MAYOR AL GUARDADO EL LA INICIACION IZQUIERDA(TOQUE)
                    ValorSombraI = SensorNum;    //GUARDA EL
NUMERO DEL RECEPTOR DONDE HUBO TOQUE
                    break;                         //SALE DEL CICLO
                }
            }

```

```

    }
    else{ //SI NO ES PARA EL
LADO IZQUIERDO SERA PARA EL DERECHO
        if(tVal > UmbralLasD[SensorNum]){ //SI EL VALOR
TOMADO ES MAYOR AL GUARDADO EL LA INICIACION DERECHA(TOQUE)
            ValorSombraD = SensorNum; //GUARDA EL
NUMERO DEL RECEPTOR DONDE HUBO TOQUE
            break; //SALE DEL CICLO
        }
    }
    SensorNum++; //SI NO HUBO
DIFERENCIAS AUMENTA EN 1 EL CONTADOR
}
}
else{ //SI NO HUBO
NINGUN TOQUE EL EL LADO DERECHO
    break; //SALE DEL CICLO
}
}
//===== TERMINA PARTE IZQUIERDA
DEL FRAME CON LASER DERECHO E IZQUIERDO

//===== PARTE DERECHA DEL FRAME
CON LASER DERECHO E IZQUIERDO
    for(int b=0; b<7; b++){ //INICIA EL CICLO
DE LOS MODULOS DE CONTROL
        if(((ValorSombraI == NULL) && (laser == 'I')) || ((ValorSombraD ==
NULL) && (laser == 'D'))){ //SI SE CUMPLE SIGUE
            if((laser == 'D') && (b >= 3)){ //SE OMITE EL
CICLO MIENTRAS SE CUMPLA LA CONDICION
                break; //SE SALTA LOS
COMANDOS Y SIGUE CON LA SIGUINETE INTERACCION
            }
            int FilaCont = DirCont[b]; //ASIGNA LA
DIRECCION DE CONTROL A VARIABLES SEPARADAS
            int p0 = FilaCont & 0x01; //GUARDA EL
PRIMER VALOR
            int p1 = (FilaCont>>1) & 0x01; //GUARDA EL
SEGUNDO VALOR CON UN CORRIMIENTO
            int p2 = (FilaCont>>2) & 0x01; //GUARDA EL
TERCER VALOR CON OTRO CORRIMINETO
            digitalWrite(Pc0, p0); //ASIGNA EL PRIMER
VALOR AL PIN DE CONTROL 0
            digitalWrite(Pc1, p1); //ASIGNA EL SEGUNDO
VALOR AL PIN DE CONTROL 1
            digitalWrite(Pc2, p2); //ASIGNA EL TERCER
VALOR AL PIN DE CONTROL 2

            for(int r=0; r<8; r++){ //INICIA EL CICLO
DE LOS MODULOS DE SENSADO
                int FilaSens = DirSens[r]; //ASIGNA LA
DIRECCION DE CONTROL DE LOS SENSORES A VARIABLES SEPARADAS
                int s0 = FilaSens & 0x01; //GUARDA EL
PRIMER VALOR
                int s1 = (FilaSens>>1) & 0x01; //GUARDA EL
SEGUNDO VALOR CON UN CORRIMIENTO

```

```

        int s2 = (FilaSens>>2) & 0x01;           //GUARDA EL
TERCER VALOR CON OTRO CORRIMINETO
        digitalWrite(Ps0, s0);                 //ASIGNA EL PRIMER
VALOR AL PIN DE CONTROL DEL SENSOR 0
        digitalWrite(Ps1, s1);                 //ASIGNA EL SEGUNDO
VALOR AL PIN DE CONTROL DEL SENSOR 1
        digitalWrite(Ps2, s2);                 //ASIGNA EL TERCER
VALOR AL PIN DE CONTROL DEL SENSOR 2

        int tVal = analogRead(AnalogicoDos);    //INICIALIZA
UNA VARIABLE TEMPORAL CON EL VALOR ANALOGICO DEL PIN 1
        if(laser == 'I'){                       //VERIFICA SI LO QUE
ESTE A CONTINUACION SERA PARA EL LADO IZQUIERDO
            if(tVal > UmbralLasI[SensorNum]){    //SI EL VALOR
TOMADO ES MAYOR AL GUARDADO EL LA INICIACION IZQUIERDA(TOQUE)
                ValorSombraI = SensorNum;        //GUARDA EL
NUMERO DEL RECEPTOR DONDE HUBO TOQUE
                break;                             //SALE DEL CICLO
            }
        }
        else{                                    //SI NO ES PARA EL
LADO IZQUIERDO SERA PARA EL DERECHO
            if(tVal > UmbralLasD[SensorNum]){    //SI EL VALOR
TOMADO ES MAYOR AL GUARDADO EL LA INICIACION DERECHA(TOQUE)
                ValorSombraD = SensorNum;        //GUARDA EL
NUMERO DEL RECEPTOR DONDE HUBO TOQUE
                break;                             //SALE DEL CICLO
            }
        }
        SensorNum++;                             //SI NO HUBO
DIFERENCIAS AUMENTA EN 1 EL CONTADOR
    }
}
else{                                            //SI NO HUBO NINGUN
TOQUE EL EL LADO DERECHO
    break;                                       //SALE DEL CICLO
}
}
//===== TERMINA PARTE DERECHA
DEL FRAME CON LASER DERECHO E IZQUIERDO

    if(laser == 'I'){                             //SI ES EL LASER
IZQUIERDO
        digitalWrite(LaserI, LOW);              // APAGA EL LASER
IZQUIERDO
    }
    else{                                         // SI NO ES EL
IZQUIERDO ES EL DERECHO
        digitalWrite(LaserD, LOW);              //APAGA EL LASER
DERECHO
    }
}

```

La función “Coordenadas” permitirá obtener la coordenada del punto de toque por medio del cálculo de una intersección obtenida por ambos lado del marco, parte derecha y parte

izquierda, este cálculo es por medio de las ecuaciones posteriormente colocadas en el código donde se obtienen las pendientes, las ordenadas al origen y dos puntos de cada una de las rectas. Por medio de las funciones “SensorX” y “SensorY” podemos aproximar el número del sensor que se está interfiriendo al relacionarlo a la resolución de la pantalla (1024x768) que estemos manejando.

```

void Coordenadas(){
    float IzqX1 = 0; //COORDENADA X1
=0 DEL LADO IZQUIERDO
    float IzqY1 = 0; //COORDENADA Y1
=0 DEL LADO IZQUIERDO
    float IzqX2 = SensorX(ValorSombraI); //MANDA A BUSCAR
LA COORDENADA X2 DEL LADO IZQUIERDO
    float IzqY2 = SensorY(ValorSombraI); //MANDA A BUSCAR LA
COORDENADA Y2 DEL LADO IZQUIERDO
    float DerX1 = 1024; //COORDENADA X1
=321 DEL LADO DERECHO
    float DerY1 = 0; //COORDENADA X1
=0 DEL LADO DERECHO
    float DerX2 = SensorX(ValorSombraD); //MANDA A BUSCAR LA
COORDENADA X2 DEL LADO DERECHO
    float DerY2 = SensorY(ValorSombraD); //MANDA A BUSCAR LA
COORDENADA Y2 DEL LADO DERECHO

    float PendienteIzq = (IzqY2-IzqY1)/(IzqX2-IzqX1); //PENDIENTE DEL
LADO IZQUIERDO
    float OrdeOrigenIzq = 0; //OREGANADA AL
ORIGEN EN LADO IZQUIERDO ES =0
    float PendienteDer = (DerY2-DerY1)/(DerX2-DerX1); //PENDIENTE DEL LADO
DERECHO
    float OrdeOrigenDer = (-PendienteDer*DerX1)+DerY1; //-MX+Y ORDENADA
AL ORIGEN EN LADO DERECHO

    if((InicioDeX != NULL) && (FinDeX != NULL)){ // CASO EN EL QUE
EXISTE MOVIMIENTO
        InicioDeX = FinDeX; //LA COORDENADA
INICIAL EN X SE CONVIERTE EN FINAL
        InicioDeY = FinDeY; //LA COORDENADA
INICIAL EN Y SE CONVIERTE EN FINAL
        FinDeX = (OrdeOrigenDer - OrdeOrigenIzq)/(PendienteIzq -
PendienteDer);
        FinDeY = PendienteIzq * FinDeX;
    }
    else if(InicioDeX == NULL){ //DIFERENTES
CASOS
        InicioDeX = (OrdeOrigenDer - OrdeOrigenIzq)/(PendienteIzq -
PendienteDer);
        InicioDeY = PendienteIzq * InicioDeX;
    }
    else if(FinDeX == NULL){
        FinDeX = (OrdeOrigenDer - OrdeOrigenIzq)/(PendienteIzq -
PendienteDer);
    }
}

```

```

    FinDeY = PendienteIzq * FinDeX;
}
}

```

```

SENSOR X
~~~~~
~~ */
float SensorX(int sensorX){           //OBTENCION DEL SENSOR X
    int tVal=1;                       //CREA UNA VARIABLE Y LA
    INICIALIZA CON 1
    if((sensorX >= 32) && (sensorX <= 39)){ //SI EL VALOR INGRESADO ES
MAYOR QUE 24 Y MENOR QUE 33
        tVal = (sensorX - 31) * 21;     //AL VALOR SE LE RESTARA 23
Y SE MULTIPLICARA POR 5
    }
    if((sensorX >= 40) && (sensorX <= 47)){ //SI EL VALOR INGRESADO ES
MAYOR QUE 34 Y MENOR QUE 45
        tVal = ((sensorX - 31) * 21) + 1; //AL VALOR SE LE RESTARA 23
SE MULTIPLICARA POR 5 Y SE LE SUMARA 1
    }
    if((sensorX >= 48) && (sensorX <= 55)){ //SI EL VALOR INGRESADO ES
MAYOR QUE 46 Y MENOR QUE 54
        tVal = ((sensorX - 31) * 21) + 2; //AL VALOR SE LE RESTARA 23
SE MULTIPLICARA POR 5 Y SE LE SUMARA 2
    }
    if((sensorX >= 56) && (sensorX <= 63)){ //SI EL VALOR INGRESADO ES
MAYOR QUE 55 Y MENOR QUE 65
        tVal = ((sensorX - 31) * 21) + 3; //AL VALOR SE LE RESTARA 23
SE MULTIPLICARA POR 5 Y SE LE SUMARA 3
    }
    if((sensorX >= 64) && (sensorX <= 71)){ //SI EL VALOR INGRESADO ES
MAYOR QUE 66 Y MENOR QUE 79
        tVal = ((sensorX - 31) * 21) + 4; //AL VALOR SE LE RESTARA 23
SE MULTIPLICARA POR 5 Y SE LE SUMARA 4
    }
    if((sensorX >= 72) && (sensorX <= 79)){ //SI EL VALOR INGRESADO ES
MAYOR QUE 80 Y MENOR QUE 86
        tVal = ((sensorX - 31) * 21) + 5; //AL VALOR SE LE RESTARA 23
SE MULTIPLICARA POR 5 Y SE LE SUMARA 5
    }
    if((sensorX >= 80) && (sensorX <= 111)){ //SI EL VALOR INGRESADO ES
MAYOR QUE 87 Y MENOR QUE 109
        tVal = 1023;                     //LA VARIABLE SERA 321
    }
    return tVal;                          //REGRESA EL VALOR DE LA
VARIABLE
}

/* ~~~~~
~~~~~
SENSOR Y
~~~~~
~~ */
float SensorY(int sensorY){           //OBTENCION DEL SENSOR Y

```

```

    int tVal=1; //CREA UNA VARIABLE Y LA
INICIALIZA CON 1
    if((sensorY >= 1) && (sensorY <= 15)){ //SI EL VALOR INGRESADO ES
MAYOR QUE 1 Y MENOR QUE 10
        tVal = sensorY * 24; //AL VALOR SE LE
MULTIPLICARA POR 5
    }
    if((sensorY >= 16) && (sensorY <= 31)){ //SI EL VALOR INGRESADO ES
MAYOR QUE 11 Y MENOR QUE 22
        tVal = sensorY * 24 + 1; //AL VALOR SE LE
MULTIPLICARA POR 5 Y SE LE SUMARA 1
    }
    if((sensorY >= 32) && (sensorY <= 79)){ //SI EL VALOR INGRESADO ES
MAYOR QUE 23 Y MENOR QUE 86
        tVal = 768; //LA VARIABLE SERA 115
    }
    if((sensorY >= 80) && (sensorY <= 95)){ //SI EL VALOR INGRESADO ES
MAYOR QUE 87 Y MENOR QUE 98
        tVal = ((111 - sensorY) * 24) + 1; //A 109 SE LE RESTARA EL
VALOR, SE LE MULTIPLICARA POR 5 Y SE LE SUMARA 1
    }
    if((sensorY >= 96) && (sensorY <= 111)){ //SI EL VALOR INGRESADO ES
MAYOR QUE 99 Y MENOR QUE 109
        tVal = (111 - sensorY) * 24; //A 109 SE LE RESTARA EL
VALOR, SE LE MULTIPLICARA POR 5
    }
    return tVal; //REGRESA EL VALOR DE LA
VARIABLE
}

```

Finalmente la función “MueveCursor” tomará los valores de las últimas coordenadas y se le restará las coordenadas anteriores para lograr una transición en la pantalla. Estos datos los guardará en una sola cadena, es decir, los datos de las coordenadas XY del punto se enviarán al puerto serie en un solo dato y así el ordenador interpretara esto como un punto de toque.

```

void MueveCursor(){
    float MueveX = FinDeX - InicioDeX; //CREA UNA VARIABLE CON
EL CORRIMIENTO DEL PUNTO FINAL AL INICIAL EN X
    float MueveY = FinDeY - InicioDeY; //CREA UNA VARIABLE CON
EL CORRIMIENTO DEL PUNTO FINAL AL INICIAL EN Y
    int xInt = (int) MueveX; //CONVIERTE LA VARIABLE
EN X, EN TIPO ENTERA
    int yInt = (int) MueveY; //CONVIERTE LA VARIABLE
EN Y, EN TIPO ENTERA /*

    Serial.println(xInt);
    Serial.println(yInt);
    delay(100);
}

```

COMPARACIÓN CON OTROS MODELOS DEL MERCADO

3.1. FABRICANTES DE MARCOS MULTITÁCTILES

Existen diferentes fabricantes de marcos multitáctil, tal es el caso de la empresa de **Music Computing** que se dedica a la interacción humana basado en computadoras. Sus principales clientes son los músicos, universidades, museos y aficionados entre otros. Esta empresa se encuentra en Austin Texas y aunque su especialidad son los instrumentos musicales con pantalla multitáctil como los teclados, también distribuyen marcos para mejorar las experiencias de sus usuarios. En la figura 3.1 se puede apreciar la imagen de un marco multitáctil de dicha empresa



Figura 3.1 Marco multitáctil de Music Computing. Figura tomada de <http://www.musiccomputing.com/motioncommand-windows>

Interactuando se ubica en Sevilla, España. Es una empresa que se ha hecho presente en varios eventos como: Fashion Week Barcelona, en el Congreso de las Enfermedades Cardiovasculares, en el Club de Empresas Antares, entre otras, para promocionar sus productos. Entre estos se encuentran: alfombras interactivas, cubos interactivos, mesas interactivas profesionales, mesas interactivas para niños, pizarrones digitales, entre otros y por supuesto cuentan con marcos multitáctil, figura 3.2.



Figura 3.2 Marco multitáctil de Interactuando. Figura tomada de <http://www.interactuando.es/index.php/marco-tactil-multipunto/>

La empresa **Peau Productions** se ubica en San Diego California, Estados Unidos. Es una compañía que ofrece un servicio completo de productos interactivos, al igual que Interactuando, cuenta con una gran variedad de productos: marcos multitáctil – figura 3.3, mesas interactivas, cámaras infrarrojas y multitáctiles, accesorios para cámaras, leds infrarrojos, pizarrones electrónicos entre otros.



Figura 3.3 Marco multitáctil de Peau Productions. Figura tomada de http://peauproductions.com/store/index.php?main_page=product_info&cPath=29_85_86&products_id=426

La idea de crear un marco multitáctil comenzó en el “Aula del futuro” ubicado en el Centro de Ciencias Aplicadas y Desarrollo Tecnológico **CCADET** de la UNAM, centro multidisciplinario en el que se busca conjuntar la investigación y el desarrollo tecnológico en diversas áreas de las ciencias físico-matemáticas y de las ingenierías, con el fin de desarrollar proyectos de amplio alcance dirigidos a resolver problemas relevantes de

nuestro entorno. En la figura 3.4 se puede apreciar un poco del trabajo que se ha realizado dentro del Aula del futuro, donde se observa una mesa interactiva con un grupo de niños que trabajan en ella.



Figura 3.4 Aula del Futuro. Figura tomada de http://fotos.eluniversal.com.mx/coleccion/muestra_fotogaleria.html?idgal=13536

3.2. FUNCIONAMIENTO

Como ya se explicó en el capítulo 1 existen diferentes tecnologías para las superficies multitáctiles, sin embargo, algunas empresas dedicadas a fabricar marcos multitáctil, no especifican el tipo de tecnología que usan para sus productos. Por ejemplo, la empresa **Music Computing** en su página de internet solo describe el producto: puntos de toque, pulgadas del marco, sistema operativo con el que funciona y precio, pero no especifica la tecnología empleada.

En el caso de la empresa **Interactuando**, ellos si especifican en su página el tipo de tecnología que están usando, para ellos la tecnología infrarroja es la idónea para los marcos

táctiles que están lanzando al mercado, ya que cuentan con una alta precisión y respuesta inmediata. De igual forma, **Peau Productions** usa la tecnología de infrarrojos para sus marcos.

Y aunque la empresa de **Music Computing** no especifique el tipo de tecnología a usar, por los conocimientos que se han adquirido durante el desarrollo de este trabajo, se puede deducir que también usan la tecnología infrarroja para sus marcos multitáctil.

La tecnología infrarroja ha demostrado ser la más idónea para este tipo de producto, tanto por el buen funcionamiento que ofrece, así como un precio más económico comparado con otras tecnologías, como la capacitiva, que ha demostrado ser una buena tecnología por sus ventajas que presenta, sin embargo, es demasiado costosa para los usuarios.

Es por ello que se analizó e investigó el funcionamiento de la tecnología infrarroja, para poder buscar los dispositivos que en conjunto formarán nuestro marco.

El funcionamiento de nuestro marco consiste en usar la tecnología infrarroja, por medio de unos láseres que estarán colocados en la parte inferior del marco, cada uno en un extremo, para generar un plano de luz. Cada láser tendrá instalado una lente generadora de línea de 120°, por medio del cual el haz del láser se expandirá como un “abanico” cubriendo toda la superficie multitouch. Alrededor del marco habrá una serie de fotodiodos, los cuales detectarán cuando se presente una variación de intensidad de la luz y por ende el voltaje de salida variará. Los voltajes de salida serán transmitidos por los multiplexores, que a su vez les enviarán los datos recabados al Arduino para que este pueda mandar la orden de donde colocar el cursor en base al punto de toque en el marco.

Este procedimiento de detección se realiza de manera veloz para cada uno de los toques que se realicen en el marco multitáctil.

3.3 ÁREA DE PROYECCIÓN Y PUNTOS DE TOQUE

Un elemento importante dentro de este proyecto es el área de proyección, donde el ancho y el alto tienen gran importancia ya que se ha identificado que el formato de proyección 4:3 no es muy adecuado visualmente ya que no permite aprovechar todos los elementos ubicándolos en un mejor espacio, es por eso que el formato 16:9 o el panorámico ofrece una mejor proporción en el espacio, además de que actualmente las pantallas se están fabricando en este formato.

En relación al área de proyección se encuentra la resolución que se refiere al número de puntos en la que está formada nuestra imagen o proyección, en las resoluciones más bajas encontramos la SVGA la cual forma una imagen de 480 mil puntos o píxeles en un formato de 4:3 (800 píxeles en horizontal por 600 píxeles en vertical), buscando el formato 16:9 encontramos la resolución XGA con una mejora en la cantidad de píxeles conformada por 786 432 puntos (1024 píxeles en horizontal por 768 píxeles en vertical), este formato es el estándar al cual los equipos de cómputo trabajan y es por ello que nos basamos en este formato.

Nos referimos como los puntos de toque a la cantidad de toques que se pueden detectar simultáneamente dentro del área de proyección, dentro de esta existen categorías para definirlos; touch se refiere a que solamente puede detectar un solo toque o punto en la pantalla, si existieran dos toques, la pantalla solamente detectaría el primero. Dual-touch como su nombre lo indica, solo detecta dos puntos de toque, no más, de igual manera si existen más toques solo tomara los dos primeros. Finalmente la Multitouch donde se le denomina a las superficies que pueden detectar 2 o más toques dentro de la superficie.

La empresa **Music Computing** tiene disponibles diferentes tamaños 17", 19", 20", 22", 24", 32", 40", 42", 55", 65" y 80". Los marcos de hasta 22" tienen capacidad para 2 puntos de toque y de 32" en adelante puede trabajar con 4 y hasta con 40 puntos de toque como máximo. El tiempo de respuesta es de 7 a 12ms pero esto depende de la potencia de la computadora en la que esté instalado el marco multitáctil

Interactuando es una empresa que maneja tamaños de 17” hasta 80” con tecnología infrarroja de alta precisión y respuesta inmediata.

Peau Productions es una compañía estadounidense dedicada a crear componentes para lograr superficies multitouch, dentro de su catálogo encontramos 3 tipos de tecnología, la primera se trata de touchscreens basados en cámaras que captan la luz infrarroja por lo que es necesario una proyección y una distancia entre el touchscreen y la cámara, la segunda se trata de un film multitouch basado en la tecnología capacitiva donde este es adherido a la pantalla de computadora o tv donde se quiera realizar la interacción, por ultimo un marco multitáctil basado en interferencia infrarroja donde se coloca por encima de la superficie de interacción. Cada una de estos dispositivos se ponen a la venta variando el precio según los puntos de toque y medidas que al usuario le convengan que van desde las 15 a las 103 pulgadas y desde 2 puntos de toque hasta los 32.

3.4 APLICACIONES

El principal objetivo de un marco multitáctil es poder colocarlo sobre una pantalla LED, LCD o PLASMA y convertirla a pantalla táctil, por lo que nuestros dedos serán los que podrán manipular la pantalla.

Las aplicaciones de los marcos multitáctiles son muy diversas, a continuación se mencionarán algunas de las tantas aplicaciones que tienen:

- Eventos y salas de presentaciones
- Demostraciones de productos
- Puntos de información
- Conciertos
- Museos
- Entorno educativo
- Hoteles
- Centros comerciales
- Centros de ocio
- Bancos y cajas

- Parques de atracciones
- Aeropuertos
- Agencia de viaje
- Casinos
- Proyectos de arquitectura
- Lugares públicos

Como se puede observar, los marcos multitáctiles se pueden colocar en cualquier lugar que se quiere convertir más novedoso, interactiva e interesante para la gente.

3.5 PRECIOS

Cuando se habla de adquirir un nuevo producto tecnológico, se consideran factores como: funcionamiento, tipo de tecnología, aplicaciones, ventajas, desventajas, pero sobre todo, el factor que más nos puede llegar a interesar es el precio. Y los marcos multitáctiles no son la excepción, sin embargo, su costo puede variar dependiendo la empresa que elijamos para comprar uno.

Los marcos multitáctiles varían su precio en base al tamaño y al número de toques que detecte en un mismo tiempo. En la mayoría de los dispositivos electrónicos, entre más grande sea, más caro es, por ejemplo: televisiones, celulares, tablets, cámaras fotográficas etc.

A continuación se muestran unas tablas de precios para marcos multitáctiles de la empresa Music Computing y Peau Productions, tabla 3.1 y 3.2 respectivamente.

Music Computing

Pulgadas	Puntos	SO	Precio	Modelo
17	2	Windows 7 y 8	\$199	MCF17W
	1	Mac OS X		
19	2	Windows 7 y 8	\$255	MCF22W2
	1	Mac OS X		
20	2	Windows 7 y 8	\$269	MCF22W21
22	2	Windows 7 y 8	\$279	MCF22W
	1	Mac OS X		
24	2	Windows 7 y 8	\$299	MCF24W1
26	2	Windows 7 y 8	\$349	MCF24W12
27	2	Windows 7 y 8	\$369	MCF24W122
32	10	Windows 7 y 8	\$1,199	MCF32W13
		Mac OS X		
	4	Windows 7 y 8	\$1,199	MCF32W1
	2	Windows 7 y 8	\$729	MCF32W
		1		
32- RA	2	Windows 7 y 8	\$2,329	MCF32W2
	1	Mac OS X		
32-RA	4	Windows 7 y 8	\$2,799	MCF32W11
	1	Mac OS X		
40	2	Windows 7 y 8	\$965	MCF42W3
	1	Mac OS X		
	4	Windows 7 y 8	\$1,459	MCF42W21
40-RA	2	Windows 7 y 8	\$2,599	MCF42W31
	1	Mac OS X		
	4	Windows 7 y 8	\$2,999	MCF42W211

Pulgadas	Puntos	SO	Precio	Modelo
42	10	Windows 7 y 8	\$1,559	MCF42W24
		Mac OS X		
	4	Windows 7 y 8	\$1,559	MCF42W2
	2	Windows 7 y 8	\$1,059	MCF42W
1	Mac OS X			
42-RA	2	Windows 7 y 8	\$2,759	MCF42W4
	1	Mac OS X		
	4	Windows 7 y 8	\$3,259	MCF42W22
46	2	Windows 7 y 8	\$1,245	MCF42W1
	1	Mac OS X		
	4	Windows 7 y 8	\$1,799	MCF42W11
46-RA	2	Windows 7 y 8	\$3,045	MCF42W12
	1	Mac OS X		
	4	Windows 7 y 8	\$3,599	MCF42W111
47	2	Windows 7 y 8	\$1,299	MCF42W13
	1	Mac OS X		
	4	Windows 7 y 8	\$1,849	MCF42W112
47-RA	2	Windows 7 y 8	\$3,145	MCF42W121
	1	Mac OS X		
	4	Windows 7 y 8	\$3,699	MCF42W1111
50	2	Windows 7 y 8	\$1,749	MCF55W41
	1	Mac OS X		
	4	Windows 7 y 8	\$2,299	MCF55W2111

Pulgadas	Puntos	SO	Precio	Modelo
52	2	Windows 7 y 8	\$1,799	MCF55W4
	1	Mac OS X		
	4	Windows 7 y 8	\$2,399	MCF55W211
52-RA	2	Windows 7 y 8	\$3,699	MCF55W31
	1	Mac OS X		
	4	Windows 7 y 8	\$4,299	MCF55W212
55	10	Windows 7 y 8	\$2,599	MCF55W214
		Mac OS X		
	4	Windows 7 y 8	\$2,599	MCF55W21
	2	Windows 7 y 8	\$1,999	MCF55W
1	Mac OS X			
55-RA	2	Windows 7 y 8	\$4,899	MCF55W3
	1	Mac OS X		
	4	Windows 7 y 8	\$5,499	MCF55W2
65	2	Windows 7 y 8	\$2,099	MCF65W
	1	Mac OS X		
	4	Windows 7 y 8	\$3,699	MCF65W2

Pulgadas	Puntos	SO	Precio	Modelo
65-RA	2	Windows 7 y 8	\$6,299	MCF65W3
	1	Mac OS X		
	4	Windows 7 y 8	\$6,699	MCF65W21
80	2	Windows 7 y 8	\$3,499	MCF65W1
	1	Mac OS X		
	4	Windows 7 y 8	\$4,099	MCF65W11

Tabla 3.1 Tabla de precios de Music Computing. Tabla de autoría propia.

NOTA:

- Todos los precios están en dólares
- RA-Resistente al Agua
- Última actualización de precios diciembre 2014
- El precio del marco no incluye el costo de envío

Peau Productions

Pulgadas	Puntos	SO	G4S 4mm	G4 8mm
15	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux	\$109	
	6		\$149	
17	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux	\$129	
	6		\$189	
19	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux	\$179	
	6		\$239	
21.5	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux	\$209	
	6		\$279	
	32		\$315	
24	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux	\$249	
	6		\$339	
	32		\$405	
27	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux	\$319	
	6		\$435	
	32		\$570	
32	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux	\$562	\$537
	6		\$749	\$709
	12		\$904	\$810
	32		\$1,039	\$934
40	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux	\$691	\$659
	6		\$955	\$900
	12		\$1,085	\$965
	32		\$1,295	\$1,165
42	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux	\$700	\$667
	6		\$987	\$855
	12		\$1,130	\$1,005
	32		\$1,410	\$1,255
46	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux	\$740	\$706
	6		\$1,058	\$932
	12		\$1,173	\$1,050
	32		\$1,513	\$1,360

Pulgadas	Puntos	SO	G4S 4mm	G4 8mm
47	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux		\$719
	6			\$937
	12			\$1,083
	32			\$1,395
50	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux		\$737
	6			\$1,010
	12			\$1,108
	32			\$1,495
52	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux		\$763
	6			\$1,020
	12			\$1,150
	32			\$1,545
55	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux	\$862	\$824
	6		\$1,157	\$1,095
	12		\$1,377	\$1,232
	32		\$1,787	\$1,640
60	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux		\$967
	6			\$1,285
	12			\$1,454
	32			\$1,765
65	2	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux		\$1,090
	6			\$1,465
	12			\$1,652
	32			\$1,858
70	6	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux		\$2,300
	12			\$2,680
	32			\$3,200
72	6	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux		\$2,360

Pulgadas	Puntos	SO	G4S 4mm	G4 8mm
80	6	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux		\$2,660
	12			\$3,100
	32			\$3,715
84	6	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux		\$2,750
	12			\$3,210
	32			\$3,845
90	6	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux		\$2,960
	12			\$3,460
	32			\$4,105
103	6	Windows 7, 8, Vista, XP, 2000, Mac OS X y Linux		\$3,290
	12			\$3,850
	32			\$4,615

Tabla 3.2 Tabla de precios de Peau Productions. Tabla de autoría propia

NOTAS:

- Todos los precios están en dólares
- Modelo G4S: marco con 4 mm de espesor
Modelo G4: marco con 8 mm de espesor
- Última actualización de precios diciembre 2014
- El precio del marco incluye costo de envío

Durante la realización de nuestro proyecto incluimos materiales que ya teníamos (Arduino UNO, láser, lentes generadoras, acrílico, transistores), sin embargo, en la tabla 3.3 se incluyen los precios de todos los dispositivos involucrados en el diseño.

Descripción	Cantidad	Usado	Precio	Total	Nota
Demultiplexor Analógico	20	16	\$5.17	\$103.44	Encapsulado DIP HCF4051
Fotodiodo PT331C	150	112	\$3.88	\$581.85	Receptor infrarrojo
Transistor NPN BC337	5	2	\$1.72	\$8.62	Encapsulado TO-92
Arduino UNO REV3	1	1	\$390.00	\$390.00	Arduino UNO REV3
Lente generadora de 120°	2	2	\$43.20	\$86.40	El precio puede variar en base al dólar.
Láser infrarrojo 850nm 28mW	2	2	\$950.40	\$1,900.80	El precio puede variar en base al dólar.
Resistencias 10kΩ	150	112	\$0.56	\$84.00	Mayoreo a partir de 25 piezas
Headers	3	3	\$10.00	\$30.00	Tira con 40 headers
Placa fenólica	4	4	\$63.00	\$252.00	Placa perforada de 10.5x14cm
Acrílico	1	1	\$60.00	\$60.00	3mm de espesor 40x50 cm
Otros	1	1	\$100.00	\$100.00	Cable, soldadura, canaleta, etc.
Total				\$3,597.11	MXN

Tabla 3.3 Tabla de precios de nuestra propuesta. Tabla de autoría propia

NOTAS:

- Los multiplexores, fotodiodos y transistores son cotizados en:
<http://www.agelectronica.com/inicio.htm>
- El generador de línea y el láser infrarrojo son cotizados en:
<http://www.peauproductions.com/main.html>
- El Arduino y las resistencias son cotizadas en:
<http://www.330ohms.com/>
- Acrílico
<http://www.showdepot.com.mx>

CAPÍTULO 4

RESULTADOS ESPERADOS.

Una vez realizado el diseño físico del marco, considerando componentes económicos, usando la menor cantidad de cables físicos en las placas fenólicas y con lo mínimo de componentes, se espera que sea una herramienta útil, que ayude al entorno educativo a que los alumnos se interesen por aprender cada día más y que tengan herramientas tecnológicas y novedosas al alcance de sus manos.

La principal idea de que los modelos de recepción tengan el mismo funcionamiento es con la finalidad de que si en un futuro se desea diseñar un marco multitáctil con una dimensión mayor, sea más fácil su acoplamiento con los demás módulos ya instalados, así que solo habría que crear más módulos de recepción, colocarlos en los extremos del marco y también crear más módulos de control para recibir las entradas de los nuevos módulos de recepción instalados, sin olvidar que cada módulo de control está diseñado para controlar hasta 8 módulos de recepción.

Pensando en expansiones futuras respecto a las dimensiones del marco, se consideró que tanto los módulos de control como los de recepción tengan interconexiones en los extremos de cada módulo, para lograr una mejor conexión entre esto. Cada una de estas conexiones son comunes entre los módulos de recepción así que es indiferente el lugar donde se conecte, de igual manera la alimentación de estos módulos es por esta misma conexión donde los 5 volts provenientes del Arduino podrá alimentar cada uno de estos sin que sea necesario una fuente extra.

Se espera que este diseño simplificado que proponemos para el reconocimiento de toques y movimientos en la pantalla, sea suficiente para poder manipular con facilidad y con exactitud un entorno gráfico.

CAPÍTULO 5

CONCLUSIONES.

El objetivo principal de esta tesis, como su nombre lo indica, fue diseñar un sistema de fácil implementación que fuera capaz de montarse a cualquier tipo de pantallas. Teniendo así una base para futuras mejoras de software y hardware; una vez logrado esto, se podrá tener una manipulación total con el entorno gráfico (Windows) de una manera más interactiva, precisa y permitiendo crear una superficie multitouch confiable.

Podemos decir que nuestro proyecto es un paso más para la evolución informática que está teniendo lugar en el país y que gran cantidad de usuarios sean beneficiados con la ayuda de nuestro marco, tanto en las áreas comerciales y de entretenimiento hasta las áreas educativas y culturales.

El diseño del marco multitáctil se optimizó para lograr un mínimo de componentes y poder bajar los costos de fabricación pero sin dejar de ser funcional. Se espera que este diseño sea capaz de capturar múltiples toques simplemente modificando el código y, si es posible, aumentar la velocidad a la que trabaje ya que entre mayor velocidad tenga el dispositivo de control será más sensible al movimiento de los toques.

En un proyecto a corto plazo se pretende que por medio de la programación se puedan reconocer gestos y poder así aumentar la funcionalidad de nuestro marco; por otro lado, se pondrá en desarrollo este proyecto para poder implementarlo en el Aula del Futuro dentro de las instalaciones del CCADET y tal vez poder instalarlo en las aulas de bachillerato como una herramienta más para el aprendizaje. Esperando así una mayor participación e interés de los alumnos en las clases, ya que contarán con herramientas de trabajo llamativas e inteligentes como lo son las superficies táctiles. Actualmente y de manera paralela, otros integrantes del grupo del Aula del Futuro han desarrollado algunas aplicaciones en matemáticas y en el área experimental para el uso de estas mesas con alumnos, de las cuales se han realizado algunas evaluaciones en las que se ha patentizado el aprovechamiento de estas mesas.

Fue bueno diseñar un hardware con tecnologías accesibles y no con las que se encuentran en los dispositivos móviles y tabletas electrónicas, ya que serían costosos para el área en la que queremos centrarnos, que serían superficies de pantallas hasta mesas de trabajo.

Esperamos que este diseño pueda implementarse en diferentes áreas y que sea un impulsor de nuevos diseños, más sofisticados, pero sin olvidar que sus componentes sean de fácil acceso y económicos para los usuarios.

Algoritmos

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

Capacitancia

Es la capacidad que tienen los conductores eléctricos de poder admitir cargas cuando son sometidos a un potencial.

Difusor

Es cualquier dispositivo que difunde, esparce o distribuye la luz de alguna manera.

Electroluminiscencia

Es un fenómeno óptico y eléctrico en el cual un material emite luz en respuesta a una corriente eléctrica que fluye a través de él, o por causa de la fuerza de un campo eléctrico.

Fiduciales

Son figuras o imágenes de referencia que al ser captadas por un dispositivo producen, alguna acción predefinida.

Gadgets

Dispositivo que tiene un propósito y una función específica, generalmente de pequeñas proporciones, práctico y a la vez novedoso.

Piezoeléctricos

Cristales que al ser sometidos a tensiones mecánicas adquieren una polarización eléctrica, si se tallan estos cristales de una determinada manera, la frecuencia de resonancia es estable y bien definida.

Polarización

Es la modificación de la distribución de carga que ocurre en un material aislador por efecto de un campo eléctrico.

Reflexión

Es un fenómeno en virtud del cual la luz al incidir sobre la superficie de los cuerpos cambia de dirección, invirtiéndose el sentido de su propagación.

Semiconductor

Es un elemento que se comporta como un conductor o como aislante dependiendo de diversos factores, como por ejemplo el campo eléctrico o magnético, la presión, la radiación que le incide, o la temperatura del ambiente en el que se encuentre.

Sistema combinacional o lógica combinacional

Sistema digital en el que sus salidas son función exclusiva del valor de sus entradas en un momento dado

Sticks

Son básicamente microordenadores con Android que mediante HDMI convierten cualquier televisión en una Android TV.

Transductores

Dispositivo que convierte una señal de un tipo de energía en otra.

Transmitancia

Magnitud que expresa la cantidad de energía que atraviesa un cuerpo en la unidad de tiempo.

REFERENCIA BIBLIOGRÁFICA

Aixiz 980nm Lasers

<http://www.aixiz.com/store/index.php/cPath/67>

(Noviembre 2013)

Brian w. Evans, **Arduino programming notebook**, primera edición agosto 2007.

(Abril 2013)

McRoberts Michael, **Beginning Arduino**, ed TECHNOLOGY IN ACTION

(Abril 2013)

Building Interactive Multi-Touch, Surfaces Vol. 14, No. 3: 35-55

<http://research.microsoft.com/pubs/132465/fulltext.pdf>

(Abril 2013)

Gonzalo Marcos Díez, **Diseño e implementación de una mesa multi-touch.**

Universidad Autónoma de Madrid.

(Abril 2013)

Grupo NUI. 2009. **Multi-Touch Technologies**. 1^a edición [Comunidad de prensa]

<http://nuicode.com/projects/wiki-book/files>.

(Diciembre 2013)

Van Der Veen Harry, **FTIR multi-touch display how-to guide**, (alpha version

0.2), May 23, 2007

<http://nuigroup.com/forums>

(Abril 2013)

<http://www.a-touch.com.tw/index.htm>

(Abril 2013)

<http://www.ardumania.es/>

(Abril 2013)

<http://www.arduteka.com/Arduino/#>

(Abril 2013)

<http://www.ccadet.unam.mx/>

(Noviembre 2013)

<http://www.elotouch.com.ar/Productos/detech2.asp>

(Abril 2013)

<http://www.interactuando.es/>

(Noviembre 2013)

<http://www.musiccomputing.com/>

(Noviembre 2013)

<http://www.peauproductions.com/main.html>

(Diciembre 2013)

<http://www.prodisplay.com/index.html>

(Abril 2013)

Schöning Johannes, **Multi-Touch Surfaces: A Technical Guide**, Technical Report TUM-I0833

<http://homepages.cs.ncl.ac.uk/jonathan.hook/publications/schoening2008multitouch.pdf>

(Abril 2013)

Pomares Baeza Jorge, **Manual de Arduino**, Universidad de Alicante, 2009
<http://www.Arduino.cc/es/>

(Abril 2013)

Natera, A. 2006. **Usos del rayo láser en odontología restauradora. Primera parte: Aspectos generales, clasificación, interrelación con los tejidos vivos y precauciones en el uso**

<http://www.ecuaodontologos.com/revistaaorybg/vol4num3/usosdellasera.html>

(Enero 2013)

Norma Oficial Mexicana: NOM-013-STPS-1993. **Relativa a las condiciones de seguridad e higiene en los centros de trabajo donde se generen radiaciones electromagnéticas no ionizantes**

<http://www.ucoi.mx/dgrh/index.php?id=31>

(Diciembre 2013)

Optitrack. Cámara Optitrack SLIM: V100

<http://www.naturalpoint.com/optitrack/products/slim-v100/features.html>

(Febrero 2013)

ReacTIVision 1.4. **A toolkit for tangible multi-touch surfaces**

<http://reactivision.sourceforge.net/>

(Febrero 2013)

Hrvoje Benko Screens, D. Wilson Andrew, Baudisch Patrick, **Precise Selection Techniques for Multi-Touch**

[http://research.microsoft.com/en-](http://research.microsoft.com/en-us/um/people/awilson/publications/BenkoCHI2006/Benko-CHI06_final.pdf)

[us/um/people/awilson/publications/BenkoCHI2006/Benko-CHI06_final.pdf](http://research.microsoft.com/en-us/um/people/awilson/publications/BenkoCHI2006/Benko-CHI06_final.pdf)

(Abril 2013)

Sergio Escalada Beltrán, Javier Cantón Ferrero y Juliet R. Moreiro Backhop.

Multitouch. Escuela técnica superior de ingeniería informática. Universidad de Sevilla.

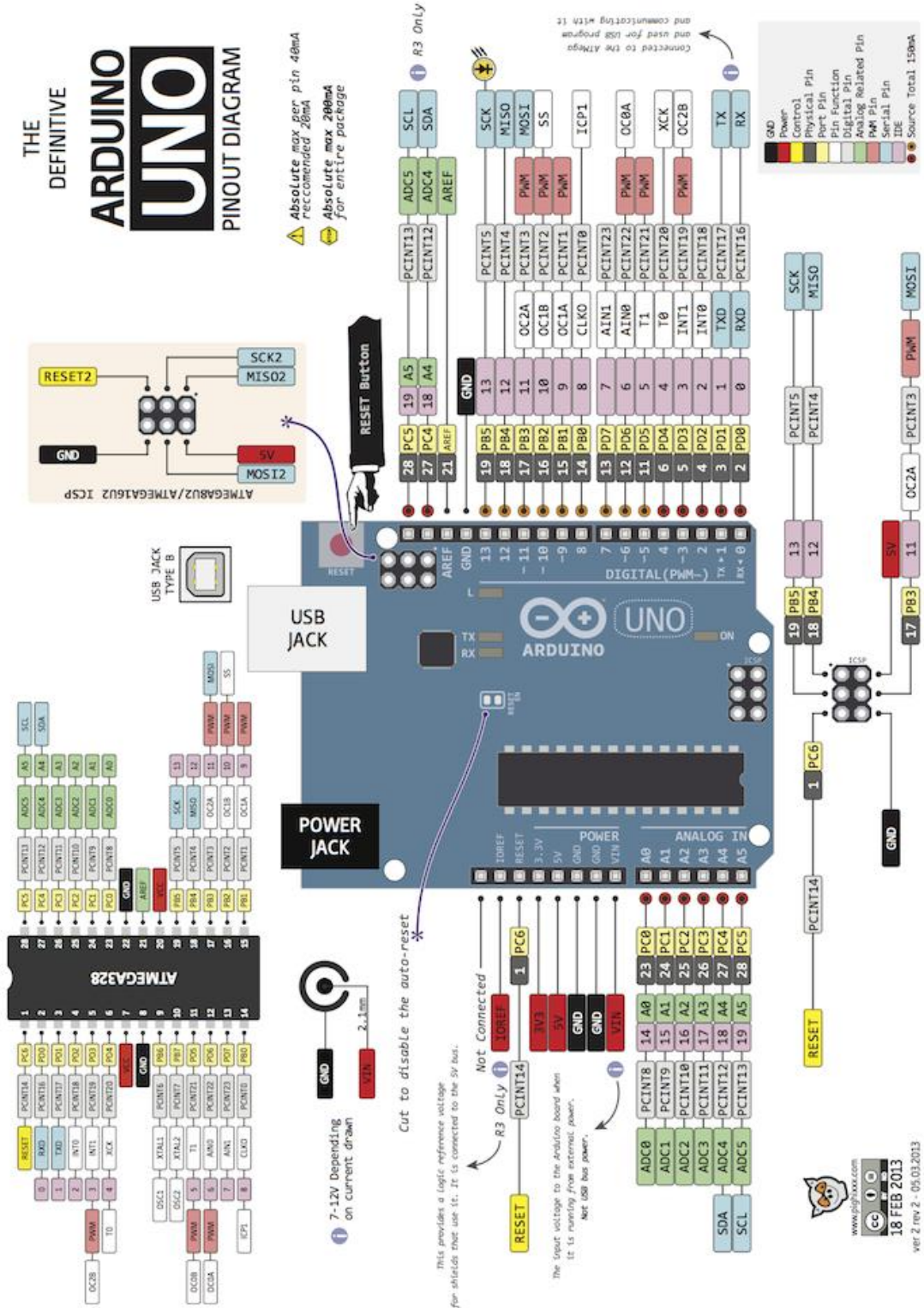
(Abril 2013)

Moeller Jon, Hamilton Bill, **ZeroTouch: A Zero-Thickness Optical, Multi-Touch Force Field**, CHI 2011, May 7–12, 2011, Vancouver, BC, Canada. ACM 978-1-4503-0268-5/11/05

<http://ecologylab.net/research/publications/moeller-zerotouch.pdf>

(Abril 2013)

Anexos.



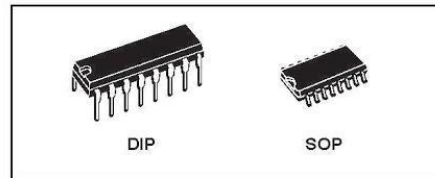
<http://forum.arduino.cc/index.php/topic,146315.0.html>



HCF4051B

SINGLE 8-CHANNEL ANALOG MULTIPLEXER/DEMULTIPLEXER

- LOW "ON" RESISTANCE : 125Ω (Typ.)
OVER 15V p.p SIGNAL-INPUT RANGE FOR
 $V_{DD} - V_{EE} = 15V$
- HIGH "OFF" RESISTANCE : CHANNEL
LEAKAGE $\pm 100pA$ (Typ.) at $V_{DD} - V_{EE} = 18V$
- BINARY ADDRESS DECODING ON CHIP
- HIGH DEGREE OF LINEARITY : < 0.5%
DISTORTION TYP. at $f_{IS} = 1KHz$, $V_{IS} = 5 V_{pp}$,
 $V_{DD} - V_{SS} \geq 10V$, $R_L = 10K\Omega$
- VERY LOW QUIESCENT POWER
DISSIPATION UNDER ALL DIGITAL
CONTROL INPUT AND SUPPLY
CONDITIONS : 0.2 μW (Typ.)
at $V_{DD} - V_{SS} = V_{DD} - V_{EE} = 10V$
- MATCHED SWITCH CHARACTERISTICS :
 $R_{ON} = 5\Omega$ (Typ.) FOR $V_{DD} - V_{EE} = 15V$
- WIDE RANGE OF DIGITAL AND ANALOG
SIGNAL LEVELS : DIGITAL 3 to 20,
ANALOG TO 20V p.p.
- QUIESCENT CURRENT SPECIFIED UP TO
20V
- 5V, 10V AND 15V PARAMETRIC RATINGS
- INPUT LEAKAGE CURRENT
 $I_I = 100nA$ (MAX) AT $V_{DD} = 18V$ $T_A = 25^\circ C$
- 100% TESTED FOR QUIESCENT CURRENT
- MEETS ALL REQUIREMENTS OF JEDEC
JESD13B " STANDARD SPECIFICATIONS
FOR DESCRIPTION OF B SERIES CMOS
DEVICES"



ORDER CODES

PACKAGE	TUBE	T & R
DIP	HCF4051BEY	
SOP	HCF4051BM1	HCF4051M013TR

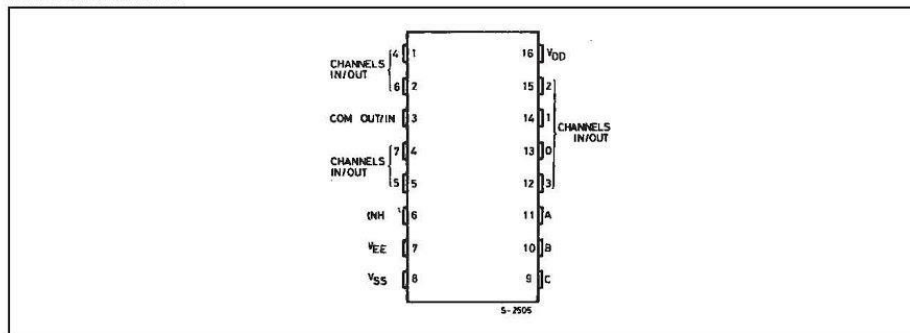
DESCRIPTION

The HCF4051B is a monolithic integrated circuit fabricated in Metal Oxide Semiconductor technology available in DIP and SOP packages.

The HCF4051B analog multiplexer/demultiplexer is a digitally controlled analog switch having low ON impedance and very low OFF leakage current. This multiplexer circuit dissipate extremely low quiescent power over the full $V_{DD} - V_{SS}$ and $V_{DD} - V_{EE}$ supply voltage range, independent of the logic state of the control signals.

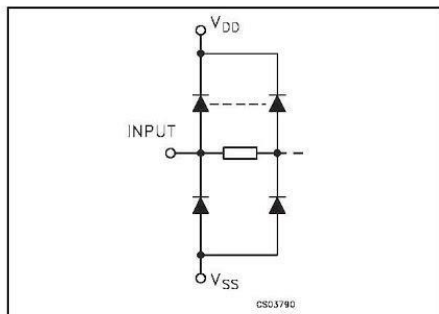
When a logic "1" is present at the inhibit input terminal all channel are off. This device is a single 8-channel multiplexer having three binary control inputs, A, B, and C, and an inhibit input. The three binary signals select 1 of 8 channels to be turned on, and connect one of the 8 inputs to the output.

PIN CONNECTION



HCF4051B

INPUT EQUIVALENT CIRCUIT



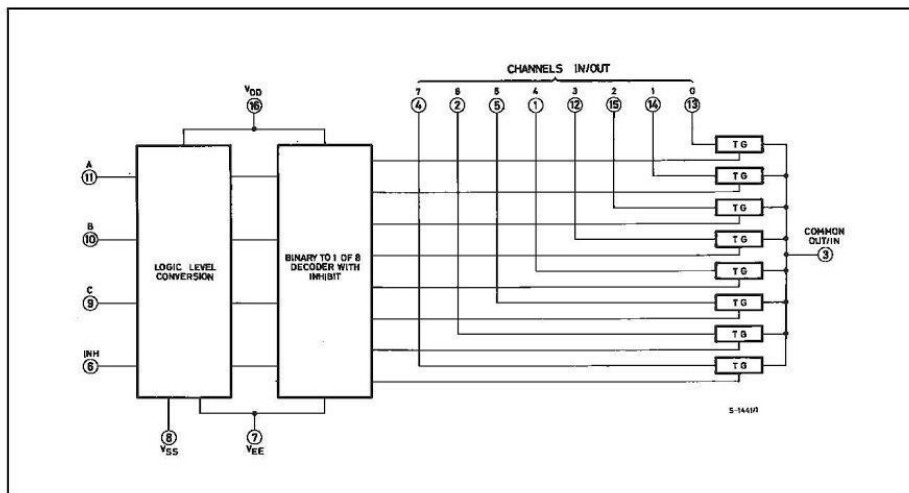
PIN DESCRIPTION

PIN No	SYMBOL	NAME AND FUNCTION
11, 10, 9	A, B, C	Binary Control Inputs
6	INH	Inhibit Inputs
13, 14, 15, 12, 1, 5, 2, 4	0 to 7 CHANNEL IN/OUT	Independent inputs/outputs
3	COM OUT/IN	Common Output/Input
7	V _{EE}	Supply Voltage
8	V _{SS}	Negative Supply Voltage
16	V _{DD}	Positive Supply Voltage

TRUTH TABLE

INPUT STATES				"ON" CHANNEL (S)
INHIBIT	C	B	A	
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	X	X	X	NONE

FUNCTIONAL DIAGRAM



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_{DD}	Supply Voltage	-0.5 to +22	V
V_I	DC Input Voltage	-0.5 to $V_{DD} + 0.5$	V
I_I	DC Input Current	± 10	mA
P_D	Power Dissipation per Package	200	mW
	Power Dissipation per Output Transistor	100	mW
T_{op}	Operating Temperature	-55 to +125	$^{\circ}\text{C}$
T_{stg}	Storage Temperature	-65 to +150	$^{\circ}\text{C}$

Absolute Maximum Ratings are those values beyond which damage to the device may occur. Functional operation under these conditions is not implied.

All voltage values are referred to V_{SS} pin voltage.

RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter	Value	Unit
V_{DD}	Supply Voltage	3 to 20	V
V_I	Input Voltage	0 to V_{DD}	V
T_{op}	Operating Temperature	-55 to 125	$^{\circ}\text{C}$

Technical Data Sheet**5mm Phototransistor T-1 3/4**

Features

- Fast response time
- High photo sensitivity
- Pb free

Descriptions

- PT331C is a high speed and high sensitive silicon NPN epitaxial planar phototransistor in a standard 5Φ package. Due to its water clear epoxy the device is sensitive to visible and near infrared radiation.

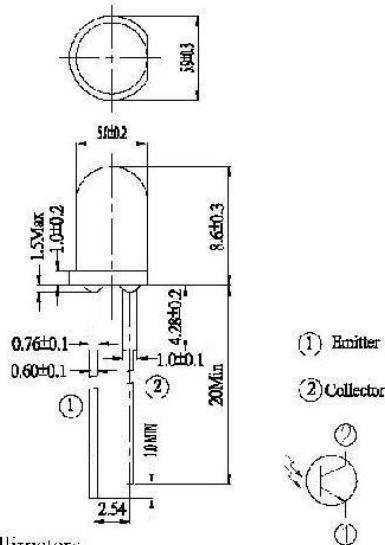
PT331C**Applications**

- Infrared applied system
- Floppy disk drive
- Optoelectronic switch

Device Selection Guide

LED Part No.	Chip	Lens Color
	Material	
PT	Silicon	Water Clear

Package Dimensions



Notes: 1.All dimensions are in millimeters
 2.Tolerances unless dimensions $\pm 0.25\text{mm}$

Absolute Maximum Ratings (Ta=25°C)

Parameter	Symbol	Rating	Units
Collector-Emitter Voltage	V_{CEC}	30	V
Emitter-Collector-Voltage	V_{ECC}	5	V
Collector Current	I_C	20	mA
Operating Temperature	T_{opr}	-25 ~ +85°C	°C
Storage Temperature	T_{stg}	-40 ~ +185°C	°C
Lead Soldering Temperature	T_{sol}	260	°C
Power Dissipation at (or below) 25°C Free Air Temperature	P_c	75	mW

Notes: *1: Soldering time ≤ 5 seconds.

Electro-Optical Characteristics (Ta=25°C)

Parameter	Symbol	Condition	Min.	Typ.	Max.	Units
Collector – Emitter Breakdown Voltage	BV_{CEC}	$I_C=100 \mu A$ $E_e=0mW/cm^2$	30	---	---	V
Emitter-Collector Breakdown Voltage	BV_{EC}	$I_E=100 \mu A$ $E_c=0mW/cm^2$	5	---	---	V
Collector-Emitter Saturation Voltage	$V_{CE(sat)}$	$I_C=2mA$ $E_e=1mW/cm^2$	---	---	0.4	V
Rise Time	t_r	$V_{OL}=5V$ $I_C=1mA$	---	15	---	μS
Fall Time	t_f	$RL=1000\Omega$	---	15	---	
Collector Dark Current	I_{CEO}	$E_e=0mW/cm^2$ $V_{CE}=20V$	---	---	100	nA
On State Collector Current	$I_{C(on)}$	$E_e=1mW/cm^2$ $V_{CE}=5V$	0.7	2.5	---	mA
Wavelength of Peak Sensitivity	λ_p	---	---	940	---	nm
Range of Spectral Bandwidth	$\lambda_{0.5}$	---	---	400-1100	---	nm

Rankings

Parameter	Symbol	Min	Max	Unit	Test Condition
G	$I_{C(on)}$	0.70	1.90	mA	$V_{CE}=5V$ $E_e=1mW/cm^2$
H		1.14	2.60		
J		1.77	3.61		
K		2.67	5.07		
L		4.18	7.07		



BC337-25
BC337-40

SMALL SIGNAL NPN TRANSISTORS

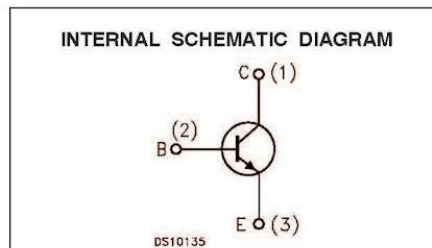
PRELIMINARY DATA

Type	Marking
BC337-25	BC337-25
BC337-40	BC337-40

- SILICON EPITAXIAL PLANAR NPN TRANSISTORS
- TO-92 PACKAGE SUITABLE FOR THROUGH-HOLE PCB ASSEMBLY
- THE PNP COMPLEMENTARY TYPES ARE BC327-25 AND BC327-40 RESPECTIVELY

APPLICATIONS

- WELL SUITABLE FOR TV AND HOME APPLIANCE EQUIPMENT
- SMALL LOAD SWITCH TRANSISTORS WITH HIGH GAIN AND LOW SATURATION VOLTAGE



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_{CB0}	Collector-Base Voltage ($I_E = 0$)	50	V
V_{CE0}	Collector-Emitter Voltage ($I_B = 0$)	45	V
V_{EB0}	Emitter-Base Voltage ($I_C = 0$)	5	V
I_C	Collector Current	0.5	A
I_{CM}	Collector Peak Current	1	A
P_{tot}	Total Dissipation at $T_C = 25^\circ\text{C}$	625	mW
T_{sig}	Storage Temperature	-65 to 150	$^\circ\text{C}$
T_j	Max. Operating Junction Temperature	150	$^\circ\text{C}$

June 2002

1/4

BC337-25 / BC337-40

THERMAL DATA

$R_{th(j-amb)}$	Thermal Resistance Junction-Ambient	Max	200	$^{\circ}\text{C}/\text{W}$
$R_{th(j-case)}$	Thermal Resistance Junction-Case	Max	83.3	$^{\circ}\text{C}/\text{W}$

ELECTRICAL CHARACTERISTICS ($T_{case} = 25^{\circ}\text{C}$ unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
I_{CBO}	Collector Cut-off Current ($I_E = 0$)	$V_{CB} = 20\text{ V}$ $V_{CB} = 20\text{ V}$ $T_C = 150^{\circ}\text{C}$			100 5	nA μA
I_{EBO}	Emitter Cut-off Current ($I_C = 0$)	$V_{EB} = 5\text{ V}$			100	nA
$V_{(BR)CBO}$	Collector-Base Breakdown Voltage ($I_E = 0$)	$I_C = 10\ \mu\text{A}$	50			V
$V_{(BR)CEO}^*$	Collector-Emitter Breakdown Voltage ($I_B = 0$)	$I_C = 10\text{ mA}$	45			V
$V_{(BR)EBO}$	Emitter-Base Breakdown Voltage ($I_C = 0$)	$I_E = 10\ \mu\text{A}$	5			V
$V_{CE(sat)}^*$	Collector-Emitter Saturation Voltage	$I_C = 500\text{ mA}$ $I_B = 50\text{ mA}$			0.7	V
$V_{BE(on)}^*$	Base-Emitter On Voltage	$I_C = 500\text{ mA}$ $V_{CE} = 1\text{ V}$			1.2	V
h_{FE}^*	DC Current Gain	$I_C = 100\text{ mA}$ for BC337-25 for BC337-40 $V_{CE} = 1\text{ V}$	160 250		400 600	
f_T	Transition Frequency	$I_C = 10\text{ mA}$ $V_{CE} = 5\text{ V}$ $f = 100\text{ MHz}$	100			MHz
C_{CBO}	Collector-Base Capacitance	$I_E = 0$ $V_{CB} = 10\text{ V}$ $f = 1\text{ MHz}$		5		pF

* Pulsed: Pulse duration = 300 μs , duty cycle $\leq 2\%$

TO-92 MECHANICAL DATA

DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A	4.32		4.95	0.170		0.195
b	0.96		0.51	0.014		0.020
D	4.45		4.95	0.175		0.194
E	3.30		3.94	0.130		0.155
e	2.41		2.67	0.095		0.105
e1	1.14		1.40	0.045		0.055
L	12.70		15.49	0.500		0.609
R	2.16		2.41	0.085		0.094
S1	1.14		1.52	0.045		0.059
W	0.41		0.56	0.016		0.022
V	4 degree		6 degree	4 degree		6 degree

