



UNIVERSIDAD TÉCNICA DE COTOPAXI
FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS
CARRERA DE INGENIERÍA ELECTROMECAÁNICA

PROPUESTA TECNOLÓGICA

TEMA: “DESARROLLO DE UN DISPOSITIVO DE IDENTIFICACIÓN DE COLORES EMPLEANDO REDES NEURONALES Y HARDWARE LIBRE, ORIENTADO A LA AYUDA DE PERSONAS DALTÓNICAS”

Proyecto de Titulación presentado previa a la obtención del Título de Ingeniero
Electromecánico

Autores:

Guambi Manitio Rubén Darío
Zambrano Plusas Sandro Joel

Tutor:

Ing. Byron Paúl Corrales Bastidas M.Sc.

Latacunga – Ecuador
Septiembre - 2020



DECLARACIÓN DE AUTORÍA

Nosotros, **Guambi Manitio Rubén Darío** y **Zambrano Pluas Sandro Joel** declaramos ser autores de la presente Propuesta Tecnológica:

“Desarrollo de un dispositivo de identificación de colores empleando redes neuronales y hardware libre, orientado a la ayuda de personas daltónicas”, siendo el Ing. Byron Paúl Corrales Bastidas M.Sc. tutor del presente trabajo; y eximimos expresamente a la Universidad Técnica de Cotopaxi y a sus representantes legales de posibles reclamos o acciones legales.

Además, certificamos que las ideas, conceptos, procedimientos y resultados vertidos en el presente trabajo investigativo, son de nuestra exclusiva responsabilidad.

Guambi Manitio Rubén Darío

CC.: 150078808-6

Zambrano Pluas Sandro Joel

CC.: 172404339-1



AVAL DEL TUTOR DE PROYECTO D E TITULACIÓN

En calidad de Tutor del Trabajo de Investigación sobre el título: “**DESARROLLO DE UN DISPOSITIVO DE IDENTIFICACIÓN DE COLORES EMPLEANDO REDES NEURONALES Y HARDWARE LIBRE, ORIENTADO A LA AYUDA DE PERSONAS DALTÓNICAS**”, de Guambi Manitio Rubén Darío y Zambrano Pluas Sandro Joel, de la Carrera de Ingeniería Electromecánica, considero que dicho Informe Investigativo cumple con los requerimientos metodológicos y aportes científico-técnicos suficientes para ser sometidos a la evaluación del Tribunal de Validación de Proyecto que el Consejo Directivo de la Facultad de Ciencias de la Ingeniería y Aplicadas de la Universidad Técnica de Cotopaxi designe, para su correspondiente estudio y calificación.

Latacunga, Septiembre del 2020

Ing. Byron Paúl Corrales Bastidas M.Sc.



APROBACIÓN DEL TRIBUNAL DE TITULACIÓN

En calidad de Tribunal de Lectores, aprueban el presente Informe de Investigación de acuerdo a las disposiciones reglamentarias emitidas por la Universidad Técnica de Cotopaxi, y por la Facultad de Ciencias de la Ingeniería y Aplicadas; por cuanto, los postulantes: Guambi Manitio Rubén Darío con C.C. 150078808-6 y Zambrano Pluas Sandro Joel con C.C 172404339-1 con el título de Proyecto de titulación:

“DESARROLLO DE UN DISPOSITIVO DE IDENTIFICACIÓN DE COLORES EMPLEANDO REDES NEURONALES Y HARDWARE LIBRE, ORIENTADO A LA AYUDA DE PERSONAS DALTÓNICAS”, han considerado las recomendaciones emitidas oportunamente y reúne los méritos suficientes para ser sometido al acto de Sustentación de Proyecto.

Por lo antes expuesto, se autoriza realizar los empastados correspondientes, según la normativa institucional.

Latacunga, septiembre del 2020

Para constancia firman:

Lector 1

Nombre: Ing. Verónica Paulina Freire
Andrade
CC: 050205622-9

Lector 2

Nombre: Ing. Edwin Homero Moreano
Martínez
CC: 050260750-0

Lector 3

Nombre: Ing. Jefferson Porras Reyes
CC: 070440044-9

AGRADECIMIENTO

Me gustaría agradecer principalmente, a Dios por darme la fuerza y sabiduría para poder culminar este proyecto. A la Universidad Técnica de Cotopaxi y a todos sus docentes por el conocimiento impartido en las aulas y laboratorios, a mi familia, personas y colegas que me han apoyado durante este proceso de investigación. También agradezco a mis padres Clara y Carlos que me han ayudado y apoyado en el transcurso de mi carrera universitaria, a mi querida Anita por su apoyo incondicional y emocional, a nuestro tutor Ing. Byron Paúl Corrales Bastidas M.Sc. por haberme orientado en todos los momentos que necesité sus consejos.

DARIO

AGRADECIMIENTO

Quiero agradecer, a Dios, a mi hermanito que está en el cielo, desde ahí me ha dado toda la fuerza que he necesitado para culminar mi Carrera, también agradecer el esfuerzo que han realizado mis padres por darme lo mejor y seguir apoyándome en todo. A la Universidad Técnica de Cotopaxi por ser el lugar donde he podido aprender muchas cosas, a sus docentes que me han enseñado con paciencia y sabiduría. También agradecer a todos mis compañeros que a lo largo de la carrera nos hemos apoyado en cada dificultad que se nos ha presentado, agradecer de una manera muy especial a todos los docentes de Ingeniería Electromecánica y su apoyo incondicional a nuestro tutor Ing. Byron Paúl Corrales Bastidas M.Sc. por habernos orientado a culminar este proyecto.

SANDRO

DEDICATORIA

Este trabajo de investigación va dedicado con todo mi cariño, a mi madre Clara por darme la vida y enseñarme muchos valores, a mi padre Carlos por enseñarme el concepto honradez y a toda mi familia en general por comprender y ayudarme en mi carrera profesional. También dedico este trabajo a todos mis amigos que me ayudaron y fueron testigos del arduo trabajo y tiempo empleado en esta investigación.

DARIO

DEDICATORIA

Este trabajo se lo dedico a mi hermanito que desde el cielo supo cuidarme y guiarme en cada paso, a mi familia en especial a mis padres ya que ellos creyeron en mí, fueron y serán un pilar fundamental para lograr cada objetivo que me he planteado y seguir motivándome, también a una persona muy especial que al final de mi carrera me apoyo en todo momento. A cada uno de mis compañeros que ya no están en este mundo. También dedico este trabajo a todos mis amigos que han sido testigos de todo el esfuerzo realizado a lo largo de toda mi formación académica

SANDRO

ÍNDICE DE CONTENIDO

DECLARACIÓN DE AUTORÍA	i
AVAL DEL TUTOR DE PROYECTO D E TITULACIÓN	ii
APROBACIÓN DEL TRIBUNAL DE TITULACIÓN.....	iii
AGRADECIMIENTO	iv
DEDICATORIA.....	vi
ÍNDICE DE CONTENIDO	viii
ÍNDICE DE FIGURAS	xii
ÍNDICE DE TABLAS	xiv
RESUMEN	xv
ABSTRACT	xvi
AVAL DE TRADUCCIÓN.....	xvii
1. INFORMACIÓN BÁSICA	1
2. DISEÑO INVESTIGATIVO DE LA PROPUESTA TECNOLÓGICA.....	2
2.1. TÍTULO DE LA PROPUESTA TECNOLÓGICA	2
2.2. TIPO DE PROPUESTA ALCANCE	2
2.3. ÁREA DEL CONOCIMIENTO	2
2.4. SINOPSIS DE LA PROPUESTA TECNOLÓGICA	2
2.5. OBJETO DE ESTUDIO Y CAMPO DE ACCIÓN	3
2.5.1. Objeto de estudio	3
2.5.2. Campo de acción	3
2.6. SITUACIÓN PROBLÉMICA Y PROBLEMA	3
2.6.1. Situación problemática	3
2.6.2. Matriz causa efecto.....	4
2.6.3. Problema.....	4
2.7. HIPÓTESIS O FORMULACIÓN DE PREGUNTAS DIRECTRICES	4
2.8. OBJETIVOS	4
2.8.1. Objetivo general	4
2.8.2. Objetivos específicos.....	4
2.9. DESCRIPCIÓN DE LAS ACTIVIDADES Y TAREAS PROPUESTAS CON OBJETIVOS ESTABLECIDOS	5
3. FUNDAMENTACIÓN CIENTÍFICO TÉCNICA.....	6
3.1. Análisis de trabajo precedentes.....	6
3.2. El daltonismo o ceguera de color.....	7

3.2.1.	Causas del daltonismo	7
3.2.2.	Problemas asociados al daltonismo	8
3.2.3.	Tipos de daltonismo.....	8
3.2.4.	Pruebas de daltonismo	9
3.2.4.1.	Test Ishihara	9
3.2.4.2.	Test Hardy-Rand-Rittler	10
3.2.5.	Tratamiento de daltonismo	10
3.2.5.1.	Tratamiento de daltonismo infantil.....	10
3.3.	Redes neuronales artificiales	11
3.3.1.	Características de una red neuronal	11
3.3.1.1.	La neurona Artificial	11
3.3.1.2.	Estado de activación	12
3.3.1.3.	Entradas a la neurona.....	12
3.3.1.4.	Función de propagación.....	12
3.3.1.5.	Función de salida o de Transferencia	14
3.3.2.	Tipos de redes neuronales.....	15
3.3.2.1.	El perceptrón simple.....	15
3.3.2.2.	Adaline.....	16
3.3.2.3.	El perceptrón multicapa.....	17
3.3.3.	Aprendizaje de las redes neuronales artificiales.....	17
3.3.3.1.	Aprendizaje supervisado.....	18
3.3.3.2.	Aprendizaje por corrección de error.....	18
3.3.4.	Algoritmos de aprendizaje	19
3.3.4.1.	Algoritmo de aprendizaje Backpropagation	19
3.3.5.	Aplicaciones de una red neuronal.....	20
3.4.	Hardware Libre	21
3.4.1.	Herramientas de diseño de Hardware libre.....	21
3.4.2.	Ventajas y desventajas hardware libre.....	21
3.5.	Arduino	22
3.5.1.	ADC Arduino	22
3.5.2.	Tipos de Arduino	23
3.5.2.1.	Arduino UNO	23
3.5.2.2.	Arduino DUE.....	23
3.5.2.3.	Arduino Leonardo.....	23

3.5.2.4.	Arduino MEGA 2560	23
3.5.2.5.	Arduino Nano	24
3.5.3.	Programación en Arduino.....	24
3.6.	Instrumentación Electrónica	25
3.6.1.	Funciones de un sistema de medida	26
3.6.2.	Fotorresistencia LDR.....	26
3.6.2.1.	Acondicionamiento de la señal.....	27
3.6.3.	Tecnología diodo Led RGB.....	28
3.7.	Repetitividad de ejecución.	29
4.	METODOLOGÍA Y DISEÑO EXPERIMENTAL	29
4.1.	Definición de variables entrada y salida	29
4.1.1.	Variable entrada.....	29
4.1.2.	Variable salida	30
4.2.	Diseño experimental	30
4.2.1.	Diseño electrónico	31
4.2.1.1.	Componentes principales.....	31
4.2.2.	Programación.....	35
4.2.2.1.	Programación red neuronal.....	35
4.2.2.2.	Programación en software Arduino	40
4.2.3.	Diseño y construcción de la caja del dispositivo	44
4.2.4.	Repetitividad de ejecución.....	44
5.	ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS	45
5.1.	Resultados de programación red neuronal-Arduino	46
5.1.1.	Entrenamiento de la red neuronal.....	46
5.1.2.	Valores de salidas de la red neuronal	46
5.1.3.	Identificación de los colores dispositivo	48
5.1.3.1.	Descripción del dispositivo	48
5.3.2.2.	Validación del funcionamiento del dispositivo	49
5.2.	Resultados diseño y construcción de la caja del dispositivo	52
5.	PRESUPUESTO Y ANÁLISIS DE IMPACTOS	53
5.3.	Presupuesto	53
5.3.2.	Materiales electrónicos	53
5.3.3.	Gastos de estructura y diseño de ingeniería del dispositivo	54
5.3.4.	Gastos varios.....	54

5.3.5.	Gastos directos totales	55
5.3.6.	Gastos indirectos totales	55
5.3.7.	Gastos totales del dispositivo	55
5.3.2.	Cálculo del TIR y VAN.....	55
5.4.	Análisis de impacto.....	56
5.4.2.	Impacto tecnológico	56
5.4.3.	Impacto social.....	56
5.5.	Beneficiarios del proyecto	56
5.5.2.	Beneficiarios directos:	56
5.5.3.	Beneficiarios indirectos:	56
6.	CONCLUSIONES Y RECOMENDACIONES	57
6.3.	Conclusiones.....	57
6.4.	Recomendaciones	57
7.	BIBLIOGRAFÍA.....	58
	ANEXOS	61

ÍNDICE DE FIGURAS

Figura 3. 1 Herencia del daltonismo.....	7
Figura 3. 2 Estructura anatómica del ojo humano.....	8
Figura 3. 3 Normal/Visión Protanopia y Deuteranopia.....	9
Figura 3. 4 Test de prueba de daltonismo(Ishihara).....	9
Figura 3. 5 Test de prueba de daltonismo láminas (Hardy Rand Rittler).....	10
Figura 3. 6 ColorADD código para personas daltónicas.....	11
Figura 3. 7 Estructura general de una neurona.....	11
Figura 3. 8 Pesos W_{ij}	13
Figura 3. 9 Función Escalón.....	14
Figura 3. 10 Función lineal y mixta.....	14
Figura 3. 11 Función Sigmoidal.....	15
Figura 3. 12 Función Gaussiana.....	15
Figura 3. 13 Perceptrón Simple.....	16
Figura 3. 14 Estructura de la red Adaline.....	17
Figura 3. 15 Perceptrón multicapa.....	17
Figura 3. 16 Estructura de un aprendizaje con corrección de error.....	18
Figura 3. 17 Tipos de señales.....	25
Figura 3. 18 Funciones de un sistema de medida.....	26
Figura 3. 19 Fotorresistencia LDR.....	26
Figura 3. 20 Circuito de acondicionamiento para un detector de oscuridad.....	27
Figura 3. 21 Circuito de acondicionamiento para un detector de luz.....	28
Figura 3. 22 Led RGB de 4 pines.....	28
Figura 4. 1 Arduino Mega 2560.....	31
Figura 4. 2 Diagrama de sensor.....	33
Figura 4. 3 Diagrama de elementos electrónico (tarjeta SD y parlante).....	34
Figura 4. 4 Programación red neuronal.....	38
Figura 4. 5 Estructura red neuronal.....	38
Figura 4. 6 Bias y pesos de la red neuronal.....	39
Figura 4. 7 Programación software arduino.....	41
Figura 4. 8 Flujograma de la programación de software.....	43
Figura 5. 1 Transferencia de datos a software matemático.....	40

Figura 5. 2 Entrenamiento de la red.	46
Figura 5. 3 Comparación de datos obtenido con datos teóricos.	47
Figura 5. 4 Dispositivo identificador de colores.	48
Figura 5. 5 Validación de la pruebas del dispositivo.....	52
Figura 5. 6 Caja del dispositivo	53
Figura 5. 7 TIR y VAN.....	55

ÍNDICE DE TABLAS

Tabla 2. 1	Sistema de tareas por objetivos.	5
Tabla 3. 1	Modelos de redes neuronales artificiales con aprendizaje supervisado	18
Tabla 3. 2	Los voltajes Max de operación y resolución de cada placa Arduino.	22
Tabla 3. 3	Estructura y funciones de lenguaje de programación de arduino	24
Tabla 4. 1	Operacionalización de variables de entrada.	30
Tabla 4. 2	Operacionalización de variables de salida.	30
Tabla 4. 3	Procedimiento experimental.	30
Tabla 4. 4	Características de la fotorresistencia.	32
Tabla 4. 5	Características diodo led RGB.	32
Tabla 4. 6	Acondicionamiento de la señal.	33
Tabla 4. 7	Acondicionamiento de los componentes.	34
Tabla 4. 8	Diseño de la red neuronal.	35
Tabla 4. 9	Combinación de colores en código RGB.	35
Tabla 4. 10	Combinación de colores en código binario.	36
Tabla 4. 11	Asignación de valores a los colores.	36
Tabla 4. 12	Muestra de datos de entrada y salidas de la red neuronal.	37
Tabla 4. 13	Transferencia de datos de la red neuronal.	38
Tabla 4. 14	Transferencia de los datos de software matemático - programación de Arduino.	40
Tabla 4. 15	Programación del dispositivo.	41
Tabla 4. 16	Diseño y construcción de la caja del dispositivo.	44
Tabla 5. 1	Análisis y resultados.	45
Tabla 5. 2	Validación de resultados de la red neuronal.	47
Tabla 5. 4	Validación del dispositivo mediante programación de la red neuronal.	49
Tabla 5. 5	Validación del dispositivo identificador de colores mediante pruebas.	50
Tabla 6. 1	Materiales Electrónicos.	53
Tabla 6. 2	Estructura del dispositivo y diseño de ingeniería.	54
Tabla 6. 3	Gastos varios.	54
Tabla 6. 4	Gastos directos totales.	55
Tabla 6. 5	Gastos indirectos totales.	55
Tabla 6. 6	Gastos totales.	55

UNIVERSIDAD TÉCNICA DE COTOPAXI

FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS

TÍTULO: “DESARROLLO DE UN DISPOSITIVO DE IDENTIFICACIÓN DE COLORES EMPLEANDO REDES NEURONALES Y HARDWARE LIBRE, ORIENTADO A LA AYUDA DE PERSONAS DALTÓNICAS”

Autores:

Rubén Darío Guambi Manitio

Sandro Joel Zambrano Plusas

RESUMEN

La palabra daltonismo proviene del inglés “daltonism”, derivado del apellido del físico y químico inglés John Dalton (1766-1844), que padecía esta enfermedad y fue el primero en describirla.

La problemática social que viven las personas que sufren de daltonismo es de orden genético y no tiene cura como tal, esto incita al rechazo, vergüenza y sentimientos de incompetencia al no poder identificar los colores de manera normal.

El dispositivo pretende resolver esta problemática basándose, en la utilización de nuevas técnicas de control, al emplear redes neuronales en la que se ingresan las características de los colores en código RGB (Rojo, Verde y Azul) y así de esta manera se pondera los valores a código binario y por medio del entrenamiento de la red neuronal se logra mejores resultados de aprendizaje y obtener la salida deseada. Mediante la programación que se realiza en Arduino y con la ayuda del diodo RGB (Rojo, Verde y Azul) se almacenan los valores obtenidos por la fotorresistencia y estos datos se comparan con los de la red neuronal, por lo cual si entran en el rango establecido por la red neuronal se imprimirá el nombre del color en la pantalla LCD (Pantalla de Cristal Líquido) y una bocina ayudará de forma auditiva a la identificación de los colores. El prototipo identificador de colores posee muy pocas limitaciones en comparación de otros dispositivos que se enfocan a la ayuda de personas daltónicas. Para comprobar el funcionamiento del prototipo se establecieron la identificación de varios colores dando como resultado 96% de efectividad, además se realizó entrevistas con personas daltónicas, resaltando la inclusión que tiene el dispositivo en lo que señala la Constitución de la República del Ecuador en la Sección Sexta del Art 47 de Personas con Discapacidad.

Palabras claves: Daltonismo, redes neuronales, hardware libre, identificación de colores.

COTOPAXI TECHNICAL UNIVERSITY
FACULTY ENGINEERING AND APPLIED SCIENCES

TOPIC: “DEVELOPMENT OF A COLOR IDENTIFICATION DEVICE USING NEURAL NETWORKS AND FREE HARDWARE, ORIENTED TO THE HELP OF BLIND PERSONS”

Authors:

Rubén Dario Guambi Manitio

Sandro Joel Zambrano Plusas

ABSTRACT

The word daltonism comes from the English "daltonism", derived from the surname of the English physicist and chemist John Dalton (1766-1844), who suffered from this disease and was the first to describe it.

The social problems experienced by people who suffer from color blindness is genetic and has no cure as such, this encourages rejection, shame and feelings of incompetence to not be able to identify the colors normally.

The device aims to solve this problem based on the use of new control techniques, using neural networks in which the characteristics of the colors are entered in RGB code (Red, Green and Blue) and thus weighs the values to binary code and through training of the neural network is achieved better learning results and obtain the desired output. By means of the programming that is carried out in Arduino and with the help of the RGB diode (Red, Green and Blue) the values obtained by the photoresistor are stored and these data are compared with those of the neuronal network, so if they enter the range established by the neuronal network the name of the color will be printed on the LCD screen (Liquid Crystal Display) and a speaker will help audibly to identify the colors. The color identification prototype has very few limitations compared to other devices that focus on helping colorblind people. In order to verify the functioning of the prototype, the identification of several colors was established, resulting in 96% effectiveness. In addition, interviews were conducted with colorblind people, highlighting the inclusion of the device in what the Constitution of the Republic of Ecuador states in Section Six of Article 47 on Persons with Disabilities.

Keywords: Colorblindness, neural networks, free hardware, color identification.



AVAL DE TRADUCCIÓN

En calidad de Docente del Idioma Inglés del Centro de Idiomas de la Universidad Técnica de Cotopaxi; en forma legal **CERTIFICO** que: La traducción del resumen del proyecto de investigación al Idioma Inglés presentado por los señores Guambi Manitio Ruben Dario, Zambrano Pluas Sandro Joel. Egresados de la Carrera de Ingeniería Electromecánica de la Facultad de Ciencias de la Ingeniería y Aplicadas, cuyo título versa “DESARROLLO DE UN DISPOSITIVO DE IDENTIFICACIÓN DE COLORES EMPLEANDO REDES NEURONALES Y HARDWARE LIBRE, ORIENTADO A LA AYUDA DE PERSONAS DALTÓNICAS” lo realizaron bajo mi supervisión y cumple con una correcta estructura gramatical del Idioma.

Es todo cuanto puedo certificar en honor a la verdad y autorizo los peticionarios hacer uso del presente certificado de la manera ética que estimaren conveniente.

Latacunga, septiembre del 2020

Atentamente,

Nelson Guagchinga, Mg. C.
DOCENTE CENTRO DE IDIOMAS
C.C. 050324641-5

1. INFORMACIÓN BÁSICA

PROPUESTO POR:

Guambi Manitio Ruben Dario.

Zambrano Pluas Sandro Joel.

TEMA APROBADO:

“Desarrollo de un dispositivo de identificación de colores empleando redes neuronales y hardware libre, orientado a la ayuda de personas daltónicas”

CARRERA:

Ingeniería Electromecánica.

DIRECTOR DEL PROYECTO DE TITULACIÓN

Ing. Byron Paúl Corrales Bastidas M.Sc.

EQUIPO DE TRABAJO:

Sr. Ruben Dario Guambi Manitio.

Sr. Sandro Joel Zambrano Pluas

Ing. Byron Paúl Corrales Bastidas M.Sc. (Tutor)

LUGAR DE EJECUCIÓN:

Zona 3, Provincia de Cotopaxi, Cantón Latacunga, Parroquia San Felipe-UTC.

TIEMPO DE DURACIÓN DEL PROYECTO:

6 meses.

FECHA DE ENTREGA:

Septiembre 2020.

EL PROYECTO RECAE SOBRE LA LÍNEA DE INVESTIGACIÓN:

De acuerdo a lo establecido por el departamento de investigación de la Universidad Técnica de Cotopaxi, línea 4 Procesos Industriales.

SUB LÍNEA DE INVESTIGACIÓN DE LA CARRERA:

Automatización, control y protecciones de sistemas electromecánicos.

TIPO DE PROPUESTA TECNOLÓGICA:

Es una propuesta tecnológica que busca desarrollar un dispositivo identificador de colores, empleando nuevas técnicas de control de procesos, además utilizando herramientas de uso libre, para la adquisición de datos.

2. DISEÑO INVESTIGATIVO DE LA PROPUESTA TECNOLÓGICA

2.1. TÍTULO DE LA PROPUESTA TECNOLÓGICA

Desarrollo de un dispositivo de identificación de colores empleando redes neuronales y hardware libre, orientado a la ayuda de personas daltónicas.

2.2. TIPO DE PROPUESTA ALCANCE

- a) **Multipropósito:** El dispositivo está diseñado para la identificación de colores, además de brindar la ayuda a personas daltónicas también se enfoca al desarrollo de nuevas tecnologías en lo que se refiere a inteligencia artificial mediante redes neuronales, que cada vez avanzan de una manera exponencial al desarrollo tecnológico
- b) **Integrador:** Las redes neuronales son un elemento importante de la inteligencia artificial. Es por ello que, al realizar un dispositivo capaz de identificar los colores, principalmente solventando un problema, el bajo costo de fabricación usando elementos de la electrónica convencional.

2.3. ÁREA DEL CONOCIMIENTO

La Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura (UNESCO) considera que el tema de propuesta tecnológica tiende a relacionarse con la ingeniería, industria y construcción.

La UNESCO, en la sub área de Ingeniería y profesiones afines, el cual consta con: Dibujo técnico, mecánica, metalistería, electricidad, electrónica, telecomunicaciones, ingeniería energética y química, mantenimiento de vehículos, topografía.

2.4. SINOPSIS DE LA PROPUESTA TECNOLÓGICA

El dispositivo pretende resolver esta problemática basándose, en la utilización de nuevas técnicas de control, al emplear redes neuronales en la que se ingresan las características de los colores en código RGB (Rojo, Verde y Azul) y así de esta manera se pondera los valores a código binario y por medio del entrenamiento de la red neuronal se logra mejores resultados de aprendizaje y obtener la salida deseada. Mediante la programación que se realiza en Arduino y con la ayuda del diodo RGB (Rojo, Verde y Azul) se almacenan los valores obtenidos por la fotorresistencia y estos datos se comparan con los de la red neuronal, por lo cual si entran en el rango establecido por la red neuronal se imprimirá el nombre del color en la pantalla LCD (Pantalla de Cristal Líquido) y una bocina ayudará de forma auditiva a la identificación de los colores. El prototipo identificador de colores posee muy pocas limitaciones en comparación de otros dispositivos que se enfocan a la ayuda de personas daltónicas. Para comprobar el

funcionamiento del prototipo se establecieron la identificación de varios colores dando como resultado 96% de efectividad, además se realizó entrevistas con personas daltónicas, resaltando la inclusión que tiene el dispositivo en lo que señala la Constitución de la República del Ecuador en la Sección Sexta del Art 47 de Personas con Discapacidad.

2.5. OBJETO DE ESTUDIO Y CAMPO DE ACCIÓN

2.5.1. Objeto de estudio

Dispositivo de identificación de colores empleando redes neuronales.

2.5.2. Campo de acción

Electrónica, Programación, Sistemas de control, Instrumentación y Microcontroladores

2.6. SITUACIÓN PROBLÉMICA Y PROBLEMA

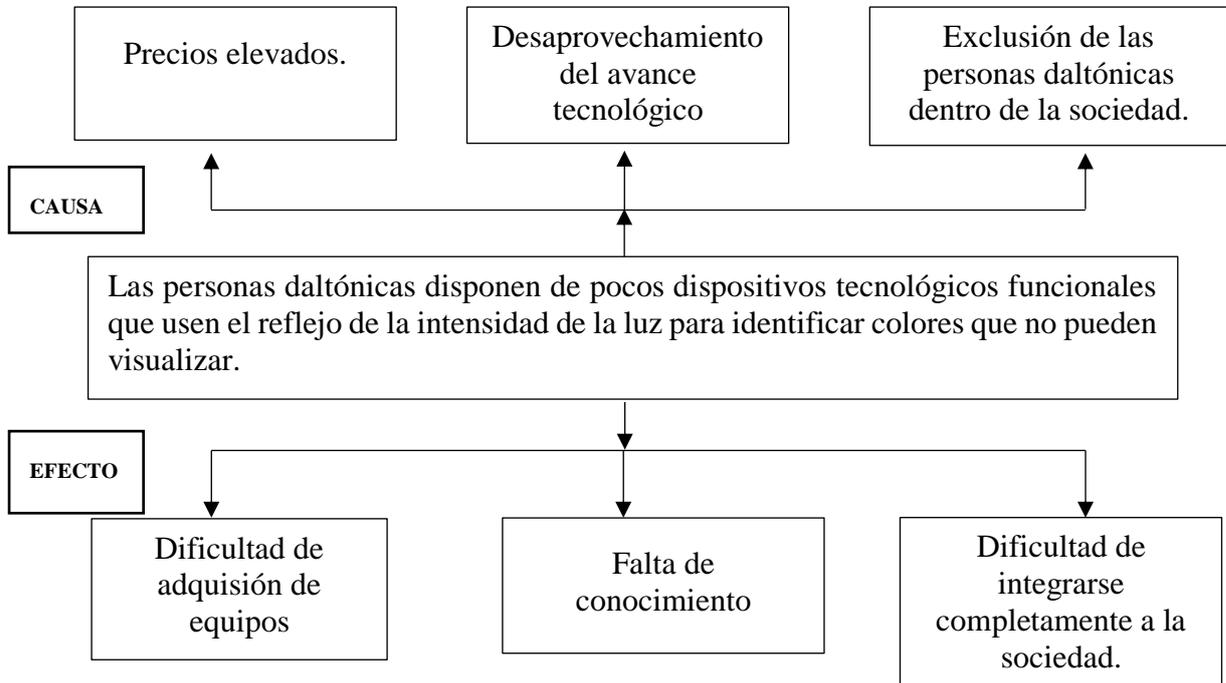
2.6.1. Situación problemática

El daltonismo es la incapacidad de visualizar los colores azul, verde, rojo y amarillo de manera normal. Esta enfermedad se ocasiona cuando existe una patología con los pigmentos en ciertas células nerviosas del ojo que captan el color. Estas células poseen el nombre de conos y están ubicadas en la capa del tejido sensible a la luz que se ubicada en la parte posterior del ojo, llamado retina[1]. Este defecto es genético se puede adquirir incluso por un traumatismo o el uso de algún medicamento[2].

El porcentaje de daltónicos en la población mundial es de aproximadamente un 8% de la población masculina y un 0,5% entre las mujeres. Además, este porcentaje se encuentra en un claro ascenso, especialmente en la población europea, lo que se atribuye a una relajación en la selección natural debido al progreso de la civilización[3]. Por otra parte, en América latina se conoce que uno de cada 12 hombres y una de cada 200 mujeres sufre de daltonismo[2]. En Ecuador el daltonismo no ha tenido relevancia, por lo cual no se han realizado suficientes investigaciones para conocer el número exacto de personas que sufren de esta patología, es así que una de estas investigaciones que se ha realizado han sido a un pequeño grupo de personas que sufren de daltonismo en la región andina del Ecuador. Por lo cual, la falta de dispositivos tecnológicos que ayuden a identificar los colores a personas daltónicas son escasos, se puede considerar que la tecnología de asistencia puede ayudar a las personas que sufren esta afección a mejorar sus habilidades. Debido a este desinterés la mayoría de personas que sufren de esta patología presentan escasa interacción social con el medio, lo cual dificulta su integración completamente con la sociedad, por ende, es necesario el implementar un dispositivo

electrónico que permita ayudar a identificar los colores que este grupo de personas no puede visualizar, y con esto mejorar su calidad de vida.

2.6.2. Matriz causa efecto



2.6.3. Problema

Las personas daltónicas disponen de pocos dispositivos tecnológicos funcionales que usen el reflejo de la intensidad de la luz para identificar colores que no pueden visualizar.

2.7. HIPÓTESIS O FORMULACIÓN DE PREGUNTAS DIRECTRICES

Con el desarrollo de este dispositivo tecnológico funcional utilizando el reflejo de la intensidad de la luz se podrá identificar los colores que las personas daltónicas tienen dificultad en reconocer.

2.8. OBJETIVOS

2.8.1. Objetivo general

Desarrollar un dispositivo tecnológico funcional empleando el reflejo de la intensidad de la luz, redes neuronales y hardware libre orientado a la identificación de colores no visualizados por daltónicos.

2.8.2. Objetivos específicos

- Investigar sobre el daltonismo y la relevancia de cada tipología del mismo en la visualización del color, mediante la búsqueda en artículos científicos, revistas y páginas web.

- Explorar el uso de redes neuronales para acoplarlas al desarrollo de los componentes del dispositivo identificador de colores.
- Implementar un dispositivo de identificación de colores orientado a la ayuda de personas daltónicas.
- Realizar la validación del prototipo mediante la ecuación del tamaño de la muestra, obteniendo la cantidad necesaria para la comprobación del reconocimiento de los colores en las diferentes tonalidades.

2.9. DESCRIPCIÓN DE LAS ACTIVIDADES Y TAREAS PROPUESTAS CON OBJETIVOS ESTABLECIDOS

Tabla 2. 1 Sistema de tareas por objetivos.

OBJETIVOS	ACTIVIDAD	RESULTADOS	MEDIO DE VERIFICACIÓN
Investigar sobre el daltonismo y la relevancia que tiene cada tipología del mismo en la visualización del color, mediante la investigación de artículos científicos, revistas y páginas web.	1. Consulta a un médico.	Tipos de daltonismo.	Fichas, audios e informe de entrevista
	2. Levantamiento de información referente al daltonismo.	Análisis de como el daltonismo afecta a las personas que lo padecen.	Artículos científicos de daltonismo
	3. Consulta de pruebas que se realizan para determinar si una persona sufre de daltonismo.	Test Ishihara, Test Hardy-Rand-Rittler.	Fichas de test Ishihara
	4. Tratamiento para las personas daltónicas.	Gafas o lentes para personas daltonismo.	Artículos científicos de lentes o gafas de daltonismo.
Explorar el uso de redes neuronales para acoplarlos al desarrollo de los componentes del dispositivo identificador de colores.	1. Análisis de los tipos de redes neuronales.	<ul style="list-style-type: none"> • Perceptrón simple. • Perceptrón multicapa. • ADELIN. 	Artículos científicos de redes neuronales.
	2. Identificación de cual tipo de red neuronal se puede relacionar con el dispositivo.	Estructuras de las redes neuronales.	Artículos científicos de redes neuronales.
Implementar un dispositivo de identificación de colores orientado a la ayuda de personas daltónicas	1. Recopilación datos de las características del color.	Datos de características del color.	Tabla de datos de características del color.
	2. Entrenamiento de la red neuronal.	Entrenamiento de la red neuronal.	Programación de entrenamiento.
	3. Diseño del dispositivo identificador de colores.	Dispositivo identificador de colores.	Dispositivo identificador de colores.

Realizar la validación del prototipo mediante la ecuación del tamaño de la muestra, obteniendo la cantidad necesaria para la comprobación del reconocimiento de los colores en las diferentes tonalidades.	1. Verificación del reconocimiento de los colores	Reconocimiento de las diferentes tonalidades de los colores	Tablas de valores.
--	---	---	--------------------

Fuente: Autores.

3. FUNDAMENTACIÓN CIENTÍFICO TÉCNICA

3.1. Análisis de trabajo precedentes

Se propuso un módulo que identifique símbolos mediante técnicas de procesamiento de imágenes para el aprendizaje del código ColorADD en personas daltónicas, utilizando el algoritmo de los momentos de Hu (momento de imagen), como técnica de procesamiento de imagen. Como parte de la investigación se utilizó una muestra de 10 símbolos de código ColorADD ideales y se manejó una red neuronal multicapa perceptrón de 4 capas entrenada con valores dados por los momentos de Hu aplicada a la muestra para obtener resultados con imágenes reales[4].

De acuerdo a la tesis [5] manifiesta el reconocimiento de imágenes con Raspberry Pi cámara para la detección de los objetos por forma o color a través de un robot que utiliza un sistema operativo robótico. Este proyecto trabaja con un lenguaje de programación Python de forma básica con el fin de desarrollar a futuro acciones con mayor grado de dificultad.

En la tesis [6] expresa el desarrollo de un sistema de códigos para personas con ceguera al color en lugares internos, como centros comerciales, con el objetivo de comunicar el color real de las prendas de vestir. Se realizaron diferentes códigos simbólicos con el objetivo de permitir el reconocimiento de los colores para personas que padecen de daltonismo. Se realizó una investigación de las formas geométricas y simbólicas para comparar colores. Este sistema se destinó a siete daltónicos, cuatro oftalmólogos y cuatro diseñadores gráficos. Calificándolo con un alto nivel de eficiencia.

3.2. El daltonismo o ceguera de color

El daltonismo es un defecto genético o adquirido que una persona posee al no poder distinguir los colores rojo, verde, azul entre otros. Se presenta en un 7% a 8% en varones y 0,3% a 0,4% en mujeres, cerca de, 1 de cada 12 hombres y 1 de cada 200 mujeres presenta daltonismo, en su mayoría afecta a los dos ojos, es hereditario y va ligado al sexo. En un mayor porcentaje es una acromatopsia parcial es decir falta de visión o percepción a uno o dos colores; pero con un porcentaje bajo de una acromatopsia que es la pérdida total del color. Esto depende del cromosoma X, por esta razón los varones son más propensos a adquirir esta enfermedad. Aunque no todas las personas que sufren de esta patología confunden las mismas tonalidades [6].

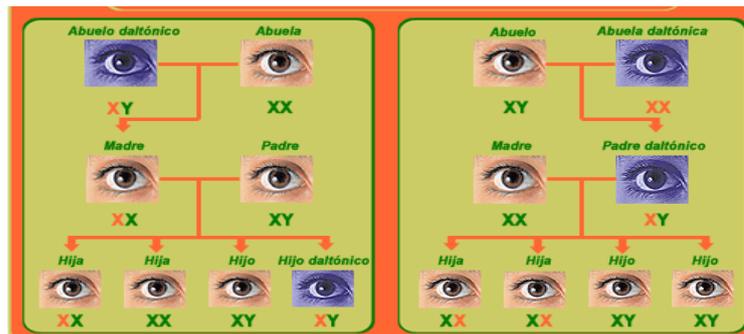


Figura 3. 1 Herencia del daltonismo.

Fuente:[7].

3.2.1. Causas del daltonismo

El daltonismo ocurre cuando existe un problema en la retina, donde encontramos las células sensoriales especializadas en captar estímulos de luz: los fotorreceptores conocidos como bastones y conos.

Estas células fotorreceptores tienen características escotópicas, es decir, tienen una alta sensibilidad a la luz acromática. Como los conos tienen la característica fotópica, son menos sensibles a la luz, pero son capaces de discriminar diferentes longitudes de onda.

Existen tres tipos diferentes de conos en el ojo humano, cada uno contiene un tipo de pigmento fotosensible. Un tipo detecta luz roja, otro detecta luz verde y el tercero detecta la luz azul, y todos juntos permiten visión del color.

Los estudios estadísticos muestran que alrededor del 8% de los hombres y el 0,4% de las mujeres tienen alguna forma de discapacidad relacionada con la percepción de los colores. Las deficiencias de color también se llaman discromatopsias, o simplemente daltonismo[8].

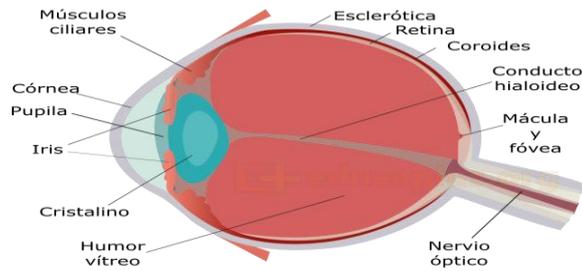


Figura 3. 2 Estructura anatómica del ojo humano.

Fuente:[9].

3.2.2. Problemas asociados al daltonismo

Las personas que padecen de daltonismo tienen la capacidad de distinguir diferentes tonos de colores violetas y la habilidad de observar objetos camuflados, que una persona con la visión normal no puede identificar, esta enfermedad va acompañada con varios inconvenientes de la vida cotidiana como son:

- No distinguir el estado de frescura o de descomposición de la comida.
- No distinguir los colores en objetos.
- Dificultad en seleccionar o combinar prendas de vestir.
- Dificultad al momento de conocer el estado de la batería del celular, debido a que el indicador de la batería viene sombreado de color rojo o verde[6].

3.2.3. Tipos de daltonismo

Aunque existen muchos tipos de daltonismos el 99% de los casos corresponde a protanopia y deuteranopia o sus equivalentes (protanomalia y deuteranomalia) [10].

- **Acromático**

El individuo ve en blanco y negro (escala de gris), por ausencia de los tres tipos de conos. Ésta condición es una de las más escasas a nivel mundial, ya que se ha visto en muy pocos casos[11].

- **Di-cromático**

Esto se da cuando el individuo presenta anomalía en la percepción de uno de los colores azul, verde o rojo. Puede ser de tres tipos diferentes:

Protanopia: Ausencia total de foto-receptores retinianos del rojo, por lo tanto, el individuo es incapaz de percibir cualquier luz roja.

Deuteranopia: Ausencia total de los fotorreceptores retinianos del color verde, siendo este el daltonismo dicromático más común; en este caso el individuo es incapaz de percibir la luz verde.

Tritanopia: es la ausencia total de los fotorreceptores retinianos del color azul, por lo que el individuo es incapaz de percibir cualquier tipo de luz azul[12].



Figura 3. 3 Normal/Visión Protanopia y Deuteranopia.

Fuente: Autores.

3.2.4. Pruebas de daltonismo

Las pruebas de visión del color se utilizan para los siguientes propósitos:

- Caracterizar el tipo de defecto.
- Diferenciar entre defectos congénitos y adquiridos.
- Evaluar la gravedad del defecto.
- Para guiar las pruebas vocacionales.

3.2.4.1. Test Ishihara

Los estudios demuestran que sigue siendo la prueba más efectiva (estándar de oro) para una identificación rápida de deficiencias congénitas para la visión del color. A pesar de haberse desarrollado para la detección y el diagnóstico de cambios congénitos en la visión del color, la prueba de Ishihara también se puede utilizar para detectar defectos adquiridos en la visión del color[13].

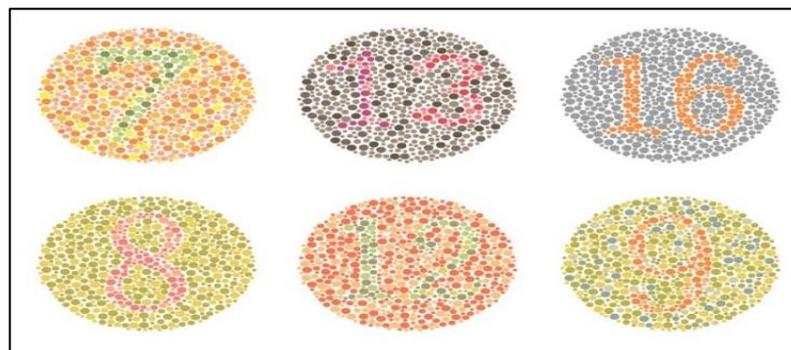


Figura 3. 4 Test de prueba de daltonismo(Ishihara).

Fuente: [14].

3.2.4.2. Test Hardy-Rand-Rittler

Esta prueba, tiene como objetivo detectar deficiencias congénitas del tipo protan, deutan, tritan y tetartan, estimar la gravedad del defecto.

Sus ventajas hacen que esta prueba sea ampliamente utilizada, junto con otras pruebas destinadas a la detección, como Ishihara. Entre ellos, podemos mencionar: la presencia de tabloncitos para la detección de defectos azul-amarillos[13].



Figura 3. 5 Test de prueba de daltonismo láminas (Hardy Rand Rittler).

Fuente: [14]

3.2.5. Tratamiento de daltonismo

En la actualidad no existe tratamiento para el daltonismo, o cura para esta enfermedad. Sin embargo, se ha desarrollado gafas y lentes de contacto que ayudan a diferenciar determinados colores.

3.2.5.1. Tratamiento de daltonismo infantil

- Si el daltonismo no es detectado a tiempo, puede causar un bajo autoestima en la niñez, y llevar este problema consigo a lo largo de su vida.
- Al descubrir el daltonismo en un niño, se debe explicar lo sucedido e informar a los docentes para que realicen actividades especiales de enseñanza.
- Los daltónicos desarrollan técnicas para aprender a vivir con su visión a los colores, pero las personas con visión normal al color, es muy difícil que entiendan como se sienten ya que esto puede influir y afectar psicológicamente a la persona que posee esta afección.



Figura 3. 6 ColorADD código para personas daltónicas.

Fuente: [15].

3.3. Redes neuronales artificiales

Las redes neuronales artificiales o sistemas conexionistas son sistemas de procesamiento de la información cuya organización y funcionamiento está basado en las redes neuronales biológicas. Consiste en un conjunto de elementos simples de procesamiento llamados nodos o neuronas acopladas entre sí por conexiones que poseen un valor numérico cambiante llamado peso. El aprendizaje en las redes neuronales artificiales es un proceso de ajuste o modificación de los valores o pesos de las conexiones[16].

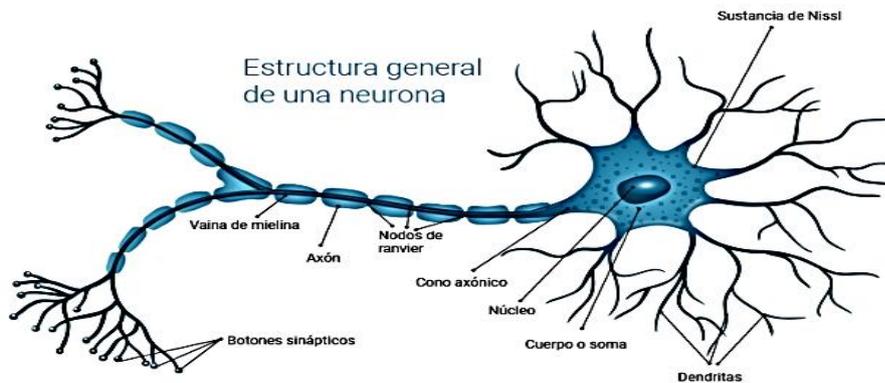


Figura 3. 7 Estructura general de una neurona.

Fuente:[17].

3.3.1. Características de una red neuronal

3.3.1.1. La neurona Artificial

La red neuronal artificial está conformada por un conjunto de neuronas artificiales, que constituyen dispositivos simples de cálculo que, a partir de un vector de entrada, la cual puede ser del mundo exterior o bien a partir de estímulos recibidos de otras neuronas, proporcionan una respuesta única. Se puede distinguir tres tipos de neuronas.

Neuronas de entrada: Recibe señales del entorno, provenientes de otras zonas del sistema o de sensores.

Neuronas de salida: Proporcionan su señal directamente fuera del sistema una vez terminado el tratamiento de la información.

Neuronas ocultas: Captan estímulos y emiten salidas dentro del sistema, sin la necesidad de mantener contacto alguno con el exterior, en ellas se realiza la función del procesamiento de información[18].

3.3.1.2. Estado de activación

Es preciso saber los estados del sistema en un tiempo t , esto se especifica mediante un vector de N números reales $A(t)$, que da inicio al estado de activación del conjunto de neuronas. Cada uno de los elementos del vector representa el estado de activación de una unidad en un tiempo t , es decir $a_i(t)$ entonces;

$$A(t) = (a_1(t), a_2(t), \dots, a_i(t), \dots, a_N(t)) \quad (3.1)$$

Donde:

$A(t)$: Vector de N números (Adimensional)

t : Es el número de neuronas de salida (Adimensional).

a_N : Estado de activación (Adimensional).

Todas las neuronas que constituyen la red neuronal se encuentra en un estado; excitado o de reposo, los cuales se los conoce como estados de activación, se les ha asignado un valor discreto o continuo[18].

3.3.1.3. Entradas a la neurona

Las variables externas que se presenta a la neurona de entrada pueden ser de distinto tipo:

- Dependiendo del tipo de red.
- Dependiendo la función que realice.

Binarias: Posee dos valores, por ejemplo, la variable sexo que adquiere los valores de mujer o hombre.

Continuas: Las variables adquieren valores en un intervalo numérico, por ejemplo, la edad[18].

3.3.1.4. Función de propagación

La función de propagación muestra el procedimiento a seguir para variar los datos de entrada y los pesos de las conexiones que llegan a la neurona. Los pesos w_{ij} se concentran en una matriz W , demostrando la influencia de la neurona i sobre la neuronal j , estos pesos pueden ser positivo, negativo o nulo[18].

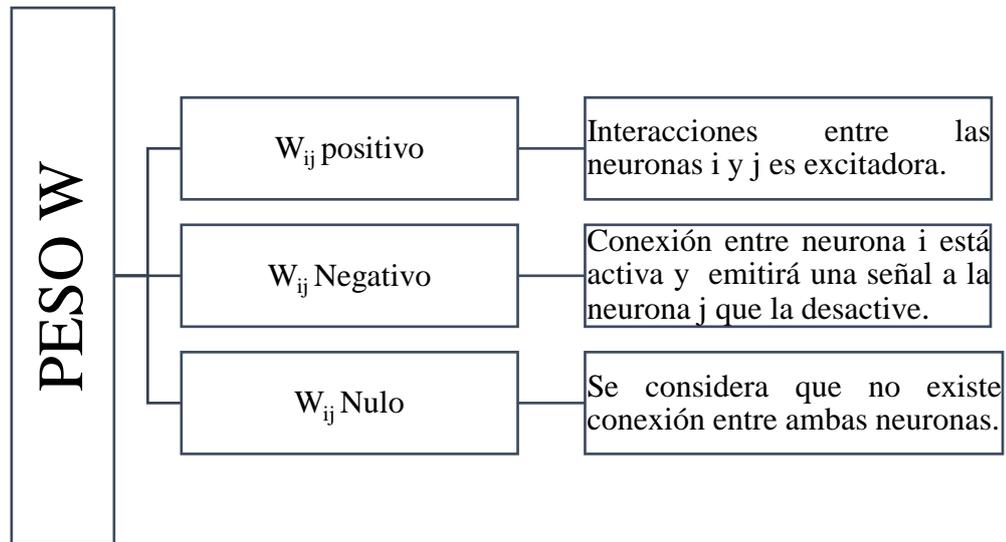


Figura 3. 8 Pesos W_{ij}

Fuente:[18]

La función de propagación obtiene el valor del potencial postsináptico Net_j de una neurona en un momento t , de acuerdo con la función σ ; el valor Net_j se obtiene en base a los valores de entrada y pesos obtenidos. La función que más se maneja es de tipo lineal: la suma ponderada de las entradas con los pesos sinápticos a ellas asociados[19].

$$Net_j(t) = \sum_i w_{ij} * x_i(t) \quad (3.2)$$

Donde:

Net_j : Potencial postsináptico(Adimensional)

w_{ij} : Pesos ij (Adimensional).

x_i : Señales de entrada (Adimensional).

Existe otra regla de propagación que también se utiliza, la distancias euclídea[18]:

$$Net_j(t) = \sqrt{\sum_i (w_{ij} * x_i)^2} \quad (3.3)$$

Donde:

Net_j : Potencial postsináptico(Adimensional)

w_{ij} : Pesos ij (Adimensional).

x_i : Señales de entrada (Adimensional).

3.3.1.5. Función de salida o de Transferencia

La función de transferencia o función de activación cumple la función de representar las salidas en función de las entradas a cada una de las neuronas. Es importante normalizar las entradas netas, y una vez normalizadas eligen su mejor intervalo[19].

Existen cuatro funciones de transferencia, como se describe que son:

- **Función Identidad o función lineal:** Esta función permite que la entrada sea igual a la salida, por lo es una regresión lineal[18].
- **Función Escalón:** Se presentan cuando las salidas de la red son binarias. Si la activación es menor que un determinado umbral, la salida se asocia con un determinado output, si esta salida es de igual valor o superior al umbral se asocia a otro output. La función puede desplazarse sobre los ejes. Esta función crea neuronas que clasifican en dos categorías distintas[18].

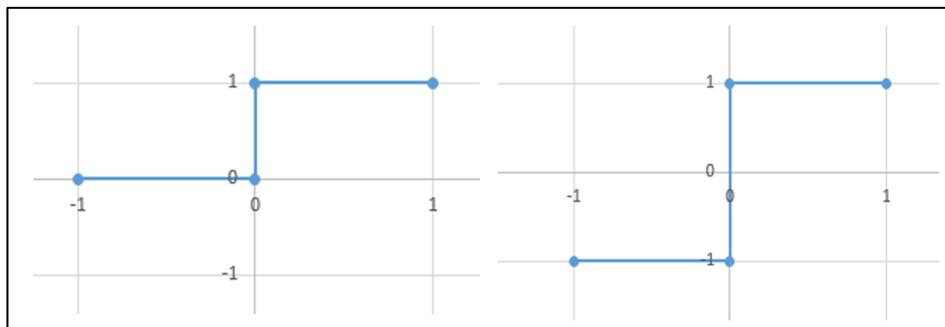


Figura 3. 9 Función Escalón.

Fuente: Autores.

- **Función lineal y mixta:** Si la activación de la unidades es menor que un límite inferior establecido, las salidas se asocia con un determinado valor[18].

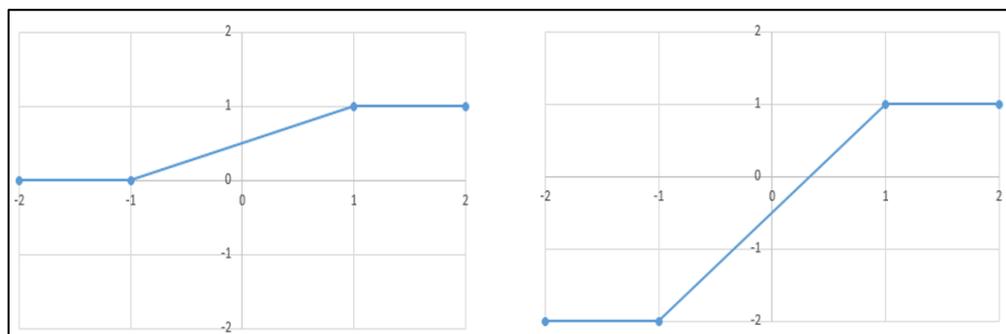


Figura 3. 10 Función lineal y mixta.

Fuente: Autores.

- **Función Sigmoidal:** Se utiliza cuando se necesita una salida de información analógica, esta función es adecuada debido a que los valores de entrada pueden variar entre más y menos infinito, y se devuelve como salida de valores entre 0 y 1; se utiliza habitualmente en redes multicapa como la Backpropagation[19].

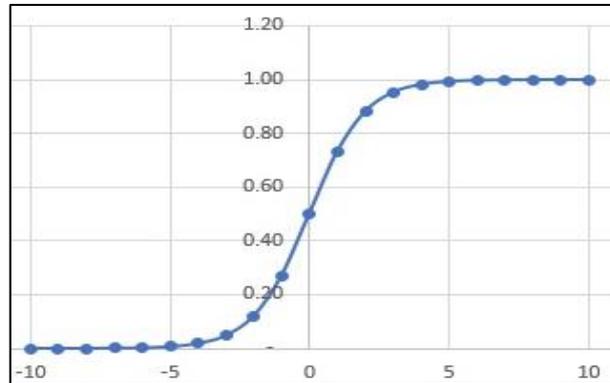


Figura 3. 11 Función Sigmoidal.

Fuente: Autores.

- **Función Gaussiana:** Adaptan la forma de una campana de Gauss cuyo centro, radio y apuntamiento son susceptibles a adaptaciones lo que las hace muy versátiles[18].

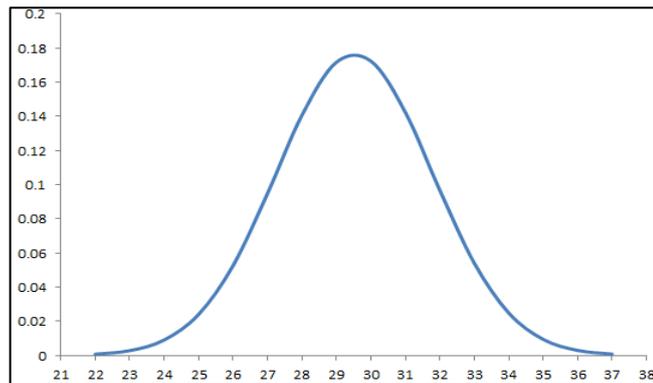


Figura 3. 12 Función Gaussiana

Fuente: Autores.

3.3.2. Tipos de redes neuronales

Existen diferentes tipos de redes neuronales, a continuación, se describen las principales:

3.3.2.1. El perceptrón simple

Esta red suma las señales de entrada y multiplica por los valores de pesos escogidos aleatoriamente; este valor es comparado con un patrón para establecer si la neurona es activada o no, si el valor comparado es mayor, la salida es 1, caso contrario 0, el perceptrón aprende de manera iterativa siguiendo estos pasos:

- Inicializar pesos y umbrales.
- Bucle: hasta resultado de pesos sea aceptable.
- Bucle: para todos los ejemplos.
- Leer valores de entrada.
- Calcular error.
- Actualizar pesos según el error.
- Actualizar pesos de entradas.
- Actualizar el umbral[19].

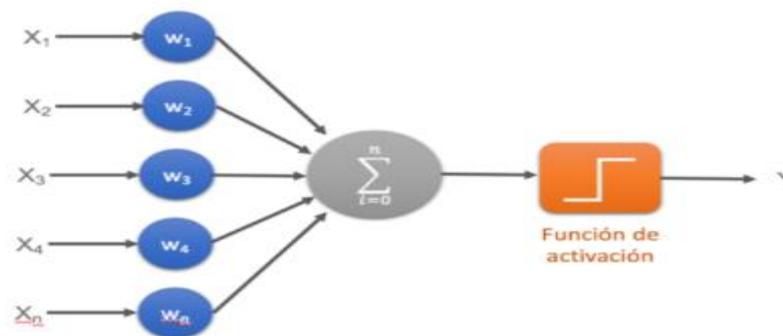


Figura 3. 13 Perceptrón Simple.

Fuente:[20].

3.3.2.2. Adaline

Esta red neuronal se asemeja a la red Perceptrón, esta utiliza una función de transferencia diferente, de tipo lineal. La importancia de esta red es que sirvió de base para el desarrollo de nuevos algoritmos.

El elemento de procesamiento realiza la suma de los productos de los vectores de entrada y pesos, y aplica una función de salida para obtener un único valor de salida, el cual debido a su función de transferencia lineal será +1 si la sumatoria es positiva o -1 si la salida de la sumatoria es negativa.

Esta red se utiliza para el procesamiento digital de señales. Su estructura se asemeja la del Perceptrón[19].

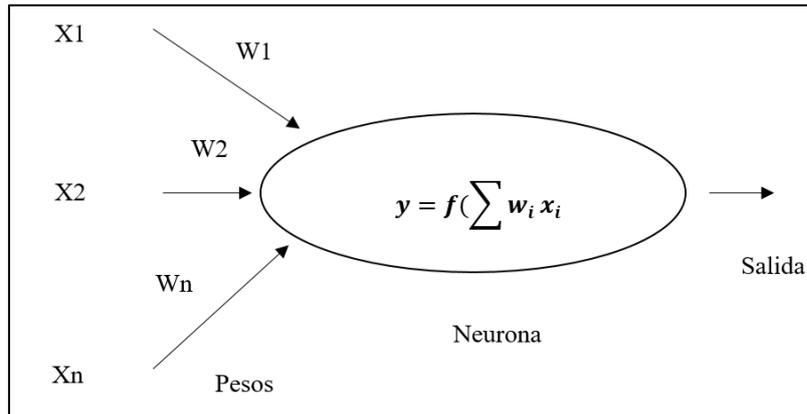


Figura 3. 14 Estructura de la red Adaline.

Fuente: Autores.

3.3.2.3.El perceptrón multicapa

Esta red utiliza aprendizaje supervisado y la que en mayores aplicaciones se ha utilizado; el MultiLayer Perceptrón MLP, está conformada por una capa de entrada, al menos una capa oculta y una capa de salida. Este tipo de redes utilizan en su entrenamiento propagación hacia atrás, o llamado retroalimentación del error.

Características de la red son:

- Se trata de una estructura altamente no lineal.
- Presenta tolerancia a fallos.
- Es capaz de establecer una relación entre dos conjuntos de datos[19].

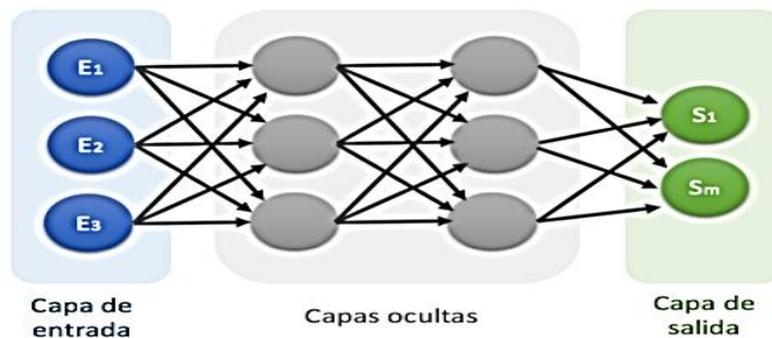


Figura 3. 15 Perceptrón multicapa.

Fuente:[20].

3.3.3. Aprendizaje de las redes neuronales artificiales

Para que una red aprenda, necesita un proceso que consta de modificar los pesos de las conexiones en a la información de entrada, el aprendizaje se encuentra representado en los pesos. Las conexiones de las neuronas pueden destruirse, modificarse o crearse en función al valor de los pesos.

3.3.3.1. Aprendizaje supervisado

El aprendizaje supervisado se caracteriza por la presencia de un agente externo que controle el proceso de entrenamiento, estableciendo la respuesta que debería generar la red (output del sistema) a partir de una entrada determinada.

De esta forma, el supervisor comprueba la salida de la red, la compara con la salida deseada, y si existe diferencias, ajusta iterativamente los pesos hasta que su salida tienda a ser la deseada utilizando para ello información detallada del error cometido en cada paso[18].

Tabla 3. 1 Modelos de redes neuronales artificiales con aprendizaje supervisado

TIPO DE APRENDIZAJE SUPERVISADO	MODELO DE RED
Aprendizaje por corrección de error	Perceptrón
	Adailen/Madaline
	Brain-State-in-a-Box

Fuente: Autores.

3.3.3.2. Aprendizaje por corrección de error

La red neuronal artificial genera una salida, esta es comparada con la que se espera obtener, la diferencia entre estos valores se utiliza para el aprendizaje por corrección del error. Los pesos de las conexiones se ajustan de acuerdo a esta diferencia, es decir en función al error cometido en la salida y van cambiando hasta lograr que la respuesta que genero la red sea la que deseada [21].

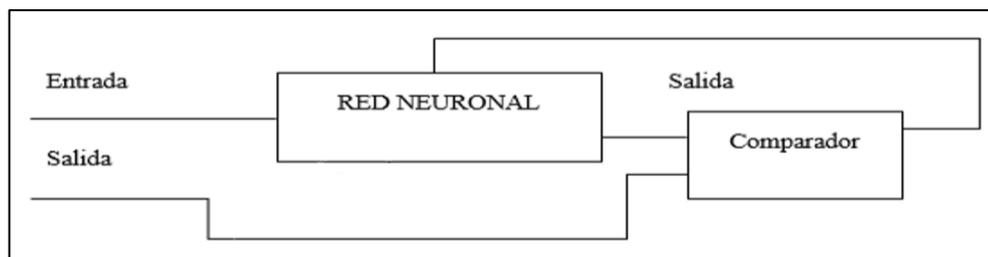


Figura 3. 16 Estructura de un aprendizaje con corrección de error.

Fuente:[22].

El método se puede describir en los siguientes pasos[22]:

1. Iniciar aleatoriamente los pesos.
2. Presentación del conjunto de entrenamiento.
3. Obtención de las salidas.
4. Comparación de salidas deseadas con las actuales.
5. Si se verifica el criterio de finalización ir el paso, si no ir al paso 2.
6. Fin.

3.3.4. Algoritmos de aprendizaje

En los algoritmos de aprendizaje, se recalca que cada uno de los algoritmos tienen especiales características, adecuadas a determinadas circunstancias. El diseñador en función de las circunstancias particulares en las que se encuentran (problema a resolver, tipo de plataforma disponible, etc.) debe escoger a la que se adapte a sus necesidades. Las características que deberían cumplir un algoritmo óptimo son:

- Eficacia.
- Robustez, para adaptarse a diferentes problemas.
- Independencia respecto a las condiciones iniciales.
- Alta capacidad de generalización.
- Coste computacional bajo.
- Sencillez en razonamiento empleado[23].

3.3.4.1. Algoritmo de aprendizaje Backpropagation

El algoritmo de aprendizaje Backpropagation es un algoritmo de descenso por gradiente que retropropaga las señales desde la capa de salida hasta la capa de entrada optimizando los valores de los pesos sinápticos mediante un proceso iterativo que se basa en la minimización de la función de coste. Por ello, puede dividirse el algoritmo en dos fases:

- Propagación hacia adelante: Se propaga las señales desde la capa de entrada hasta la de salida, determinando la salida de la red y el error cometido al comparar esta con el valor de la salida deseada que se facilita a la red durante la etapa de aprendizaje.
- Propagación hacia atrás: En función de los errores cometidos en la capa de salida, el algoritmo se encarga de optimizar los valores de los pesos sinápticos que determina las conexiones entre las neuronas mediante la retropropagación del error desde la capa de salida a la de entrada a través de las sucesivas capas ocultas.

Para la descripción de este algoritmo se tiene en cuenta el esquema de una neurona y el error como la diferencia existente entre la salida deseada y la salida observada. Se obtendrá el algoritmo para el caso más sencillo de una red formada por una capa de entrada, una capa oculta y una salida; la conexión entre una neurona oculta y una de salida. La extensión para el caso de más de una capa oculta es inmediatamente, se tratará de realizar una retropropagación del error durante un número mayor de capas.

En lo sucesivo $\{x_1, \dots, x_n\}$ serán las entradas a la red, es decir, que formara la capa de entrada y constituirá las entradas a las neuronas de la capa oculta mientras que x_0 es la entrada que,

dependiendo de que su valor sea +1 o -1, se denomina bias o umbral. Por otra parte $\{w_{m1}, \dots, w_{mn}\}$ serán los pesos sinápticos que conectan las entradas con la neurona oculta m y w_{m0} es el peso (peso sináptico correspondiente a x_0)[23].

$$v_m(t) = \sum_{i=0}^n w_{mi} * x_i(t) \quad (3.4)$$

Donde:

t : Es el número de neuronas de salida(Adimensional).

w_{mi} : Pesos(Adimensional).

x_i : Señales de entrada(Adimensional).

3.3.5. Aplicaciones de una red neuronal

La computación neuronal tiene ciertas ventajas con respecto a los métodos tradicionales, por ejemplo, pueden funcionar razonablemente incluso cuando se tiene entradas incompletas o con ruido. Por tal motivo se puede aplicar en una extensa variedad de aplicaciones[19].

- **Conversión texto a voz**

El cual convierte un texto en fonemas y con la ayuda de un sintetizador de voz, Dectalk genera voz a partir de un texto escrito. Dentro de esta área conversión texto-voz, las redes neuronales tiene gran ventaja con respecto a la computación, debido a que se programar un complejo conjunto de reglas de pronunciación en la computadora [19].

- **Proceso natural de lenguaje**

Esta aplicación ha aprendido el tiempo verbal past tense de los verbos en inglés; su funcionamiento es el siguiente:

Las características propias de la computación neuronal como la capacidad de generalizar a partir de datos incompletos y la capacidad de abstraer, permiten al sistema generar buenos pronósticos para verbos nuevos o verbos desconocidos[19].

- **Reconocimiento de caracteres**

Se ha desarrollado un sistema que es capaz de reconocer caracteres que no se han visualizado antes, después de un entrenamiento con un conjunto de tipos de caracteres de letras[19].

- **Procesado de la señal**

En este caso el sistema a controlar se modeliza para poder realizar predicciones a futuro de su comportamiento y, de esta forma, poder controlarlo fácilmente, por ejemplo, un sistema de vertido de residuos y aceites[23].

3.4. Hardware Libre

Hardware libre, se refiere a todos los recursos de hardware cuyas especificaciones de diseño y funcionamiento sean accesibles al público en general, de tal manera que cualquier persona con los conocimientos técnicos y herramientas necesarios se permita programar controladores, librerías y aplicaciones para hacer funcionar el dispositivo de la manera deseada[24].

3.4.1. Herramientas de diseño de Hardware libre

Se ha desarrollado muchas herramientas de hardware libre, y cada día nuevas con mejoras en sus capacidades y facilidades de uso. Algunas de éstas son software libre, otras son gratuitas o disponen de versiones gratuitas y otras que requieren de licencias pagas de uso; pero, en definitiva, todas ellas permiten y/o ayudan en el desarrollo de productos de hardware libre. Los diseñadores de hardware libre prefieren herramientas de software libre para el desarrollo de nuevos productos, pues les permite agregar funcionalidad y corregir errores de software por ellos mismos. Entre muchas de las herramientas, podemos destacar:

- FreeCAD.
- Arduino IDE.
- Eclipse IDE.
- Proteus Design Suite.
- NI Multisim.
- GNU Octave[24].

3.4.2. Ventajas y desventajas hardware libre

- Realizar dispositivos bajo la filosofía de hardware libre permite conocer las destrezas de personas talentosas distribuidas a nivel global que son capaces de ofrecer valiosa ayuda en la mejora consecutiva de los diseños.
- El aumento de la popularidad de las tecnologías de hardware libre, el desarrollo de dispositivos bajo esta filosofía permite una percepción de beneficios económicos que resultan interesantes de evaluar.
- Como desventaja las ganancias por ventas de dispositivos son más reducidas que si se hubiese desarrollado por métodos no libres, debido a que, al tener acceso a las especificaciones de diseño, se pueden realizar duplicados del dispositivo por parte de otras personas con la suficiente base económica para realizar una fabricación en serie[24].

3.5. Arduino

Arduino es una plataforma de hardware de código abierto, que consiste en una placa de circuito impreso que contiene un microcontrolador “ATMEL” que posee entradas y salidas, analógicas y digitales, en un entorno que está basado en el lenguaje de programación C++. El dispositivo conecta variables externas con el mundo virtual, o el mundo analógico con el digital controlando, sensores, alarmas. Sistemas de luces, motores, sistemas comunicacionales y actuadores físicos[25].

3.5.1. ADC Arduino

Cuando se trabaja con el ADC Arduino (conversor análogo digital) se debe tener en consideración que solo podemos colocar voltajes de máximo 5v o de 3.3v.

A continuación, se muestra una tabla con los voltajes máximos de operación y la máxima resolución de cada placa Arduino.

Tabla 3. 2 Los voltajes Max de operación y resolución de cada placa Arduino.

PLACA	VOLTAJE OPERACIÓN	PINES	MAX RESOLUCIÓN
Uno	5 Volts	A0 a A5	10 bits
Mini, Nano	5 Volts	A0 a A7	10 bits
Mega, Mega2560, MegaADK	5 Volts	A0 a A14	10 bits

Fuente: Autores.

Como se puede ver en la tabla 3.2, la lectura que haga el Arduino en la entrada análoga va a depender de la resolución ADC del Arduino que tenga. Por ejemplo, si estamos leyendo voltajes en un Arduino UNO, como es de 10 bits en realidad el Arduino va a ver una variación de un entero entre 0 a 1023.

Un bit es un binario que puede tomar dos valores (0 o 1), o sea que si tiene una resolución de 10 bits indica que ($2^{10} = 1024$), pero como empezamos desde 0, se dice que el valor entero toma valores entre 0 a 1023. Esto quiere decir que si Arduino mide:

- El máximo voltaje (5v) va a almacenar un valor entero de 1023.
- Voltaje intermedio (2.5v) va a almacenar un valor entero de 512.
- Voltaje mínimo (0v) va a alm
- almacenar un entero de 0

A través de lo visto anteriormente podemos mostrar entonces la ecuación de la resolución del ADC Arduino.

$$R = \frac{V_{ref}}{2^{10} - 1} \quad (3.5)$$

Donde:

R: Es la resolución (mV).

Vref: Es el voltaje de referencia ADC de la placa (V).

N: son los bits del convertidor ADC Arduino (bits).

Es decir que para el Arduino UNO la resolución es:

$$R = \frac{5v}{2^{10} - 1} = \frac{5v}{1023} = 4,88mV$$

Eso quiere decir que, en el caso de un arduino Uno, el valor de 0V analógico es expresado en digital como B0000000000 (0) y el valor de 5V analógico es B1111111111 (1023). Por lo tanto, todo valor analógico intermedio es expresado con un valor entre 0 y 1023, es decir, se suma 1 en binario cada 4,883 mV[26].

3.5.2. Tipos de Arduino

3.5.2.1. Arduino UNO

Arduino de una amplia gama básica, todas las shields están diseñadas para utilizarse en esta placa. Cuenta con una disposición de 14 pines entradas/salidas digitales de las cuales 6 se puede utilizar como PMW y cuenta con 6 entradas analógicas, I2C y SPI.

3.5.2.2. Arduino DUE

Arduino cuenta con un microcontrolador de 32 Bits, posee 54 entradas/salidas digitales y 12 entradas analógicas, 2 buses TWI, SPI Y 4 UARTs. Se acopla con todos los módulos basados en 3.3V y no soporta 5V debido a que puede quemar la placa.

3.5.2.3. Arduino Leonardo

Este Arduino posee 12 entradas analógicas y 20 entradas/salidas digitales. Se puede distinguir del resto de arduinos por el microcontrolador Atmega32u4 que no posee un controlador adicional para controlar el USB.

3.5.2.4. Arduino MEGA 2560

Arduino posee un microcontrolador ATmega2560. Cuenta con 54 entradas/ salidas digitales, 16 de ellos se puede usar como PMW, 16 entradas analógicas, 4 UART, 2 modos PWI y uno SPI

3.5.2.5.Arduino Nano

Está basado en un microcontrolador ATmega328. Posee similitud con un arduino UNO, las diferencias se encuentran en la forma de conectarlo al ordenador para programar, es compatible con la mayoría de shields de arduino.

3.5.3. Programación en Arduino

La estructura del lenguaje de programación de Arduino es básicamente simple y se compone de al menos dos partes. Estas dos partes necesarias, o funciones, encierran bloques que contienen declaración, estamentos o instrucciones.

Tabla 3. 3 Estructura y funciones de lenguaje de programación de arduino

ESTRUCTURA	
Estructura principal	Sintaxis
<ul style="list-style-type: none"> • Void setup () (Estructura de configuración) • Void loop () (Estructura de configuración) 	<ul style="list-style-type: none"> • ; (punto y coma) • { } (corchetes) • // (comentario línea única) • /**/ (comentario multilínea)
Estructura de control	Operadores aritméticos
<ul style="list-style-type: none"> • If() • If()...else • for() • switch()...case • while() • do()... while 	<ul style="list-style-type: none"> • = (asignación) • + (suma) • - (resta) • * (multiplicación) • / (división) • % (módulo)
Operadores booleanos	Constantes
<ul style="list-style-type: none"> • && (and) • (or) • ! (not) 	<ul style="list-style-type: none"> • HIGH LOW • INPUT OUTPUT • True false
Tipos de datos	Operadores compuestos
<ul style="list-style-type: none"> • Boolean (booleano) • char (carácter) • Byte (byte) • int (entero) • unsigned int (entero largo sin signo) • long (entero largo) 	<ul style="list-style-type: none"> • ++ (incremento) • -- (decremento) • += (suma compuesta) • -= (resta compuesta) • *= (multiplicación compuesta) • /= (división compuesta)
FUNCIONES	
Funciones I/O digitales	Comunicación serial
<ul style="list-style-type: none"> • pinMode (pin, mode) • digitalWrite (pin, value) • Int digitalWrite (pin) 	<ul style="list-style-type: none"> • Serial.begin (baudios) • int.Serial.available () • int Serial.read ()

Funciones I/O analógicas	<ul style="list-style-type: none"> • Serial.flush () • Serial.print (datos) • Serial.println (datos)
<ul style="list-style-type: none"> • int analogRead (pin) • analogWrite (pin, value) 	

Fuente: Autores.

En donde setup () es encargada de recoger la configuración y loop () es la que posee el programa que se ejecuta cíclicamente (de ahí el termino loop-bucle). Ambas funciones son necesarias para que el programa se ejecute.

La función de configuración debe poseer la declaración de las variables. Es la primera función a ejecutar en el programa, se ejecuta una sola vez, y se utiliza para configurar o inicializar pinMode (Modo de trabajo de las E/S), configuración de la comunicación en series y otras.

La función bucle(loop) siguiente contiene el código que se ejecuta continuamente (lectura de entradas, activación de salidas, etc.). Esta función es núcleo de todos los programas de Arduino[27].

3.6. Instrumentación Electrónica

La instrumentación electrónica es la técnica que se ocupa de la medición de cualquier tipo de magnitud física, de su conversión a magnitudes eléctricas y de su tratamiento para proporcionar información para un sistema de control o una persona[28].

La instrumentación está asociada con la medición de las variables de procesos, tipos de variables de acuerdo a su comportamiento en el tiempo son de tipo analógica y digital.

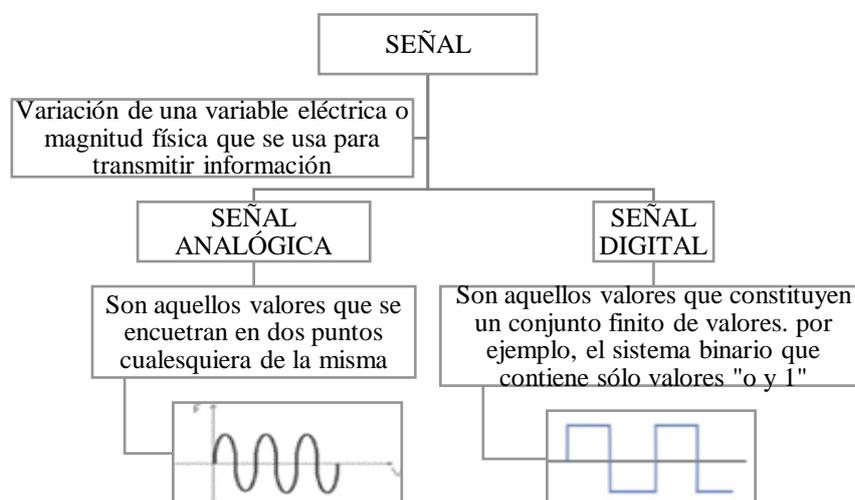


Figura 3. 17 Tipos de señales.

Fuente: Autores.

Un sistema de medida electrónico es aquel cuya finalidad es obtener información acerca de un proceso físico y presentar dicha información en la forma adecuada a un observador o a otro sistema de control.

3.6.1. Funciones de un sistema de medida

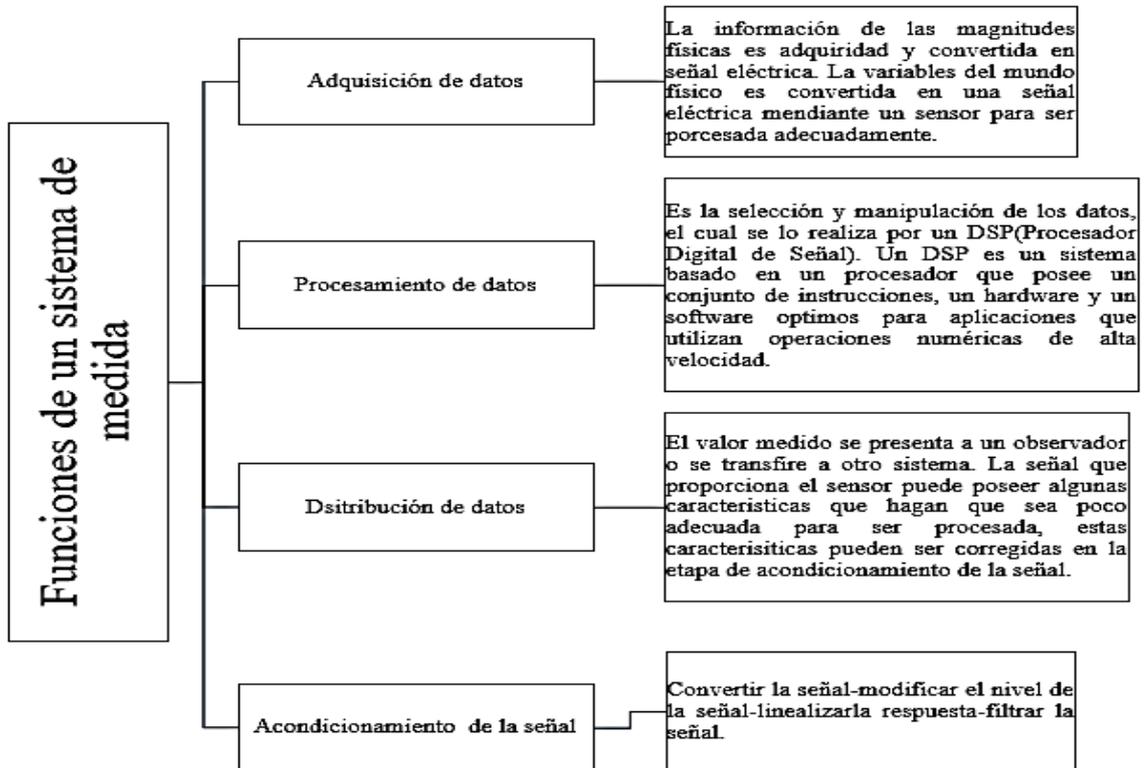


Figura 3. 18 Funciones de un sistema de medida.

Fuente: [28].

3.6.2. Fotorresistencia LDR

La LDR es una resistencia variable que cambia su óhmico (resistencia) en función de la densidad de luz que incide sobre ella (al aumentar la intensidad de luz que incide sobre LDR disminuye su resistencia)[29].



Figura 3. 19 Fotorresistencia LDR.

Fuente: Autores.

3.6.2.1. Acondicionamiento de la señal

La mayoría de las señales de preparación antes de poder ser digitalizadas. Aun las señales de voltaje puro requieren de tecnología para bloquear señales grandes de modo común o picos. Todas estas tecnologías de preparación son formas de acondicionamiento de señal. Para los sensores resistivos en general, en cuales caben las fotorresistencias los circuitos de acondicionamiento más utilizados son:

Divisores de Tensión o voltaje

El circuito esencial de un divisor de tensión, también llamado divisor de potencial o divisor de voltaje, su fórmula, es:

$$V_s = \left(\frac{R_2}{R_1 + R_2} \right) * V_e \quad (3.6)$$

Donde:

V_e : en el voltaje que alimenta el circuito.

V_s : es una fracción del voltaje de entrada.

R_1 y R_2 : es el valor de las resistencias que conforman el divisor.

Dependiendo de la configuración del divisor de voltaje con la fotorresistencia sustituyendo a la R_1 o R_2 , trabajara como un sensor de oscuridad o un sensor de luz. Este circuito da una tensión baja en la salida cuando el LDR está en la luz, y una tensión alta cuando la LDR está en la penumbra. El circuito divisor de tensión dará una tensión de la salida que cambia con la iluminación, de forma inversamente proporcional a la cantidad de luz que reciba.

Un sistema de sensor que funcione como esto se podría pensar como 'sensor de oscuridad' y se podría utilizar para controlar los circuitos de iluminación que se encienden (conectan) automáticamente por la tarde.

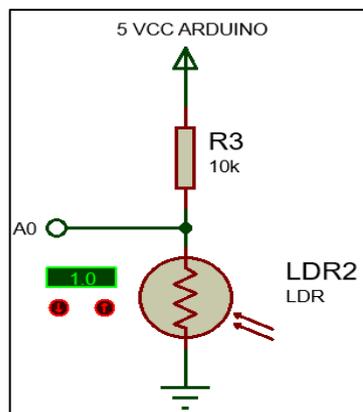


Figura 3. 20 Circuito de acondicionamiento para un detector de oscuridad.

Fuente: Autores.

Otro tipo de circuito de acondicionamiento es el caso de la figura 8 donde se observa un detector de luz, en el cual el principio físico consiste en entregar en la salida, en presencia de luz, un voltaje menor al que hay en presencia de oscuridad puesto que la caída de voltaje de la salida se mide directamente en la LDR. Baja cuando el LDR está en la oscuridad. Invertir el circuito, es decir, el voltaje de salida llega a ser alto, cuando la LDR está en la luz, y baja cuando el LDR está en la oscuridad[29].

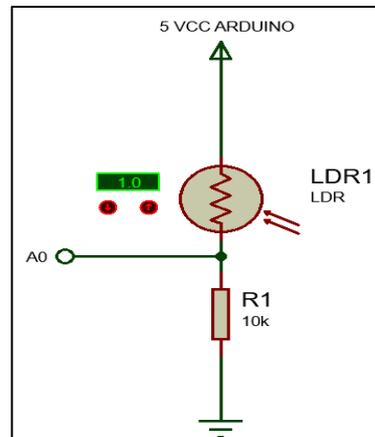


Figura 3. 21 Circuito de acondicionamiento para un detector de luz.

Fuente: Autores.

3.6.3. Tecnología diodo Led RGB

Tiene 2 diodos Led internos que emiten tres colores diferentes Red-Green-Blue(RGB), este método es particularmente interesante en muchos usos, debido a la flexibilidad de la mezcla de diferentes colores. Los Leds RGB de cuatro patas permiten manejarlos a gusto debido a que los pines corresponden a cada color están en el exterior. Los leds RGB de dos pines que, contiene todo el circuito de control en su interior[30].



Figura 3. 22 Led RGB de 4 pines.

Fuente:[30].

3.7. Repetitividad de ejecución.

El análisis de repetitividad se considera el cálculo de la muestra mediante la aplicación de la ecuación 3.7, dato a considerarse para el procedimiento y repetitividad en la identificación de los colores del dispositivo, seleccionados aleatoriamente de un total de 65 tonalidades de colores (blanco, negro, rojo, azul, verde, rosado, amarillo y celeste).

$$n = \frac{Z_{\alpha}^2 * N * p * q}{i^2 * (N - 1) + Z_{\alpha}^2 * p * q} \quad (3.7)$$

Donde:

n : Tamaño de muestra.

Z α : Distribución de Gauss.

N : Población.

p : Prevalencia esperada del parámetro a evaluar

q : 1-p

i : Error que se prevé cometer.

4. METODOLOGÍA Y DISEÑO EXPERIMENTAL

El presente capítulo se detalla los métodos empleados para el desarrollo de la propuesta tecnológica, con el fin de cumplir con los objetivos propuestos.

El método inductivo permitirá comprender la información bibliográfica recopilada en revistas, libros y páginas web por medio de la observación, análisis y registro de información de las redes neuronales aplicadas a dispositivos tecnológicos, esto sustentará el trabajo teórico y práctico, también ayudará al cumplimiento de los objetivos planteados.

Con el método deductivo y el método inductivo se buscará implementar el sistema de control para las variables del dispositivo; así como el diseño final de los componentes electrónicos y la estructura del dispositivo, por medio del análisis de resultados obtenidos de las pruebas experimentales del funcionamiento.

4.1. Definición de variables entrada y salida

4.1.1. Variable entrada

Tabla 4. 1 Operacionalización de variables de entrada.

Variable	Indicadores	Ítem	Técnicas o instrumentos
1. Intensidad de luz	Intensidad de luz reflejada en un objeto	Voltaje	Fotorresistencia. Acondicionamiento de la señal.
2. Identificación de los colores.	Identificación de colores mediante la utilización de redes neuronales	Porcentaje de exactitud en el control	Redes neuronales Algoritmos

Fuente: Autores.

4.1.2. Variable salida

Tabla 4. 2 Operacionalización de variables de salida.

Variable	Categoría	Indicadores	Ítems	Técnicas o instrumentos
1. Dispositivo funcional que permite la identificación de los colores para personas daltónicas.	Repetitividad en la identificación de los colores.	Número de colores que se identifica.	Tipo de color (0-7)	Programación Algoritmos
		Margen de exactitud en la identificación de los colores	Exactitud	Matriz comparativa(datos de colores)

Fuente: Autores.

4.2. Diseño experimental

Para lograr el cumplimiento de los objetivos de la propuesta tecnológica se dividirá en procedimientos fundamentales, que se mencionan en la siguiente tabla.

Tabla 4. 3 Procedimiento experimental.

PROCEDIMIENTOS	RESULTADOS
DISEÑO ELECTRÓNICO	<ul style="list-style-type: none"> • Número de componentes. • Acondicionamiento de los componentes electrónicos

PROGRAMACIÓN RED NEURONAL Y SOFTWARE ARDUINO	<ul style="list-style-type: none"> • Entrenamiento de la red neuronal • Valores de salidas de la red neuronal • Identificación de colores del dispositivo
DISEÑO Y CONSTRUCCIÓN DE LA CAJA DEL DISPOSITIVO	<ul style="list-style-type: none"> • Estructura del dispositivo

Fuente: Autores.

4.2.1. Diseño electrónico

Basándonos en las normas IEC 60717 la cual indica la clasificación de los componentes en activos y pasivos, mismo que serán utilizados para el funcionamiento del circuito.

4.2.1.1. Componentes principales

Arduino MEGA

Para la selección de este microcontrolador se basa en sus especificaciones técnicas como las entradas y salidas tanto analógicas como digitales y la memoria que posee, pero la conclusión definitiva para su selección frente a otro tipo de placa controladora es por la distribución de pines, sin afectar la eficiencia de control frente a otros controladores más avanzados y robustos.

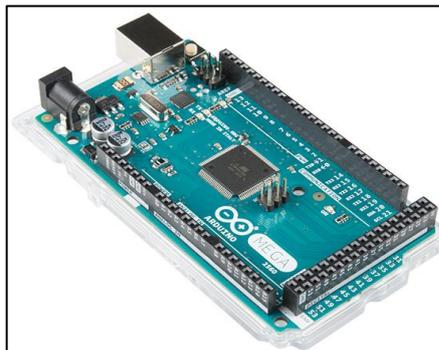


Figura 4. 1 Arduino Mega 2560.

Fuente:[31].

Fotorresistencia LDR

La fotorresistencia es uno de los principales componentes del sistema electrónico. Su función consiste en leer la variación de la intensidad luminosa que produce el diodo led RGB.

Las fotorresistencias se fabrican en muchos tamaños y formas para diferentes aplicaciones de la electrónica. Dentro del dispositivo identificador de color, básicamente va a proporcionar valores de la variación de la intensidad luminosa a la tarjeta de adquisición de datos(Arduino) por lo que para su selección se toma en cuenta los siguiente:

- Disponibilidad en el mercado y costo.
- Alta sensibilidad (debido a la gran superficie que puede abarcar).
- Alta relación resistencia Luz-oscuridad.
- Fácil empleo.

La fotorresistencia a adquirirse debe contar con las siguientes características:

Tabla 4. 4 Características de la fotorresistencia.

CARACTERÍSTICAS ESENCIALES DE LA FOTORRESISTENCIAS	
Resistencia(con luz)	50 Ohm
Resistencia(oscuridad)	10K Ohm
Voltaje máximo	150V
Disipación	100Mw máx.

Fuente: Autores.

Diodo led RGB

El diodo led RGB es un componente importante para la identificación de los colores. Cumple la función de proporcionar un haz de luz sobre el objeto a identificar.

Para la selección del tipo de diodo led RGB a utilizar, se considera los siguientes aspectos: La tarjeta de adquisición de datos (Arduino MEGA 2560) maneja voltajes de 0-5V y la corriente de 40mA.

Una vez mencionadas las características que se necesita, el led RGB que mejor se acopla a nuestras necesidades y a adquirirse cuenta con las siguientes características:

Tabla 4. 5 Características diodo led RGB.

CARACTERÍSTICAS LED RGB	
Corriente	20 mA
Tensión led(rojo)	2.1V
Tensión led(verde)	3.3V
Tensión led(azul)	3.3V
Polarizado	Negativo

Fuente: Autores.

Acondicionamiento del sensor

La fotorresistencia LDR (Ligth Dependent Resistor) es un resistor que cambia su valor de resistencia eléctrica dependiendo de la intensidad de luz que proporcione el diodo led RGB, la resistencia de la LDR es menor cuando la intensidad de luz es alta y en ausencia de luz la resistencia varía.

Para que la LDR trabaje como sensor de intensidad luz se debe emplear un divisor de voltaje, el cual es un circuito de resistencias en serie, el voltaje de salida es una fracción del voltaje de entrada.

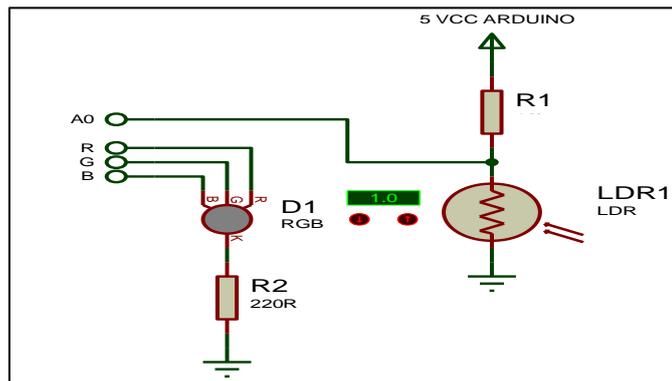


Figura 4. 2 Diagrama de sensor.

Fuente: Autores.

Para determinar el voltaje de salida del pin A0 se emplea la siguiente ecuación.

$$V_s = \left(\frac{R_2}{R_1 + R_2} \right) * V_e \quad (3.6)$$

Donde:

V_e : es el voltaje que alimenta el circuito.

V_s : es una fracción del voltaje de entrada.

R_1 y R_2 : es el valor de las resistencias que conforman el divisor.

El sensor permite ingresar los datos de la variación de intensidad luz ocasionada por el diodo led RGB al microcontrolador.

Tabla 4. 6 Acondicionamiento de la señal

Color	Escala de 0-1023 bits valores mínimos y máximos						Valores convertidos a escala 0-255		
	Voltaje	R(Bits)	Voltaje	G(Bits)	Voltaje	B(Bits)	R(Bits)	G(Bits)	B(Bits)
Blanco	4.07v	832	4.04v	828	4.05v	830	255	255	255
Negro	0.37v	76	0.36	75	0.35v	72	0	0	0

Fuente: Autores

Para el control de esta señal se debe acondicionar la entrada de la LDR, para este acondicionamiento se utiliza los valores mínimos y máximos que corresponden a la colores negro y blanco (0-255) para el control estos datos se dividen para 255 por lo cual los valores que ingresan a la programación de la red neuronal tienen el rango de (0-1).



Tabla 4. 7 Acondicionamiento de los componentes.

Fuente: Autores.

Como se muestran en la figura 4.6 se observa como los componentes electrónicos (Led RGB, Fotorresistencia) se encuentran dentro de una estructura en forma de cilindro, que permite que la intensidad de luz generada por el led, se concentren en el objeto a identificar el color y se refleje esta luz en la fotorresistencia y lea esta variación.

Acondicionamiento del parlante en el dispositivo

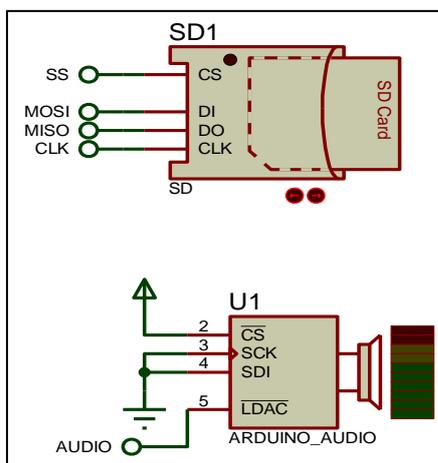


Figura 4. 3 Diagrama de elementos electrónico (tarjeta SD y parlante).

Fuente: Autores.

4.2.2. Programación

4.2.2.1. Programación red neuronal

La programación que se realiza en la red neuronal utiliza un lenguaje C y JAVA en el cual se detalla el procedimiento que se realizó para la creación de la red neuronal.

Tabla 4. 8 Diseño de la red neuronal.

Ítem	Diseño de la red neuronal.
Método	Modelamiento en software
Herramienta o Equipos	Computadora, software
Procedimiento	
1. Determinar los datos de entrada y salida de la red neuronal. 2. Parametrización de los datos de entrada de la red. 3. Realizar la programación de la red neuronal. 4. Entrenamiento de la red neuronal. 5. Comprobación del funcionamiento de la red neuronal. 6. Si con el entrenamiento de la red neuronal no se obtiene valores cercanos a los requeridos repetir el literal 5, hasta que los valores se aproximen a los requeridos	

Fuente: Autores.

A continuación, se detalla los componentes que intervinieron en la creación de la red neuronal:

Colores en código RGB

Los datos mostrados en la siguiente tabla muestran las combinaciones de código RGB para los 8 colores que el dispositivo identifica.

Tabla 4. 9 Combinación de colores en código RGB.

	Combinación de colores en código RGB		
	R	G	B
NEGRO	0	0	0
BLANCO	255	255	255
ROJO	255	0	0
VERDE	0	255	0
AZUL	0	0	255
AMARILLO	255	255	0

CELESTE	0	255	255
ROSADO	255	0	255

Fuente: Autores.

Parametrizar los códigos RGB a códigos binarios

Los datos mostrados en la siguiente tabla muestran la parametrización de los datos del color, código RGB a código binario.

Tabla 4. 10 Combinación de colores en código binario.

	Combinación de colores en código binario		
	R	G	B
NEGRO	0	0	0
BLANCO	1	1	1
ROJO	1	0	0
VERDE	0	1	0
AZUL	0	0	1
AMARILLO	1	1	0
CELESTE	0	1	1
ROSADO	1	0	1

Fuente: Autores.

Para esta parametrización se toma en cuenta que el código RBG toma los siguientes rangos 0-255 y el código binario valores de 0-1, por lo cual se asigna esta comparación; 0 = 0 y 255 = 1.

Asignación de valores a los colores

En la siguiente tabla se asigna valores a los colores para su identificación.

Tabla 4. 11 Asignación de valores a los colores.

	Combinación de colores en código RGB			
	R	G	B	Valor
NEGRO	0	0	0	0
AZUL	0	0	1	1

VERDE	0	1	0	2
CELESTE	0	1	1	3
ROJO	1	0	0	4
ROSA	1	0	1	5
AMARILLO	1	1	0	6
BLANCO	1	1	1	7

Fuente: Autores.

Datos ingresados a la red neuronal

En el diseño de la red neuronal se debe tener en cuenta los datos de entrada, así como las salidas requeridas. En el anexo 3 se visualizan los datos completos ingresados en la red neuronal.

Tabla 4. 12 Muestra de datos de entrada y salidas de la red neuronal.

	Parametrización de los datos en código RGB			Parametrización de los datos en código binario ingresados en la red neuronal			Salidas requeridas de la red neuronal	
	R	G	B	R	G	B		
NEGRO	0	0	0	0	0	0	0	NEGRO
AZUL	0	0	255	0	0	1	1	AZUL
VERDE	0	255	0	0	1	0	2	VERDE
CELESTE	0	255	255	0	1	1	3	CELESTE
ROJO	255	0	0	1	0	0	4	ROJO
ROSADO	255	0	255	1	0	1	5	ROSADO
AMARILLO	255	255	0	1	1	0	6	AMARILLO
BLANCO	255	255	255	1	1	1	7	BLANCO

Fuente: Autores

Código de programación de la red neuronal

En la figura 4.3 se detalla la programación ingresada para la ejecución de la red neuronal. Programación realizada en el software.

```

5   %entradas
6
7   r = [0 0.03 0.045 0.06 0 0 0 0 0.05 0.06 0 0 0 0 0.05 0.1 0.15 0.2 0.25 0.3 0
8   g = [0 0.03 0.045 0.06 0 0 0.06 0.1 0.15 0.2 0.25 0.3 0 1 0.95 0.9 0.85 0.8 0
9   b = [0 0.03 0.045 0.06 0.1 0.2 0.5 0.6 0.7 0.489 0.8 0.9 1 0 0.01 0.02 0.03 0
10
11  entrada = [r;g;b]; %entradas de la red
12
13
14
15  s = [0 0 0 0 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3
16
17  pesosIniciales = [-1,8];
18  bias = [-1,5];
19
20  net = newff(entrada, s);% newff crear una red de retroalimentación
21  [net,tr] = train(net,entrada,s);% Entrenamiento de la red
22
23  % net = newp([0 1 ; 0 1; 0 1],1); %red perceptron de 3 entradas y 1
24  % neurona
25  % %net.iw(1) = pesosIniciales;
26  % %net.b(1)=bias;
27  % net.trainParam.epochs = 100; % iteraciones de entrenamiento
28  % net = train(net, entrada, s);

```

Figura 4. 4 Programación red neuronal.

Fuentes: Autores.

Estructura de la red neuronal

La red neuronal está constituida de:

- 3 entradas.
- 1 capa oculta (1 bias 3 pesos, función de salida o transferencia).
- la salida.

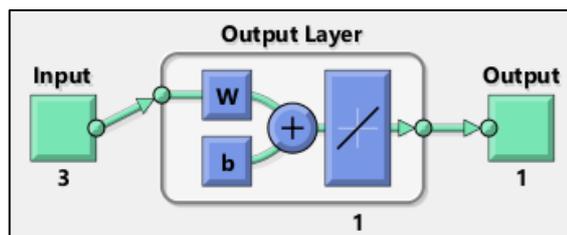


Figura 4. 5 Estructura red neuronal.

Fuente: Autores.

Transferencia de datos de la red neuronal a software matemático

Esta transferencia de datos ayuda a realizar la programación en software Arduino.

Tabla 4. 13 Transferencia de datos de la red neuronal.

Ítem	Transferencia de datos red neuronal.
Método	Modelamiento en software matemático.
Herramienta o Equipos	Computadora, software matemático.
Procedimiento	

1. Obtener los pesos y el bias de la red neuronal.
2. Conocer la función de salida o de transferencia de la red neuronal, con el empleo de la ecuación (3.2)
3. Colocar la fórmula de la función de transferencia lineal en el software matemático.
4. Colocar los pesos y el bias en el software matemático.
5. Ejecutar el software matemático.
6. Verificar si los valores de salida de la red neuronal coinciden con los valores de software matemático.

Fuente: Autores.

A continuación, se detalla los componentes que intervinieron en la transferencia de resultados de la red neuronal:

Bias y pesos de la red neuronal

En figura 4.6 se muestra los datos del bias y los 3 pesos de la red neuronal, estos valores son importantes al momento de validar los resultados de la red neuronal.

```
>> wb=formwb(net,net.b,net.iw,net.lw)
[b,iw,lw]= separatewb(net,wb)

wb =

    0.0321
    0.6351
    0.2219
    0.1590
```

Figura 4. 6 Bias y pesos de la red neuronal.

Fuente: Autores.

Función de salida o de transferencia y tratamiento de datos

Se utilizó esta función de activación lineal, debido a que el software de la red neuronal ya establece esta función en base a los datos ingresados y las salidas requeridas.

$$z = \text{Entrada}_1 * w_1 + \text{Entrada}_2 * w_2 + \dots + \text{Entrada}_N * w_N + b * 1 \quad (4.1)$$

Donde:

z: Función de activación (Adimensionales).

Entradas_N: Entradas (Adimensionales).

w_n : Pesos(Adimensionales).

b: pesos

El tratamiento de los datos es importante debido a que como presentan los datos a la red, estos influyen en su respuesta. Para el tratamiento de datos se utilizo es siguiente método:

Método: MÍNIMO-MÁXIMO (Conversión de entrada y salida)

El método a seguir es el siguiente:

- 1.- Se busca el máximo y el mínimo.
- 2.- Los datos serán normalizados según la relación.

$$Datos_{normalizados} = \frac{(max-min)}{(Xmax-xmin)} (x - xmin) + min \quad (4.2)$$

Sin importar cuales sean los datos originales, los datos caen dentro del intervalo $\{-1,1\}$, en la conversión de entrada.

Sin importar cuales sean los datos originales, los datos caen dentro del intervalo $\{0,7\}$, en la conversión de salida.

Transferencia de datos de la red neuronal a software matemático

Esta transferencia de datos permite validar el correcto funcionamiento de la red neuronal.

```
In[68]:= b = 0.0321;
        w1 = 0.6351;
        w2 = 0.2219;
        w3 = 0.1590;
        Entrada1 = 1;
        Entrada2 = 1;
        Entrada3 = 1;
        z = w1 * Entrada1 + w2 * Entrada2 + w3 * Entrada3 + b

Out[75]:= 1.0481

In[16]:= Conversion Entrada
        max = 1;
        min = -1;

        xmax = 1;
        xmin = 0;

        x = 1;
        Scala = (max - min) / (xmax - xmin) * (x - xmin) + min

Out[16]:= Conversion Entrada

Out[22]:= 1
```

Figura 5. 1 Transferencia de datos a software matemático.

Fuente: Autores.

4.2.2.2. Programación en software Arduino

Para realizar una programación completa (Componentes-red neuronal) en Arduino primero se necesita transferir los datos del software matemático a la programación de Arduino.

Tabla 4. 14 Transferencia de los datos de software matemático - programación de Arduino.

Ítem	Transferencia de los datos de software matemático a la programación de Arduino.
-------------	---

Método	Modelamiento en software Arduino.
Herramienta o Equipos	Computadora, software Arduino.
Procedimiento	
<ol style="list-style-type: none"> 1. Realizar una programación para ingresar los datos software matemático – Arduino 2. Transferir los datos del software matemático a la programación Arduino. 3. Verificar si los resultados de la red neuronal coinciden con los datos de la programación de Arduino. 	

Fuente: Autores.

Programación realizada en arduino

En la figura 4.7 se muestra la programación que permite transferir los datos del software matemático a la programación Arduino.

```

1 void setup() {
2     Serial.begin(9600);
3 }
4
5 void loop() {
6     double REDBGR = RED(0.6351, 0.2219, 0.1590, 0.0321, 1, 1, 1);
7     Serial.println(REDBGR);
8     delay(5000);
9 }
10
11 double ConvEntrada1(double x){
12     double max1 = 1;
13     double min1 = -1;
14     double xmax = 1;
15     double xmin = 0;
16     double Scala;
17     Scala = (max1 - min1)/(xmax - xmin)*(x - xmin) + min1;
18     return Scala;
19 }
20
21
22
23
24
25
26
27
28
29

```

Figura 4. 7 Programación software arduino.

Fuente: Arduino.

Después de haber realizado la transferencia de datos se procede a realizar una programación completa en Arduino con todos los componentes, Red Neuronal, fotorresistencia, diodo Led RGB y bocina.

Tabla 4. 15 Programación del dispositivo.

Ítem	Programación del dispositivo.
Método	Modelamiento en software Arduino.
Herramienta o Equipos	Computadora, software Arduino.
Procedimiento	
<ol style="list-style-type: none"> 1. Programación para el microcontrolador arduino con los componentes (Led RGB, Fotorresistencia LDR y bocina) 	

2. Asignar valores mínimos y máximos para la calibración del dispositivo.
3. Colocar en la programación los rangos de los datos que ingresan de la fotorresistencia para ser comparados con los datos entrenados en la red neuronal y si estos entran en los valores estudiados.
4. Si estos datos ingresan en ese rango se imprimirá en la pantalla el nombre del color.
5. Desarrollar la programación para el microcontrolador Arduino de forma auditiva.

Fuente: Autores.

En la siguiente figura se describe el proceso que va a tener dentro de la programación para lo cual se empleó un diagrama de flujo.

En el anexo 10 se describe la programación realizada en el software arduino.

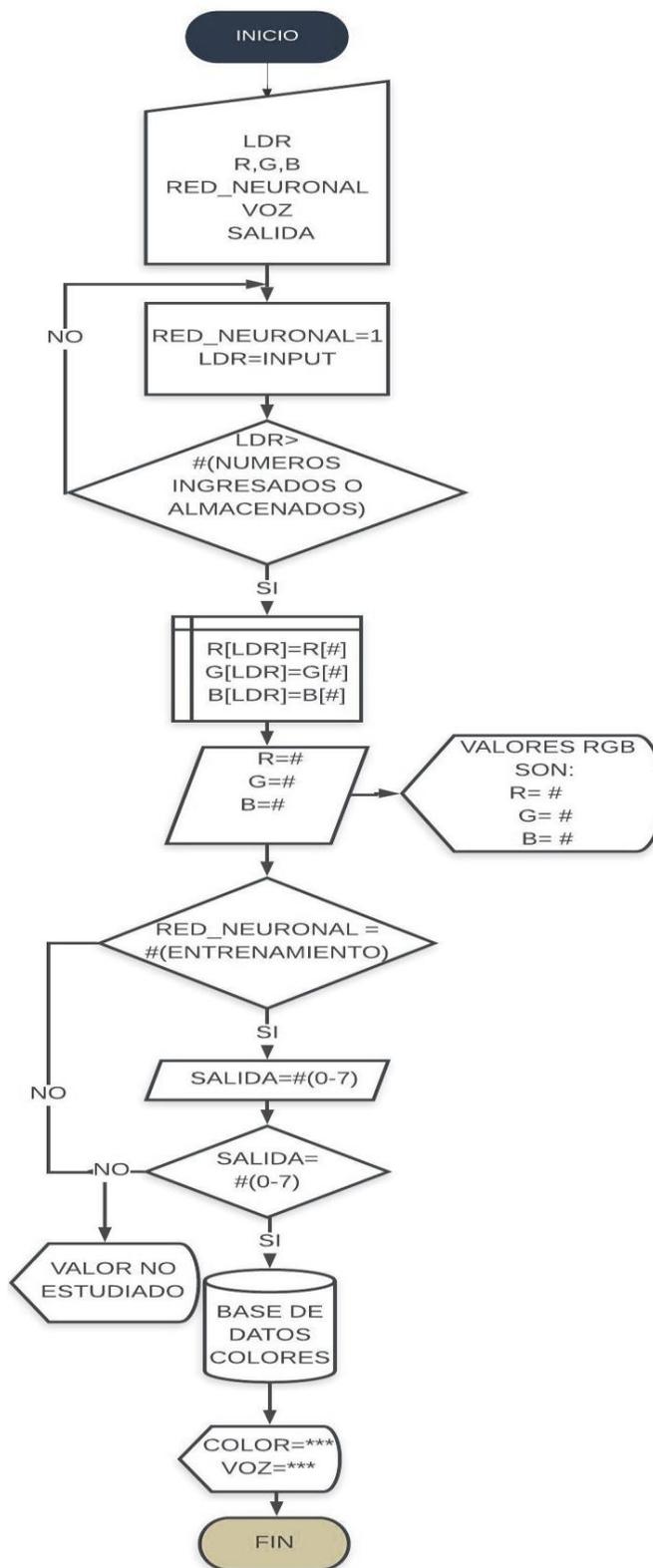


Figura 4. 8 Flujograma de la programación de software.

Fuente: Autores.

Para la ejecución de la programación se inicia declarando cada uno de los componentes tanto de entrada y de salida como lo son: fotorresistencia LDR, diodo RGB, red neuronal, pantalla LCD y bocina, posteriormente se realiza la activación de la red neuronal mediante los valores de entrada que tendrá la LDR, una vez ponderado los valores recibidos de la LDR son almacenados en la matriz que esta a su vez comparará, y con ayuda de la red neuronal elegirá los datos correspondientes al código de colores que fue entrenada la red neuronal, en caso de que los valores de entrada captados por la LDR sean distintos a los entrenados, se mostrara en la pantalla LCD un mensaje “NO ESTUDIO”, cabe recalcar que antes de utilizar el dispositivo para la identificación de colores se debe realizar la calibración que consiste en equilibrar los datos analógicos que ingresan al Arduino y se convierten en digitales y toman valores máximos y mínimos(255-0) que corresponden a los colores blanco y negro, estos dividirlos para 255 para que tomen un rango de 0-1 respectivamente, dando así la correcta respuesta en el momento de la identificación de los diferentes colores.

4.2.3. Diseño y construcción de la caja del dispositivo

Tabla 4. 16 Diseño y construcción de la caja del dispositivo.

Ítem	Diseño y construcción de la caja del módulo
Método	Diseño en software de elementos finitos
Herramienta o Equipos	Computadora, software CAD
Procedimiento	
<ol style="list-style-type: none"> 1. Crear un nuevo archivo CAD 2. Conociendo previamente las dimensiones del dispositivo se crea la estructura de la caja. También se diseña las salidas de audio de la bocina, y otros elementos que conforman el modulo. 3. Definido el diseño estructural del dispositivo se procede a imprimir 3D la caja del dispositivo. 	

Fuente: Autores.

4.2.4. Repetitividad de ejecución

El análisis de repetitividad se considera el cálculo de la muestra mediante la aplicación de la ecuación 3.7, dato a considerarse para el procedimiento y repetitividad en la identificación de los colores del dispositivo, a continuación, se expersa el número de muestras.

$$n = \frac{1.96^2 * 65 * (0.5) * (0.5)}{0.05^2 * (65 - 1) + 1.96^2 * (0.5) * (0.5)}$$

$$n = 55$$

Por lo cual el numero de muestras son 55.

5. ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS

Para los análisis y resultados nos guiaremos en la temática de las tablas 4.3, la misma que se ha dividido en varias áreas técnicas, los resultados se detallan en la siguiente tabla 13.

Tabla 5. 1 Análisis y resultados.

ANÁLISIS Y RESULTADOS		Resultados
DISEÑO ELECTRÓNICO	<ul style="list-style-type: none"> Número de componentes 	10 Componentes tanto activos como pasivos
	<ul style="list-style-type: none"> Acondicionamiento de los componentes electrónicos 	Acondicionamiento óptimo de la fotorresistencia y diodo led RGB.
PROGRAMACIÓN	<ul style="list-style-type: none"> Entrenamiento de la red neuronal 	Valores entrenados.
	<ul style="list-style-type: none"> Valores de salidas de la red neuronal 	Datos de salida de la red neuronal cercanos a las salidas requeridas.
	<ul style="list-style-type: none"> Identificación de los colores dispositivo 	Validación del funcionamiento del dispositivo.
DISEÑO Y CONSTRUCCIÓN DE LA CAJA DEL DISPOSITIVO	<ul style="list-style-type: none"> Estructura del dispositivo 	Dimensiones de la estructura.

Fuente: Autores.

En la anterior tabla se describe los resultados que se pretendieron llegar a lo largo del proyecto, pero no son todos, existen otros que se detallan en el apartado de anexos.

5.1. Resultados de programación red neuronal-Arduino

5.1.1. Entrenamiento de la red neuronal

En la figura 5.3 se puede visualizar como los datos ingresados y entrenados en la red neuronal se acercan a los valores de salidas requeridas.

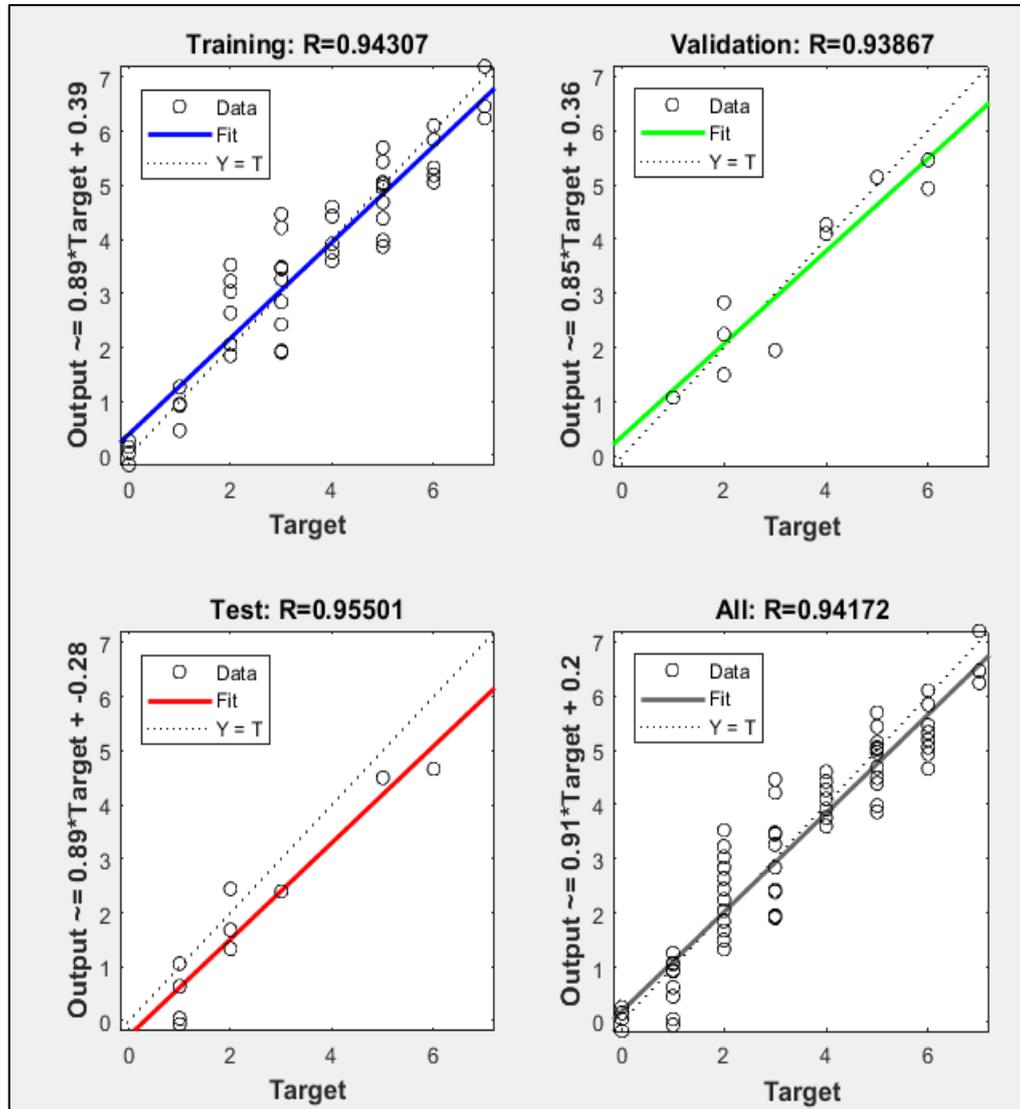


Figura 5. 2 Entrenamiento de la red.

Fuente: Autores.

Si en la red neuronal se incrementan el número de entrenamientos los valores de salida se acercaron a los valores deseados o requeridos.

5.1.2. Valores de salidas de la red neuronal

Estos valores son comparados datos teóricos- datos obtenidos por la red neuronal.

Tabla 5. 2 Validación de resultados de la red neuronal.

DATOS TEÓRICOS					DATOS OBTENIDOS POR LA RED NEURONAL				
R	G	B	RANGO	COLOR	R	G	B	RANGO	COLOR
1	1	1	7	BLANCO	1	1	1	7.17	BLANCO
0	0	1	1	AZUL	0	1	1	1.16	AZUL
0	0	0	0	NEGRO	0	0	0	0.0560	NEGRO
1	0	0	4	ROJO	1	0	0	4.50	ROJO
0	1	0	2	VERDE	0	1	0	1.6092	VERDE
1	0	1	5	ROSA	1	0	1	5.61	ROSA
0	1	1	3	CELESTE	0	1	1	2.72	CELESTE
1	1	0	6	AMARILLO	1	1	0	6.0553	AMARILLO

Fuente: Autores.

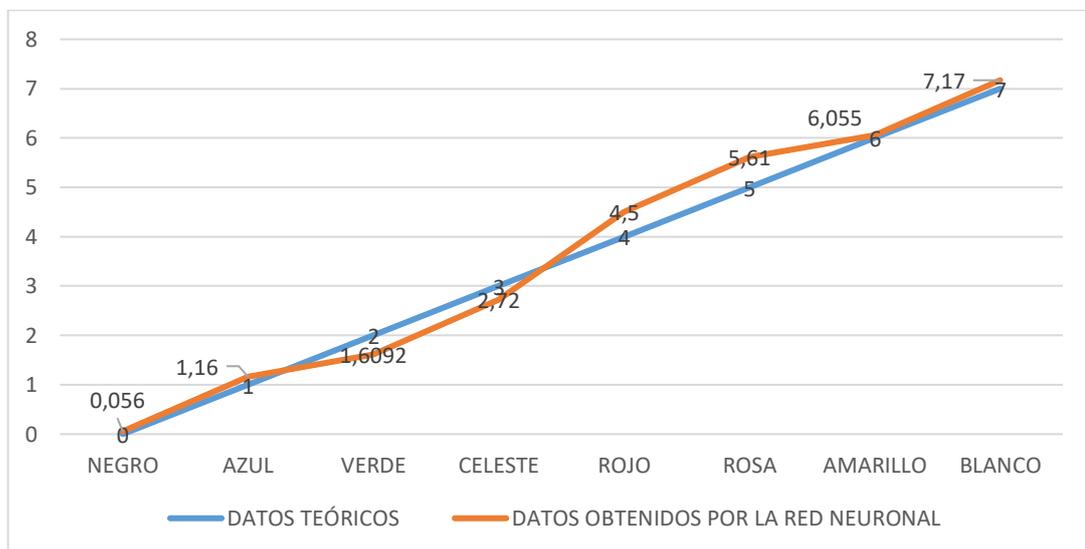


Figura 5. 3 Comparación de datos obtenido con datos teóricos.

Fuente: Autores.

Como se muestra en la figura 5.3 se toma una muestra de los valores (Teóricos-Obtenidos) y estos no se alejan significativamente el uno del otro, por lo cual son resultados aproximables que están dentro de los rangos, por lo cual se puede afirmar que la red neuronal está entrenada correctamente.

5.1.3. Identificación de los colores dispositivo

5.1.3.1.Descripción del dispositivo

El dispositivo identificador de colores está diseñado con una tecnología que hoy en día está a la vanguardia de nuevas formas de control de procesos, como lo es la inteligencia artificial y por ende se refiere a la red neuronal que es donde se hace énfasis en el desarrollo del prototipo para la ayuda de las personas que sufren de daltonismo



Figura 5. 4 Dispositivo identificador de colores.

Fuente: Autores.

El dispositivo es básicamente un circuito electrónico, el cual identifica los colores para personas daltónicas haciendo uso de las nuevas tecnologías, como son las redes neuronales en el control de procesos. Los componentes principales que abarca el prototipo, son elementos de la electrónica convencional, como lo es la fotorresistencia (LDR), diodo led RGB y un microcontrolador (Arduino MEGA) los cuales reciben y procesan datos, para ser comparados con el entrenamiento de la red neuronal y así determinar el color en base al aprendizaje realizado.

En el anexo 2. Se muestran las conexiones de todos los componentes del dispositivo identificador de colores.

5.3.2.2. Validación del funcionamiento del dispositivo

Tabla 5. 3 Validación del dispositivo mediante programación de la red neuronal

VALIDACIÓN DEL DISPOSITIVO IDENTIFICADOR DE COLORES MEDIANTE PROGRAMACIÓN														
Valores de la pantalla LCD del dispositivo				<ul style="list-style-type: none"> Datos ideales de los colores en código RGB Valores de entrenamiento de la red neuronal. 					Comprobación de cada uno de los colores identificados del dispositivo, mediante programación					
R	G	B	R	Código Binario	G	Código Binario	B	Código Binario	Rango De Estimación 0-7	Comprobación Del Entrenamiento	% Exactitud	Reconocimiento del color		
												SI	NO	
NEGRO	4	1	0	0	0	0	0	0	0	0.12	88 %	✓		
AZUL	0	77	158	0	0	0	255	1	1	1.17	83%	✓		
VERDE	102	110	5	0	0	255	1	0	2	2.12	88%	✓		
CELESTE	105	169	202	0	0	255	1	255	3	3.13	87%	✓		
ROJO	232	0	0	255	1	0	0	0	4	4.09	91%	✓		
ROSA	220	50	82	255	1	0	255	1	5	5.11	89%	✓		
AMARILLO	250	231	56	255	1	255	1	0	6	6.19	81%	✓		
BLANCO	250	249	246	255	1	255	1	255	7	7.16	84%	✓		

Fuente: Autores.

Para la comprobación de los colores en código RGB que fueron entrenados en la red neuronal, se realizó la verificación con ayuda de un software el cual analizó cada uno de los valores dando como resultado el menor error en cada interacción correspondiente a cada color, siendo así el resultado un 86,37 % de eficiencia en la validación Dispositivo-Red neuronal.

Análisis de repetitividad

El Análisis de repetitividad parte del cálculo de la ecuación 3.7 la misma que arroja el valor de la muestra de 55, es decir que para el procedimiento se aplica el reconocimiento de los colores del dispositivo de manera aleatoria para el análisis.

Tabla 5. 4 Validación del dispositivo identificador de colores mediante pruebas.

Número de pruebas	Color mostrado	Color reconocido	Funciona (Si / No)
1	Amarillo Oscuro	Amarillo	Si
2	Amarillo Claro	Amarillo	Si
3	Amarillo De Cromo	Amarillo	Si
4	Amarillo Verdoso	Verde	Si
5	Amarillo Puro	Amarillo	Si
6	Ámbar	Amarillo	Si
7	Azul	Azul	Si
8	Azul Acero	Azul	Si
9	Azul Alicia	Blanco	Si
10	Azul Cadete	Celeste	Si
11	Azul Marino	Azul	Si
12	Azul Maya	Celeste	Si
13	Azur	Azul	Si
14	Rojo	Rojo	Si
15	Rojo De Venecia	Rojo	Si
16	Rojo Indio	Rosa	Si
17	Rojo Remolacha	Rosa	Si
18	Rosa	Rosa	Si
19	Rosa Fucsia	Rosa	Si
20	Rubí	Rosa	Si

21	Sable	Blanco	Si
22	Salmón	Rosa	No
23	Verde	Verde	Si
24	Verde Amarillento	Verde	Si
25	Verde Bosque	Verde	Si
26	Verde Camuflaje	Negro	No
27	Verde Caribe	Celeste	Si
28	Verde Oscuro	Verde	Si
29	Cian	Celeste	Si
30	Celeste	Celeste	Si
31	Negro	Negro	Si
32	Negro Claro	Negro	Si
33	Negro Intenso	Negro	Si
34	Negro Brilloso	Negro	Si
35	Blanco	Blanco	Si
36	Blanco Intenso	Blanco	Si
37	Blanco Puro	Blanco	Si
38	Blanco Seda	Blanco	Si
39	Blanco Hueso	Blanco	Si
40	Blanco Antiguo	Blanco	Si
41	Blanco Roto	Blanco	Si
42	Blanco Tiza	Blanco	Si
43	Azul Medio	Azul	Si
44	Azul Oscuro	Azul	Si
45	Azul Capri	Celeste	Si
46	Azul Francia	Celeste	Si
47	Amarillo Canario	Amarillo	Si
48	Azul Eléctrico	Azul	Si
49	Verde Brillante	Verde	Si
50	Verde Agua	Celeste	Si
51	Blanco Antiguo	Blanco	Si

52	Azul	Azul	Si
53	Negro intenso	Negro	Si
54	Amarillo claro	Amarillo	Si
55	Azul marino	Azul	Si

Fuente: Autores.



Figura 5. 5 Validación de la pruebas del dispositivo.

Fuente: Autores.

Mediante al análisis de repetitividad se obtuvieron los siguientes resultados, de 55 pruebas del dispositivo, identifique 53 colores y 2 colores que no se identificaron, dando un porcentaje de eficiencia del dispositivo de un 96% pero debemos tener en cuenta que el prototipo está diseñado para una cantidad limitada de colores (8) debido a que se enfoca a un cierto grupo de personas como se las denomina daltónicas.

5.2. Resultados diseño y construcción de la caja del dispositivo

En el diseño de la caja de dispositivo intervinieron factores importantes como son el número de componentes, así como las dimensiones de cada componente, otro aspecto importante es la maniobrabilidad del dispositivo debido a que este debe estar diseñado para personas con daltonismo. Revisar anexo 7.



Figura 5. 6 Caja del dispositivo

Fuente: Autores

5. PRESUPUESTO Y ANÁLISIS DE IMPACTOS

5.3. Presupuesto

Este consiste en implementar los gastos que intervinieron en la ejecución del proyecto, en las tablas que se presentan a continuación se fijan los precios de los materiales a utilizar en el proyecto.

5.3.2. Materiales electrónicos

Los materiales electrónicos utilizados para el dispositivo identificador de colores son:

Tabla 6. 1 Materiales Electrónicos.

DETALLE	DESCRIPCIÓN	CANTIDAD	VALOR UNITARIO	VALOR TOTAL
Arduino MEGA	2560 (8Bits)	1	\$ 23.00	\$ 23.00
Led RGB	40 mA- Polarización negativa	1	\$ 1.50	\$ 1.50
Resistencia LDR	Resistencias (luz) 1K ohm Resistencias (oscuridad) 10 K ohm	1	\$ 0.5	\$ 0.50
Resistencias	220 Ohm y 10K ohm	2	\$ 0.10	\$ 0.20
Cables de Arduino	10 cm Longitudinales	25	\$ 0.10	\$ 2.50
Pantalla LCD 16X2	Alfanumérico 5V	1	\$ 15.00	\$ 15.00
Parlante	0.25 W – 8 Ohm	1	\$ 1.70	\$ 1.70

Modulo amplificador de sonido	2.5 – 5.5V 3W	1	\$ 5.50	\$ 5.50
Tarjeta Micro SD	4 GB Clase 10	1	\$ 5.00	\$ 5.00
Modulo lector de tarjetas	Micro SD arduino	1	\$ 5.75	\$ 5.75
			TOTAL	\$ 60.75

Fuente: Autores.

5.3.3. Gastos de estructura y diseño de ingeniería del dispositivo

La estructura del dispositivo fue diseñada por una impresora 3D.

Tabla 6. 2 Estructura del dispositivo y diseño de ingeniería.

DETALLE	DESCRIPCIÓN	CANTIDAD	VALOR UNITARIO	VALOR TOTAL
Estructura de caja	Dimensiones	1	\$ 20.00	\$ 20.00
Programación	Código de programación	1	\$ 50.00	\$ 50.00
			TOTAL	\$ 70.00

Fuente: Autores.

5.3.4. Gastos varios

Los gastos varios realizados durante el proceso del desarrollo del dispositivo identificador de color.

Tabla 6. 3 Gastos varios

DETALLE	DESCRIPCIÓN	VALOR TOTAL
Material Bibliográfico.	Consultas, etc.	\$ 25.00
TOTAL		\$ 25.00

Fuente: Autores.

5.3.5. Gastos directos totales

Tabla 6. 4 Gastos directos totales.

GASTOS DIRECTOS	COSTOS
Materiales electrónicos	\$ 60.65
Estructura del dispositivo y diseño de ingeniería	\$ 70.00
TOTAL	\$ 130.65

Fuente: Autores.

5.3.6. Gastos indirectos totales

Tabla 6. 5 Gastos indirectos totales.

GASTOS INDIRECTOS TOTALES	COSTOS
Material Bibliográfico.	\$ 25.00
TOTAL	\$ 25.00

Fuente: Autores.

5.3.7. Gastos totales del dispositivo

Tabla 6. 6 Gastos totales.

Gastos indirectos totales	\$ 130.65
Gastos directos totales	\$ 25.00
Total	\$ 155.65

Fuente: Autores.

5.3.2. Cálculo del TIR y VAN

Inversión Inicial	150			
FLUJO DE INGRESOS	A	FLUJO DE EGRESOS	B	
AÑO	VALOR	AÑO	VALOR	
1	150	1	15	
2	150	2	15	
3	150	3	20	
FLUJO EFECTIVO NETO	A-B			
AÑO	VALOR			
		-200		
1	135	135		
2	135	135		
3	130	130		
			FORMULACIÓN DE DATOS	
			F1	90
			F2	110
			F3	130
			i	10% tasa de interés
			IO	200
		VAN	\$ 181,97	
		TIR	45%	

Figura 5. 7 TIR y VAN.

Fuente: Autores.

Se tomó la inflación del $5\% = 0.05$ de incremento por cada año, en la figura 5.7 se muestra el flujo de caja durante los 3 años en el cual se debe recuperar la inversión, además para el VAN y TIR se utilizó la hoja de cálculo de Excel, con una tasa de interés del 10%. Con los resultados obtenidos el proyecto es viable, donde se obtiene un VAN (Valor Actual Neto) positivo de USD 181.97 de una inversión inicial de 150. Además, el TIR obtenido del 45%.

5.4. Análisis de impacto

5.4.2. Impacto tecnológico

Las nuevas tendencias tecnológicas que existe a nivel mundial, las cuales hablan de la inteligencia artificial para el control de nuevos procesos, basándose en la utilización de redes neuronales. El dispositivo identificador de colores está construido con la electrónica convencional añadiéndole nuevas técnicas de control.

5.4.3. Impacto social

Después de haber realizado una investigación acerca del daltonismo, se obtuvo que existen un limitado número dispositivos para resolver esta problemática debido a que los casos son muy reducidos y las personas que lo padecen tienen que vivir y adaptarse a esa forma de vida, al ser una enfermedad genética, el dispositivo sirve como tratamiento para las personas daltónicas y mejora su calidad de vida, argumentando a la consulta médica por parte del área de medicina general de la Universidad Técnica de Cotopaxi sede la matriz.

5.5. Beneficiarios del proyecto

5.5.2. Beneficiarios directos:

El proyecto de investigación tiene como beneficiarios directos a las personas que sufren de daltonismo, debido a que el dispositivo ayuda a identificar los colores que estas personas no pueden reconocer.

5.5.3. Beneficiarios indirectos:

El proyecto tiene como beneficiarios indirectos a la comunidad académica, al indagar las nuevas técnicas que permiten controlar de procesos en el área de la Electromecánica (Instrumentación, sistemas de control y programación).

6. CONCLUSIONES Y RECOMENDACIONES

6.3. Conclusiones

- En base a la información recopilada acerca del daltonismo se determinó, que la enfermedad es de índole genético y se presenta en un 7% a 8% en varones y 0,3% a 0,4% en mujeres, por lo cual 1 de cada 12 hombres y 1 de cada 200 mujeres.
- La red neuronal que se acopló a nuestras necesidades es la perceptrón simple debido a que utiliza 3 entradas con las características en código RGB respectivamente y la salida en el rango de 0-7.
- Una vez realizado la implementación del dispositivo concluimos que, a pesar de utilizar los elementos de la electrónica convencional, se logró acoplar estos componentes al control de la identificación de los colores de la red neuronal.
- Para corroborar el funcionamiento del dispositivo fue necesario comparar los datos de la red neuronal, con los valores obtenidos por el dispositivo para la verificación del reconocimiento de los colores.
- Con la validación del funcionamiento mediante la ecuación del tamaño de la muestra se realizaron 55 pruebas y se evidenció que el dispositivo falló en 2 pruebas y en las 53 restante si identifico el color por lo cual al prototipo se le da un rango de eficiencia del 96%.

6.4. Recomendaciones

- Se recomienda que, si se desea ampliar el número de colores a identificar, se deberá utilizar una red neuronal que se acople a los datos ingresados y a las salidas requeridas.
- Al utilizar elementos de la electrónica convencional el nivel de respuesta de la red neuronal es menos precisa, por el motivo de la calibración del dispositivo, por lo cual se recomienda cambiar la fotorresistencia y el diodo RGB por un sensor de colores, debido a que las librerías ya se encuentran explícitas para su ejecución.
- Se recomienda investigar otro tipo de red neuronal que se acople a los valores de entrada y salida que posee actualmente el dispositivo identificador de colores.
- Los entrenamientos que se realizaron en el aprendizaje de la red neuronal en cada uno de los colores nos lleva a verificar el grado de exactitud que posee el sistema para la identificación del color para así garantizar la correcta función del prototipo.

- Para la calibración del dispositivo es necesario usar los valores mínimos y máximos como los son el color blanco y negro dentro de la red neuronal, con esta calibración el dispositivo tendrá una alta eficiencia al momento de la identificación de colores.

7. BIBLIOGRAFÍA

- [1] Medlineplus, “Daltonismo,” *Medlineplus*, 2014. [Online]. Available: <https://medlineplus.gov/spanish/ency/article/001002.htm>.
- [2] A. Méndez, “DALTONISMO,” *blog.ciencias-medicas*, 2017. [Online]. Available: <https://blog.ciencias-medicas.com/archives/1869#:~:text=En América Latina 1 de,o uso de algún medicamento.&text=La mayoría de los casos,problema genético ligado al sexo.>
- [3] Á. Menéndez, “Hipótesis para explicar las ventajas evolutivas de ser daltónico.,” 2014.
- [4] A. Gallegos, Dharmia; Gonzales del Valle Romero, “SIMULACIÓN DE MÓDULO RECONOCEDOR DE SÍMBOLOS MEDIANTE TÉCNICAS DE PROCESAMIENTO DE IMÁGENES PARA APRENDIZAJE DEL CÓDIGO COLOR ADD EN PERSONAS DALTÓNICAS,” *Universidad Ricardo Palma*, 2019. [Online]. Available: http://repositorio.urp.edu.pe/bitstream/handle/URP/2743/ELECT030_73242034_T_GONZALES_DEL_VALLE_ROMERO_ANGELO.pdf?sequence=1&isAllowed=y.
- [5] R. Palacios, “Detectar los objetos por forma o color a través de un robot,” *Universidad Abierta de Cataluña*, 2018. [Online]. Available: <http://hdl.handle.net/10609/71945>.
- [6] M. Guamán, “DISEÑO DE UN SISTEMA DE CÓDIGOS PARA PERSONAS DALTÓNICAS EN ESPACIOS FÍSICOS INTERNOS: CENTRO COMERCIAL DE RIOBAMBA,” *Escuela Superior Politécnica de Chimborazo*, 2015. [Online]. Available: <http://dspace.esPOCH.edu.ec/handle/123456789/3784>.
- [7] A. DE REMBRANDT, “Daltonismo,” *ANATOMIA DE REMBRANDT*, 2018. [Online]. Available: <https://anatomiarembrant.wordpress.com/2015/03/30/daltonismo/>.
- [8] J. Lee, “Simulación basada en difusos del daltonismo,” *IEEE Xplore*, 2010. [Online]. Available: <https://sci-hub.tw/10.1109/IEMBS.2010.5627128>.
- [9] CERTIMEDIC, “¿Cómo afecta el daltonismo en la Navegación?,” *CERTIMEDIC*, 2018. [Online]. Available: <https://certimedic.es/como-afecta-el-daltonismo-en-la-navegacion/>.

- [10] O. Cruz, Carlos; Sinclair, *El poder del color*. Panamá: like, 2015.
- [11] V. Moreno, Maria; Sánchez, “PREVALENCIA DE DALTONISMO EN ESTUDIANTES DE LAS UNIDADES EDUCATIVAS: HERLINDA TORAL, MANUEL J. CALLE, FRANCISCO FEBRES CORDERO, DOLORES J. TORRES Y OCTAVIO CORDERO,” *Universidad de Cuenca*, 2015. [Online]. Available: <http://dspace.ucuenca.edu.ec/bitstream/123456789/25348/1/TESIS.pdf>.
- [12] K. Gutiérrez, “Diseño de un artefacto interactivo para detectar el tipo y la amplitud del daltonismo en los niños,” *Universidad de Azuay*, 2019. [Online]. Available: http://201.159.222.99/bitstream/datos/9095/1/14739.pdf?fbclid=IwAR3KB0l2YqCHkODHLhA1kHnC8dWOrJpf82hwMAe-7YWil_7PdZePaUZR4XM.
- [13] A. Bruni, Fernanda; Velasco, “Sentido cromático: tipos de defectos y pruebas de evaluación clínica.,” *Scielo*, 2016. [Online]. Available: https://www.scielo.br/scielo.php?script=sci_arttext&pid=S0004-27492006000500028&lang=en.
- [14] AEOPTOMETRISTA, “Tipos de daltonismo,” *AEOPTOMETRISTA ORG*, 2018. [Online]. Available: <https://optometristas.org/tipos-de-daltonismo>.
- [15] M. Escorial, “Coloradd el código braille para daltónicos,” *Compromiso Empresarial*, 2015. [Online]. Available: <https://www.compromisoempresarial.com/tercersector/2015/10/coloradd-el-codigo-braille-para-daltonicos/>.
- [16] C. Sarraute, “Redes Neuronales Artificiales aplicadas al Análisis de Datos,” *UNIVERSIDAD DE LAS ISLAS BALEARES*, 2017. [Online]. Available: <https://www.tesisenred.net/bitstream/handle/10803/9441/tjjmm1de1.pdf?sequence=1>.
- [17] M. Campos, “Inspiración biológica de las redes neuronales artificiales,” *Soldai*, 2018. [Online]. Available: <https://medium.com/soldai/inspiración-biológica-de-las-redes-neuronales-artificiales-9af7d7b906a>.
- [18] J. López, Raquel; Fernández, *Las redes neuronales artificiales fundamentos teóricos y aplicaciones*. Netbiblo, 2018.
- [19] E. Tepán, “ESTUDIO DE LOS PRINCIPALES TIPOS DE REDES NEURONALES Y LAS HERRAMIENTAS PARA SU APLICACIÓN,” *Universidad Politécnica Salesiana*, 2013. [Online]. Available: <https://dspace.ups.edu.ec/bitstream/123456789/4098/1/UPS-CT002584.pdf>.

- [20] D. Calvo, “Perceptrón – Red neuronal,” *Diego Calvo*, 2018. [Online]. Available: <https://www.diegocalvo.es/perceptron/>.
- [21] Thales, “Características de las redes neuronales,” *Thales.Cica*, 2012. [Online]. Available: <https://thales.cica.es/rd/Recursos/rd98/TecInfo/07/capitulo3.html>.
- [22] Varpa, “Aprendizaje y entrenamiento redes neuronales,” *Varpa.org*, 2012. [Online]. Available: <http://www.varpa.org/~mgpenedo/cursos/scx/archivospdf/Tema2-0.pdf>.
- [23] A. Serrano, E. Soria, and J. Martín, “Redes Neuronales Artificiales,” *Universidad de Valencia*, 2010. [Online]. Available: http://ocw.uv.es/ingenieria-y-arquitectura/1-2/libro_ocw_libro_de_redes.pdf.
- [24] J. Hernández, “Hardware Libre,” *INNOTICA*, 2016. [Online]. Available: <https://innotica.net/blog/articulo/hardware-libre>.
- [25] C. Tapia and H. Manzano, “EVALUACIÓN DE LA PLATAFORMA ARDUINO E IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL DE POSICIÓN HORIZONTAL,” *Universidad Politécnica Salesiana*, 2013. [Online]. Available: <https://dspace.ups.edu.ec/bitstream/123456789/5522/1/UPS-GT000511.pdf>.
- [26] A. Castaño, “ADC Arduino – Entradas Analógicas,” *Control automático educación*, 2015. [Online]. Available: <https://controlautomaticoeducacion.com/arduino/entradas-analogicas-adc/>.
- [27] R. Gutiérrez, “Manual de Programación,” *ArduinoBot*, 2018. [Online]. Available: <https://arduino-bot.pbworks.com/f/Manual+Programacion+Arduino.pdf>.
- [28] E. Mandano, P. Mariño, and A. Lago, *INSTRUMENTACIÓN ELECTRÓNICA*. Marcombo, 1995.
- [29] J. Martínez, M. Villar, and M. Francisco, *Tecnología*. MAD, 2003.
- [30] C. Suárez, “IMPLEMENTACIÓN DE UN SISTEMA DE INFORMACIÓN PARA LA FIEE UTILIZANDO MÓDULOS LEDS RGB.,” *Escuela Politécnica Nacional*, 2013. [Online]. Available: <https://bibdigital.epn.edu.ec/bitstream/15000/6057/1/CD-4786.pdf>.
- [31] ARDUINO, “ARDUINO MEGA 2560,” *ARDUINO.CL*, 2020. [Online]. Available: <https://arduino.cl/producto/arduino-mega-2560/>.

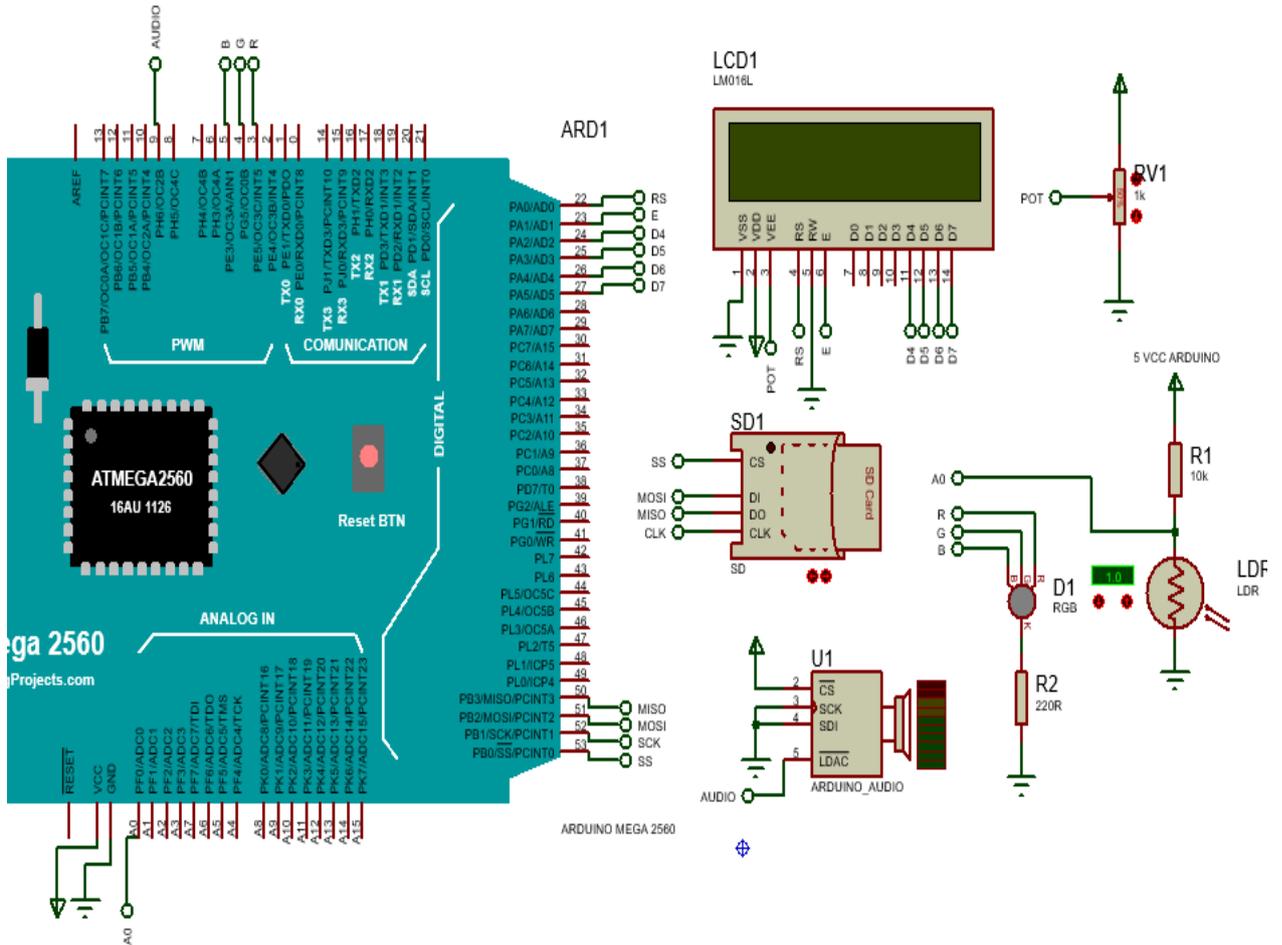
ANEXOS

Anexo 1. Tabla con los datos ingresados en la red neuronal

DATOS INGRESADOS EN LA RED NEURONAL				DATOS OBTENIDOS EN CÓDIGO RGB			
R2	G2	B2	VALOR	R255	G255	B255	COLOR
0	0	0	0	0	0	0	NEGRO
0,03	0,03	0,03	0	7,65	7,65	7,65	NEGRO
0,045	0,045	0,045	0	11,475	11,475	11,475	NEGRO
0,06	0,06	0,06	0	15,3	15,3	15,3	NEGRO
0	0	0,1	1	0	0	25,5	AZUL
0	0	0,2	1	0	0	51	AZUL
0	0,06	0,5	1	0	15,3	127,5	AZUL
0	0,1	0,6	1	0	25,5	153	AZUL
0,05	0,15	0,7	1	12,75	38,25	178,5	AZUL
0,06	0,2	0,489	1	15,3	51	124,695	AZUL
0	0,25	0,8	1	0	63,75	204	AZUL
0	0,3	0,9	1	0	76,5	229,5	AZUL
0	0	1	1	0	0	255	AZUL
0	1	0	2	0	255	0	VERDE
0,05	0,95	0,01	2	12,75	242,25	2,55	VERDE
0,1	0,9	0,02	2	25,5	229,5	5,1	VERDE
0,15	0,85	0,03	2	38,25	216,75	7,65	VERDE
0,2	0,8	0,06	2	51	204	15,3	VERDE
0,25	0,75	0,09	2	63,75	191,25	22,95	VERDE
0,3	0,7	0,12	2	76,5	178,5	30,6	VERDE
0,35	0,65	0,15	2	89,25	165,75	38,25	VERDE
0,4	0,6	0,18	2	102	153	45,9	VERDE
0,45	0,55	0,21	2	114,75	140,25	53,55	VERDE
0,5	0,5	0,24	2	127,5	127,5	61,2	VERDE
0,56	0,48	0,2785	2	142,8	122,4	71,0175	VERDE
0	1	1	3	0	255	255	CELESTE
0,393	0,69	0,65	3	100,215	175,95	165,75	CELESTE
0,56	0,76	0,75	3	142,8	193,8	191,25	CELESTE
0,35	0,7	0,8	3	89,25	178,5	204	CELESTE
0,5	0,75	0,85	3	127,5	191,25	216,75	CELESTE
0,3	0,65	0,55	3	76,5	165,75	140,25	CELESTE
0,05	0,6	0,9	3	12,75	153	229,5	CELESTE
0,1	0,5	0,8	3	25,5	127,5	204	CELESTE
0,15	0,4	0,7	3	38,25	102	178,5	CELESTE
0,18	0,7	0,6	3	45,9	178,5	153	CELESTE
0,25	0,8	0,95	3	63,75	204	242,25	CELESTE
1	0	0	4	255	0	0	ROJO
0,95	0,03	0,02	4	242,25	7,65	5,1	ROJO
0,9	0,07	0,04	4	229,5	17,85	10,2	ROJO
0,85	0,1	0,06	4	216,75	25,5	15,3	ROJO

0,8	0,12	0,08	4	204	30,6	20,4	ROJO
0,75	0,16	0,1	4	191,25	40,8	25,5	ROJO
0,7	0,2	0,12	4	178,5	51	30,6	ROJO
1	0	1	5	255	0	255	ROSA
0,95	0,03	0,94	5	242,25	7,65	239,7	ROSA
0,9	0,05	0,86	5	229,5	12,75	219,3	ROSA
0,9	0,05	0,78	5	229,5	12,75	198,9	ROSA
0,9	0,05	0,7	5	229,5	12,75	178,5	ROSA
0,85	0,08	0,62	5	216,75	20,4	158,1	ROSA
0,8	0,1	0,54	5	204	25,5	137,7	ROSA
0,7	0,13	0,46	5	178,5	33,15	117,3	ROSA
0,95	0,16	0,38	5	242,25	40,8	96,9	ROSA
0,85	0,19	0,3	5	216,75	48,45	76,5	ROSA
0,75	0,22	0,22	5	191,25	56,1	56,1	ROSA
1	1	0	6	255	255	0	AMARILLO
0,95	0,95	0,05	6	242,25	242,25	12,75	AMARILLO
0,88	0,88	0,1	6	224,4	224,4	25,5	AMARILLO
0,85	0,85	0,15	6	216,75	216,75	38,25	AMARILLO
0,82	0,82	0,2	6	209,1	209,1	51	AMARILLO
0,79	0,79	0,25	6	201,45	201,45	63,75	AMARILLO
0,76	0,76	0,31	6	193,8	193,8	79,05	AMARILLO
0,73	0,73	0,23	6	186,15	186,15	58,65	AMARILLO
1	1	1	7	255	255	255	BLANCO
0,9	0,9	0,9	7	229,5	229,5	229,5	BLANCO
0,87	0,87	0,87	7	221,85	221,85	221,85	BLANCO

Anexo 2. Diagrama de los componentes.



Anexo 3. Cconsulta al departamento de Bienestar Estudiantil de la Universidad Técnica de Cotopaxi.



**BIENESTAR
UNIVERSITARIO**

Latacunga 17 de julio del 2020

Señores

Sandro Joel Zambrano Pluas
Rubén Darío Guambi Manitio

Estudiantes de la Carrera de Ingeniería en Electromecánica

Presente

Luego de expresarles un saludo cordial, me dirijo a ustedes y dar contestación a su pedido.

Sugiero de la manera más comedida tener un acercamiento por medio de video conferencia, a fin de coordinar y llegar a determinar el alcance y magnitud de su proyecto, como también orientarles, para realizar y tratar de llegar al diagnóstico o si ustedes cuentan con el grupo de pacientes con diagnostico confirmado, de igual forma para que realicen la encuesta a la que se someterán a las personas seleccionadas.

Fono. 0987085270

Atentamente

A handwritten signature in blue ink, appearing to read 'Orlando Aguirre'.

Dr. Orlando Aguirre

Médico UTC

www.utc.edu.ec

Av. Simón Rodríguez s/n Barrio El Ejido / San Felipe. Tel: (03) 2252346 - 2252307 - 2252205

Anexo 4. Solicitud de asesoramiento médico



**BIENESTAR
UNIVERSITARIO**

Latacunga 22 de julio del 2020

Señores

Sandro Joel Zambrano Plusas
Rubén Darío Guambi Manitio

Estudiantes de la Carrera de Ingeniería en Electromecánica

Presente

Luego de expresarles un saludo cordial, me dirijo a ustedes y dar contestación a su pedido.

El suscrito médico de la Universidad Técnica de Cotopaxi certifica que una vez leída la solicitud de los estudiantes de la carrera de Electromecánica, brida asesoramiento Medico sobre el tema diagnóstico y tratamiento dela patología de Daltonismo

Atentamente,

“POR LA VINCULACION DE LA UNIVERSIDAD CON EL PUEBLO”

Dr. Orlando Aguirre J.
MEDICO UTC



www.utc.edu.ec

Av. Simón Rodríguez s/n Barrio El Ejido /San Felipe. Tel: (03) 2252346 – 2252307 - 2252205

Anexo 5. Validación del dispositivo a personas daltónicas.

	UNIVERSIDAD TÉCNICA DE COTOPAXI		BIENESTAR UNIVERSITARIO
---	--	---	------------------------------------

UNIVERSIDAD TÉCNICA DE COTOPAXI
FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS
INGENIERÍA ELECTROMECAÁNICA

Nombre: *GONZALO ENRIQUE HINOJOSA RODRIGUEZ*

Objetivo:
Validar el impacto del dispositivo identificador de color en personas daltónicas.

Instructivo:
Basándose en su experiencia, por favor, valore del 1 al 10 (donde 1 es "Deficiente" y 10 es "Excelente") los siguientes aspectos del Dispositivo

Facilidad para la utilización del dispositivo

1	2	3	4	5	6	7	8	9	10
									✓

Fue capaz de solucionar su problema

1	2	3	4	5	6	7	8	9	10
									✓

Y ahora valore del 1 al 10 el nivel de legibilidad del dispositivo identificador de colores en los siguientes aspectos:

Está en forma clara y fácil de entender

1	2	3	4	5	6	7	8	9	10
									✓

No está en forma clara y no es fácil de entender

1	2	3	4	5	6	7	8	9	10
✓									



UNIVERSIDAD
TÉCNICA DE
COTOPAXI



BIENESTAR
UNIVERSITARIO

Tiene intención de utilizar este dispositivo

- Si
 No

Recomendaría este dispositivo a otras personas

- Si
 NO

¡Gracias por su colaboración!



Anexo 6. Validación del dispositivo a personas daltónicas.



**UNIVERSIDAD
TÉCNICA DE
COTOPAXI**



**BIENESTAR
UNIVERSITARIO**

UNIVERSIDAD TÉCNICA DE COTOPAXI
FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS
INGENIERÍA ELECTROMECAÁNICA

Nombre: *Cristina Caicedo*

Objetivo:
 Validar el impacto del dispositivo identificador de color en personas daltónicas.

Instructivo:
Basándose en su experiencia, por favor, valore del 1 al 10 (donde 1 es "Deficiente" y 10 es "Excelente") los siguientes aspectos del Dispositivo

Facilidad para la utilización del dispositivo

1	2	3	4	5	6	7	8	9	10
									X

Fue capaz de solucionar su problema

1	2	3	4	5	6	7	8	9	10
									X

Y ahora valore del 1 al 10 el nivel de legibilidad del dispositivo identificador de colores en los siguientes aspectos:

Está en forma clara y fácil de entender

1	2	3	4	5	6	7	8	9	10
									X

No está en forma clara y no es fácil de entender

1	2	3	4	5	6	7	8	9	10
X									

www.utz.edu.ec

Av. Simón Rodríguez s/n Barrio El Ejido / San Felipe. Tel: (03) 2252346 - 2252307 - 2252205



Tiene intención de utilizar este dispositivo

Si

No

Recomendaría este dispositivo a otras personas

Si

NO

¡Gracias por su colaboración

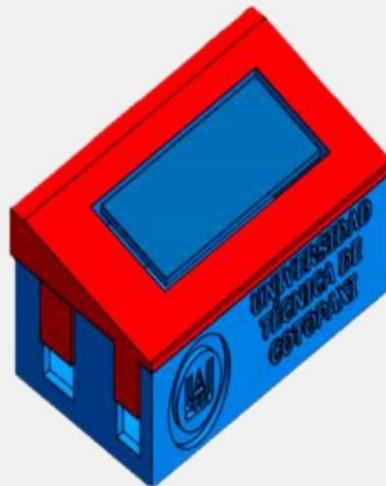
Anexo 7. Planos de la diseño y estructura.

	1	2	3	4	
A	VISTA FRONTAL		VISTA POSTERIOR		
B	VISTA LATERAL DERECHO		VISTA LATERAL IZQUIERDO		
C	ISOMETRÍA				
D					
E					
UNIVERSIDAD TÉCNICA DE COTOPAXI					
	Fecha:	Nombre:	Escala:	INGENIERÍA ELECTROMECÁNICA	
Dib.	26/06/2020	Guambi. D.Zambrano.S	1:1		
Rev.	26/06/2020	Ing. Byron Corrales M.Sc			
Apro	26/06/2020	Ing. Byron Corrales M.Sc			
Materiales:		Tolerancias:	Número Lamina	Asignatura:	Denominación:
Plástico - Impresión 3D		2 mm	N° 1	Desarrollo de Tesis	Modelo de caja del dispositivo para personas daltónicas

Anexo 8. Manual de usuario.



MANUAL DE USUARIO DEL DISPOSITIVO IDENTIFICADOR DE COLORES PARA PERSONAS DALTÓNICAS

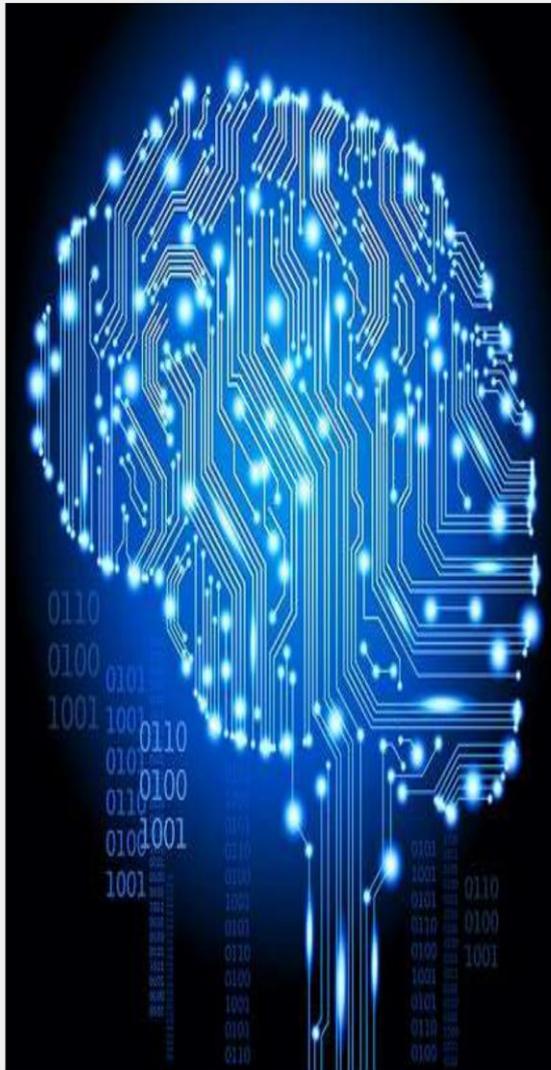


AUTORES:

Guambi Manítio Rubén Dario

Zambrano Plusas Sandro Joel

DISPOSITIVO IDENTIFICADOR DE COLORES PARA PERSONAS DALTÓNICAS



Objetivo General

Desarrollar un dispositivo de identificación de colores empleando redes neuronales y hardware libre orientado a la ayuda de personas daltónicas

Objetivos Específicos

- Investigar sobre el daltonismo y la relevancia de cada tipología del mismo en la visualización del color, mediante la búsqueda en artículos científicos, revistas y páginas web.
- Explorar el uso de redes neuronales para acoplarlos al desarrollo de los componentes del dispositivo de identificación del color.
- Diseñar dispositivo de identificación de colores (prototipo), orientado a la ayuda de personas daltónicas.
- Validar los resultados obtenidos del dispositivo para la interpretación del como el prototipo ayuda a personas que sufren de daltonismo.



Prólogo

La palabra daltonismo proviene del inglés “daltonism”, derivado del apellido del físico y químico inglés John Dalton (1766-1844), que padecía esta enfermedad y fue el primero en describirla.

La problemática social que viven las personas que sufren de daltonismo es de orden genético y no tiene cura como tal, esto incita al rechazo, vergüenza y sentimientos de incompetencia al no poder identificar los colores de manera normal. Además, los dispositivos identificadores de colores que ayuden a esta población son escasos.

El dispositivo pretende resolver esta problemática basándose, en la utilización de nuevas técnicas de control, al emplear redes neuronales y por medio del entrenamiento se logra mejores resultados de aprendizaje, finalmente se realiza las pruebas del dispositivo con los colores entrenados y verificando sus datos correctos en la pantalla LCD con sus respectivos códigos RGB y una bocina que refuerza la identificación del color.

1° Energización del dispositivo



Para la energización del dispositivo identificador de colores se debe tener en cuenta las siguientes consideraciones:

- En la placa de Hardware libre (Arduino Mega) que trabaja con dos entradas de alimentación en corriente continua: un voltaje de 5 voltios en el puerto USB y en la conectividad de alimentación de 6-12 voltios.
- Las conexiones de los demás elementos del dispositivo están conectadas al Arduino Mega, por lo que no debemos exponer al dispositivo a un voltaje superior al ya antes mencionado.
- El dispositivo puede ser conectado a la entrada USB o a la conectividad de alimentación.

2° Encendido del dispositivo

Una vez energizado el dispositivo, se encenderá de manera automática por lo cual se procede de la siguiente manera:

- En la pantalla LCD, aparece un mensaje: “DETECTOR DE COLORES”, conjuntamente con la bocina.
- Posteriormente se procederá a los mensajes de calibración del dispositivo.

Como podemos observar en la siguiente imagen.



3° Calibración del dispositivo



El dispositivo debe calibrarse para una mejor respuesta de la red neuronal al momento de utilizarse.

- Después de salir el mensaje correspondiente al encendido del dispositivo, este indicara que ingrese un color.
- Para la calibración se toma en cuenta los rangos máximos y mínimos de los colores blanco y negro, cada uno con una duración de 10 segundos respectivamente.
- Mantener por el tiempo indicado en la pantalla LCD, la superficie correspondiente a la calibración.
- Dispositivo listo para la identificación de colores que fueron entrenados en la red neuronal.

4° Finalización de la calibración del dispositivo

En la etapa final de la calibración, se procede de la siguiente manera:

- Para una respuesta más rápida y precisa la superficie a identificar debe ser lo más sólida posible.
- Los colores para las que esta entrenada la red neuronal son: negro, azul, verde, celeste, rojo, rosa, amarillo y blanco.
- Los colores que no están entrenados en la red neuronal, aparecerá un mensaje diciendo NO ESTUDIO.
- Al identificar los colores dependerá de la correcta calibración.
- En el cambio de identificación de colores no será mayor a 5 segundos.



5° Dispositivo listo para ser usado por la persona Daltónica

En este punto el dispositivo pasó un tiempo no mayor a 1 minuto para su calibración y está listo para la identificación de los colores que tendrá una salida visual y una salida auditiva.

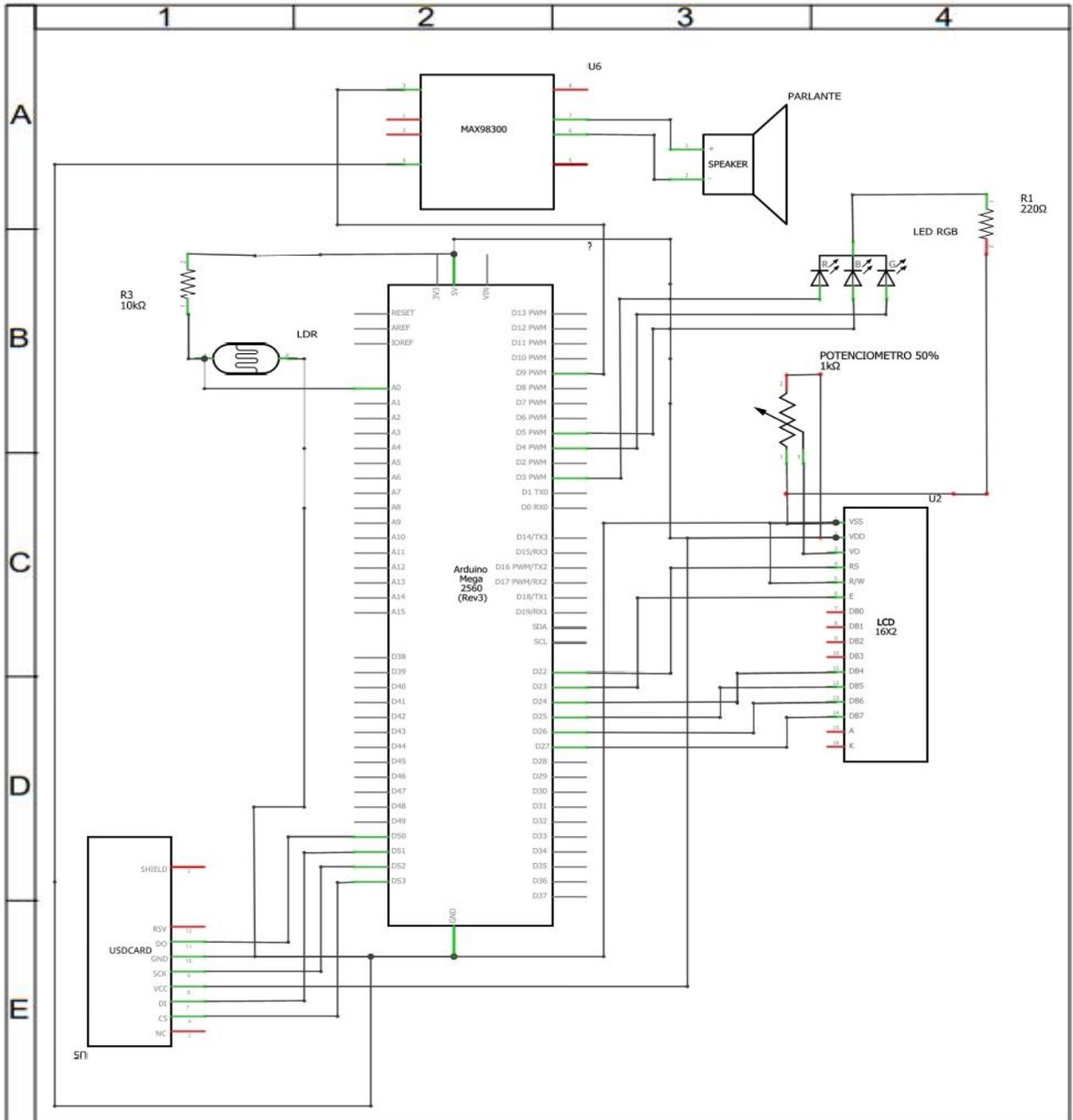


En este punto el dispositivo pasó un tiempo no mayor a 1 minuto para su calibración y está listo para la identificación de los colores que tendrá una salida visual y una salida auditiva.

Recomendaciones

- Energizar al dispositivo con un voltaje de 6-12 voltios en corriente continua por medio de la conectividad de alimentación al Arduino Mega, para que el sonido de la bocina sea más fuerte.
- Realizar una correcta calibración del dispositivo antes de cada uso.
- La superficie a identificar el color, deberá ser lo más sólida posible.
- Mantener el dispositivo en un lugar que no exista filtración de suciedad.

Anexo 9. Planos electrónicos del dispositivo.



UNIVERSIDAD TÉCNICA DE COTOPAXI



Fecha:	Nombre:	Escala:	INGENIERÍA ELECTROMECAÁNICA	
Dib. 26/06/2020	Guambi, D; Zambrano, S	1:1		
Rev. 26/06/2020	Ing. Byron Corrales M.Sc			
Apro 26/06/2020	Ing. Byron Corrales M.Sc		Asignatura:	Denominación:
Materiales: Arduino mega 2560; diodo led RGB; fotoresistencia; pantalla LCD; parlante		Tolerancias: 2 mm	Desarrollo de Tesis	Planos Eléctricos del dispositivo identificador de colores para personas daltónicas

Anexo 10. Programación del dispositivo identificador de colores.

**// UNIVERSIDAD TÉCNICA DE COTOPAXI
// CIENCIAS DE LA INGENIERÍA Y APLICADAS
// CARRERA DE INGENIERÍA ELECTROMECAÁNICA**

**// CÓDIGO PARA LA IDENTIFICACIÓN DE LOS COLORES
//RUBEN DARIO GUAMBI MANITIO
//SANDRO JOEL ZAMBRANO PLUS**

```
#include <LiquidCrystal.h>
#include <SD.h>           //incluimos las librerías necesarias
#include <SPI.h>
#include <TMRpcm.h>

const int rs = 22, en = 23, d4 = 24, d5 = 25, d6 = 26, d7 = 27;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

// Definicion de los pines
#define ldr A0
#define led_r 3
#define led_g 4
#define led_b 5

// Parametros para calibracion del sensor
#define toleranciaDeteccion 14
#define promedioSensor 14

#define SD_ChipSelectPin 53 // Seleccion del dispositivo SD
TMRpcm Audio;

int lectura_ldr[3];           // Variable para almacenar la lectura actual del sensor
int color_max[3], color_min[3]; // Variables para almacenar el valor máximo y mínimo de lectura del sensor
int ambiente_cal, objeto_cal; // Variables para almacenar la intensidad luminosa del entorno y el sensor tapado

void setup() {

  !SD.begin(SD_ChipSelectPin); //iniciamos la com con la SD respectivo

  Audio.speakerPin = 11; //Salida para el auido
  Audio.quality(1);
  Audio.setVolume(5);

  lcd.begin(16, 2);           // inicializa la lcd
  lcd.setCursor(0, 0);
  lcd.print(" DETECTOR DE");
  lcd.setCursor(0, 1);
  lcd.print(" COLORES");

  delay(200);
```

```

Audio.play("uno.wav");
delay(1200);

// Definicion de los pines del LED RGB como saída
pinMode(led_r, OUTPUT);
pinMode(led_g, OUTPUT);
pinMode(led_b, OUTPUT);

// define las salidas del LED en apagado
digitalWrite(led_r, HIGH);
digitalWrite(led_g, HIGH);
digitalWrite(led_b, HIGH);

Serial.begin(9600); // inicializa la comunicacion serial
delay(1000);

lcd.setCursor(0, 0);
lcd.print("      ");
lcd.setCursor(0, 1);
lcd.print("      ");

calibrar_sensor(); //funcion de calibracion

pinMode(9, INPUT_PULLUP);
}
int br0 = 0, br1 = 0;
void loop() {

int sensorVal = digitalRead(9);
if (sensorVal == HIGH) {
  if (br0 == 0) {
    lcd.clear();
    lcd.setCursor(4, 0);
    lcd.print("Presione");
    lcd.setCursor(4, 1);
    lcd.print("Pulsador");
    br0 = 1;
  }
} else {
  lcd.clear();
  while (analogRead(ldr) > objeto_cal); //espera el siguiente objeto
  delay(750);

  lecturaSensor(); // redirecciona a la funcion: lectura del sensor
  Colores(); // Envia los colorespor puerto serial

  while (analogRead(ldr) < ambiente_cal); // espera para remocion del objeto
  br0 = 0;
}

}

void calibrar_sensor() { // función de calibración

```

```

lcd.setCursor(0, 0);
lcd.print("  ingrese un");
lcd.setCursor(0, 1);
lcd.print("  color");
Serial.println("Aproxime cualquier objeto");
delay(2000);

ambiente_cal = promediolectura(promedioSensor); // Analiza a luminosidad del ambiente
ambiente_cal -= ambiente_cal / toleranciaDeteccion; // Tolerancia de deteccion

lcd.setCursor(0, 1);
lcd.print("          ");
Serial.println("Calibrando...");
lcd.setCursor(0, 0);
lcd.print("  calibrando  ");
delay(2000);

objeto_cal = promediolectura(promedioSensor); //analiza la luminosidad del sensor con un objeto
objeto_cal += objeto_cal / toleranciaDeteccion; //tolerancia para la deteccion

lcd.setCursor(0, 0);
lcd.print("          ");
lcd.setCursor(0, 1);
lcd.print("          ");
lcd.setCursor(0, 0);
lcd.print("  remueva el");
lcd.setCursor(0, 1);
lcd.print("  color");
Serial.println("Remueva el objeto ");
delay(1500);

while (analogRead(ldr) < ambiente_cal);           //espera para remocion del objeto

color_max[0] = 0;                                //inicio variables del color max
color_max[1] = 0;
color_max[2] = 0;

lcd.setCursor(0, 0);
lcd.print("          ");
lcd.setCursor(0, 1);
lcd.print("          ");
lcd.setCursor(0, 0);
lcd.print(" ingrese color");
lcd.setCursor(0, 1);
lcd.print("  blanco");
Serial.println("Aproxime objeto blanco");
delay(1500);

while (analogRead(ldr) > objeto_cal);           //espera el siguiente objeto
delay(750);

lcd.setCursor(0, 0);

```

```

lcd.print("      ");
lcd.setCursor(0, 1);
lcd.print("      ");
Serial.println("Calibrando...");
lcd.setCursor(0, 0);
lcd.print("  calibrando");

for (int i = 0; i < 10; i++) //genera 10 valores para encontrar el valor maximo
{
  digitalWrite(led_r, LOW);          // enciende led
  delay(100);
  lectura_ldr[0] = promediolectura(promedioSensor); // Lee el sensor y evalúa si es el valor
  if (lectura_ldr[0] > color_max[0]) {          //más alto encontrado para rojo
    color_max[0] = lectura_ldr[0];
  }
  digitalWrite(led_r, HIGH);          // Apaga led
  delay(100);

  digitalWrite(led_g, LOW);          // enciende led
  delay(100);
  lectura_ldr[1] = promediolectura(promedioSensor); // Lee el sensor y evalúa si es el valor
  if (lectura_ldr[1] > color_max[1]) {          //más alto encontrado para verde
    color_max[1] = lectura_ldr[1];
  }
  digitalWrite(led_g, HIGH);          // Apaga el led
  delay(100);

  digitalWrite(led_b, LOW);          //enciende led
  delay(100);
  lectura_ldr[2] = promediolectura(promedioSensor); // Lee el sensor y evalúa si es el valor
  if (lectura_ldr[2] > color_max[2]) {          //más alto encontrado para azul
    color_max[2] = lectura_ldr[2];
  }
  digitalWrite(led_b, HIGH);          // Apaga led
  delay(500);
}

lcd.setCursor(0, 0);
lcd.print("      ");
lcd.setCursor(0, 1);
lcd.print("      ");
Serial.println("Remueva objeto blanco");
lcd.setCursor(0, 0);
lcd.print("  remueva el");
lcd.setCursor(0, 1);
lcd.print("  color blanco");
delay(1500);

while (analogRead(ldr) < ambiente_cal);          //espera para remocion del objeto

color_min[0] = 1023;          //inicio variables del color min

```

```

color_min[1] = 1023;
color_min[2] = 1023;

lcd.setCursor(0, 0);
lcd.print(" ");
lcd.setCursor(0, 1);
lcd.print(" ");
lcd.setCursor(0, 0);
lcd.print(" ingrese color");
lcd.setCursor(0, 1);
lcd.print(" negro");
Serial.println("Aproxime un objeto negro");
delay(1500);

while (analogRead(A0) > objeto_cal); //espera el siguiente objeto
delay(500);

lcd.setCursor(0, 0);
lcd.print(" ");
lcd.setCursor(0, 1);
lcd.print(" ");
Serial.println("Calibrando...");
lcd.setCursor(0, 0);
lcd.print(" calibrando");

for (int i = 0; i < 10; i++) //genera 10 valores para encontrar el valor maximo
{
  digitalWrite(led_r, LOW);
  delay(100);
  lectura_ldr[0] = promediolectura(promedioSensor);
  if (lectura_ldr[0] < color_min[0]) {
    color_min[0] = lectura_ldr[0];
  }
  digitalWrite(led_r, HIGH);
  delay(100);

  digitalWrite(led_g, LOW);
  delay(100);
  lectura_ldr[1] = promediolectura(promedioSensor);
  if (lectura_ldr[1] < color_min[1]) {
    color_min[1] = lectura_ldr[1];
  }
  digitalWrite(led_g, HIGH);
  delay(100);

  digitalWrite(led_b, LOW);
  delay(100);
  lectura_ldr[2] = promediolectura(promedioSensor);
  if (lectura_ldr[2] < color_min[2]) {
    color_min[2] = lectura_ldr[2];
  }
}

```

```

    digitalWrite(led_b, HIGH);
    delay(100);
}

lcd.setCursor(0, 0);
lcd.print(" ");
lcd.setCursor(0, 1);
lcd.print(" ");
Serial.println("Remueva objeto negro");
lcd.setCursor(0, 0);
lcd.print(" remueva el");
lcd.setCursor(0, 1);
lcd.print(" color negro");
delay(1500);

while (analogRead(ldr) < ambiente_cal); //espera para remocion del objeto

lcd.setCursor(0, 0);
lcd.print(" ");
lcd.setCursor(0, 1);
lcd.print(" ");
Serial.println("Calibracion finalizada");
lcd.setCursor(0, 0);
lcd.print(" Calibracion");
lcd.setCursor(0, 1);
lcd.print(" Finalizada");
delay(2000);

Serial.println("empieza la lectura"); //listo para lectura
lcd.setCursor(0, 0);
lcd.print(" ");
lcd.setCursor(0, 1);
lcd.print(" ");
}

int promediolectura(int nveces) { // funcion para realizar el promedio de las lecturas
    int lectura;
    int acumulador = 0;

    for (int i = 0; i < nveces; i++) { //genera el promedio de la lectura
        lectura = analogRead(ldr);
        acumulador += lectura;
        delay(10);
    }
    return ((acumulador) / nveces); //calcula el promedio
}

void lecturaSensor() { // funcion para la lectura del sensor

    digitalWrite(led_r, LOW); // Acende o LED rojo
    delay(100); // Espera 100ms para o LDR estabilizar
}

```

```

lectura_ldr[0] = promediolectura(promedioSensor); // Realiza a leitura do sensor

digitalWrite(led_r, HIGH); // Apaga o LED rojo
delay(100); // Aguarda 100ms para o LDR estabilizar

digitalWrite(led_g, LOW); // Acende o LED verde
delay(100); // Aguarda 100ms para o LDR estabilizar

lectura_ldr[1] = promediolectura(promedioSensor); // Realiza lectura del sensor

digitalWrite(led_g, HIGH); // // Apaga o LED verde
delay(100); // Aguarda 100ms para o LDR estabilizar

digitalWrite(led_b, LOW); // Acende o LED azul
delay(100); // Aguarda 100ms para o LDR estabilizar

lectura_ldr[2] = promediolectura(promedioSensor); // Realiza a leitura do sensor

digitalWrite(led_b, HIGH); // Apaga o LED azul

}

// Rotina para enviar o valor da cor lida, convertida para a escala RGB
void Colores() {

    // Converte o valor lido do sensor para a escala RGB (0 - 255)
    unsigned char red = constrain(map(lectura_ldr[0], color_min[0], color_max[0], 0, 255), 0, 255);
    unsigned char green = constrain(map(lectura_ldr[1], color_min[1], color_max[1], 0, 255), 0, 255);
    unsigned char blue = constrain(map(lectura_ldr[2], color_min[2], color_max[2], 0, 255), 0, 255);

    // Envía os valores convertidos para a porta serial
    Serial.print("R = ");
    Serial.println(red, DEC);
    Serial.print("G = ");
    Serial.println(green, DEC);
    Serial.print("B = ");
    Serial.println(blue, DEC);

    float rojo = red / 255.00;
    float verde = green / 255.00;
    float azul = blue / 255.00;

    lcd.setCursor(0, 0);
    lcd.print("Color:");
    lcd.setCursor(0, 1);
    lcd.print("R:");
    lcd.print(red);
    if (red < 100) { //pregunto si la dist.es menor de 10
        lcd.setCursor(4, 1); //si es menor le pido que en la segunda posicion...
        lcd.print ('\32'); //me escriba un espacio
    }
    if (red < 10) { //pregunto si la dist.es menor de 10
        lcd.setCursor(3, 1); //si es menor le pido que en la segunda posicion...

```

```

    lcd.print ('\32');    //me escriba un espacio
}
lcd.setCursor(5, 1);
lcd.print("G:");
lcd.print(green);
if (green < 100) {    //pregunto si la dist.es menor de 10
    lcd.setCursor(9, 1);    //si es menor le pido que en la segunda posicion...
    lcd.print ('\32');    //me escriba un espacio
}
if (green < 10) {    //pregunto si la dist.es menor de 10
    lcd.setCursor(8, 1);    //si es menor le pido que en la segunda posicion...
    lcd.print ('\32');    //me escriba un espacio
}
lcd.setCursor(10, 1);
lcd.print("B:");
lcd.print(blue);
if (blue < 100) {    //pregunto si la dist.es menor de 10
    lcd.setCursor(14, 1);    //si es menor le pido que en la segunda posicion...
    lcd.print ('\32');    //me escriba un espacio
}
if (blue < 10) {    //pregunto si la dist.es menor de 10
    lcd.setCursor(13, 1);    //si es menor le pido que en la segunda posicion...
    lcd.print ('\32');    //me escriba un espacio
}

double REDRGB = RED(0.6560, 0.2295, 0.1334, 0.0028, rojo, verde, azul);

Serial.println(REDRGB );

if (REDRGB < 0.40)
{
    lcd.setCursor(7, 0);
    lcd.print("    ");
    delay(100);
    Audio.play("negro.wav");
    lcd.setCursor(7, 0);
    lcd.print("NEGRO");
    delay(500);
}
else if (REDRGB > 0.45 and REDRGB < 1.50)
{
    lcd.setCursor(7, 0);
    lcd.print("    ");
    delay(100);
    Audio.play("azul.wav");
    lcd.setCursor(7, 0);
    lcd.print("AZUL");
    delay(500);
}
else if (REDRGB > 1.55 and REDRGB < 2.9)
{
    lcd.setCursor(7, 0);
    lcd.print("    ");

```

```

    delay(100);
    Audio.play("verde.wav");
    lcd.setCursor(7, 0);
    lcd.print("VERDE");
    delay(500);
}
else if (REDRGB > 3 and REDRGB < 4)
{
    lcd.setCursor(7, 0);
    lcd.print("    ");
    delay(100);
    Audio.play("celeste.wav");
    lcd.setCursor(7, 0);
    lcd.print("CELESTE");
    delay(500);
}
else if (REDRGB > 4.01 and REDRGB < 4.4)
{
    lcd.setCursor(7, 0);
    lcd.print("    ");
    delay(100);
    Audio.play("rojo.wav");
    lcd.setCursor(7, 0);
    lcd.print("ROJO");
    delay(500);
}
else if (REDRGB > 4.45 and REDRGB < 5)
{
    lcd.setCursor(7, 0);
    lcd.print("    ");
    delay(100);
    Audio.play("rosa.wav");
    lcd.setCursor(7, 0);
    lcd.print("ROSA");
    delay(500);
}
else if (REDRGB > 5.05 and REDRGB < 6.45)
{
    lcd.setCursor(7, 0);
    lcd.print("    ");
    delay(100);
    Audio.play("amarillo.wav");
    lcd.setCursor(7, 0);
    lcd.print("AMARILLO");
    delay(500);
}
else if (REDRGB > 6.5)
{
    lcd.setCursor(7, 0);
    lcd.print("    ");
    delay(100);
    Audio.play("blanco.wav");
    lcd.setCursor(7, 0);

```

```

    lcd.print("BLANCO");
    delay(500);
}
else
{
    lcd.setCursor(7, 0);
    lcd.print(" ");
    delay(100);
    lcd.setCursor(7, 0);
    lcd.print("NO ESTUDIO");
    delay(500);
}
}

```

```

double ConvEntrada1(double x) {

```

```

    double max1 = 1;
    double min1 = -1;

```

```

    double xmax = 1;
    double xmin = 0;

```

```

    double Scala;
    Scala = (max1 - min1) / (xmax - xmin) * (x - xmin) + min1;

```

```

    return Scala;
}

```

```

double ConvEntrada2(double x) {

```

```

    double max1 = 1;
    double min1 = -1;

```

```

    double xmax = 1;
    double xmin = 0;

```

```

    double Scala = (max1 - min1) / (xmax - xmin) * (x - xmin) + min1;

```

```

    return Scala;
}

```

```

double ConvEntrada3(double x) {

```

```

    double max1 = 1;
    double min1 = -1;

```

```

    double xmax = 1;
    double xmin = 0;

```

```

    double Scala = (max1 - min1) / (xmax - xmin) * (x - xmin) + min1;

```

```

    return Scala;
}

```

```
double ConvSalida(double x) {
```

```
    double max1 = 7;  
    double min1 = 0;
```

```
    double xmax = 1;  
    double xmin = -1;
```

```
    double Scala = (max1 - min1) / (xmax - xmin) * (x - xmin) + min1;
```

```
    return Scala;
```

```
}
```

```
double RED (double w1, double w2, double w3, double bias, double Ent1, double Ent2, double Ent3)
```

```
{
```

```
    double Entrada1 = ConvEntrada1(Ent1);
```

```
    double Entrada2 = ConvEntrada2(Ent2);
```

```
    double Entrada3 = ConvEntrada2(Ent3);
```

```
    double z = w1 * Entrada1 + w2 * Entrada2 + w3 * Entrada3 + bias;
```

```
    double salida = ConvSalida(z);
```

```
    return salida;
```

```
}
```