

Modulo motorizado para ilustrar la difracción, atenuación y absorción de ondas electromagnéticas en el espectro infrarrojo; utilizando tecnologías libres y de bajo costo.

Diego Alberto Parra Garzón

Universidad Distrital “Francisco José de Caldas”, Facultad de Ciencias y Educación
Grupo de Física e Informática “FISINFOR”

El presente escrito, describe con detalle un vehículo motorizado que ilustra tres propiedades físicas de las ondas electromagnéticas en el espectro infrarrojo, aprovechando las nuevas tecnologías en semiconductores para el control de hardware con un microcontrolador atmega328P-PU, comunicación serial vía bluetooth con un ordenador cuyo sistema operativo es GNU-Linux, equipado con un sensor infrarrojo lateral que ilustra el fenómeno de difracción y calcula la longitud de onda emitida por el diodo led emisor en infrarrojo, tiene un par de sensores emisor-receptor infrarrojo en la parte del frente del vehículo.

El sensor infrarrojo frontal y el emisor infrarrojo frontal ilustran la ley del decaimiento de la densidad de radianas con el cuadrado de la distancia a la fuente radiante, la reflectancia y transmitancia que ocurre debido a la interacción de esta radiación con la materia. Este proyecto cuenta con las indicaciones para el uso del software de control FREEinfraROSSI y su respectiva instalación, cuya función es controlar las acciones que realiza el vehículo infraROSSI, tanto avanzar y capturar datos de los sensores lateral y frontal como de controlar la fuente radiante de fotones en el infrarrojo, para su respectivo análisis.

Cabe mencionar que toda esta comunicación entre el modulo motorizado infraROSSI y su software de control viene pre diseñada para que sea vía bluetooth. EL software de control realiza el análisis estadístico de los diferentes montajes de laboratorio para la ilustración de las propiedades de absorción, atenuación, difracción de las ondas electromagnéticas en el espectro infrarrojo, brindando una gráfica con todo el estudio sobre el experimento que se este llevando a la practica con el vehículo motorizado infraROSSI y su software de control FREEinfraROSSI .

Este proyecto esta diseñado para ser utilizado tanto por estudiantes como docentes de muy diversas ramas de las ciencias y la ingeniería o como una herramienta muy útil para los educadores y alumnos de media vocacional.

Palabras clave: Modulo motorizado infraROSSI, software de control FREEinfraROSSI, ley de decaimiento de la densidad de radianas con el inverso del cuadrado, reflectancia, software libre, GNU-Linux, difracción, comunicación serial, bluetooth, hardware libre, tesis de grado.

Agradecimientos. Agradeciendo primeramente a Dios por brindarme su sabiduría, ayuda y las mil bendiciones que nos regala cada día; a mi familia por el amor, comprensión, confianza, el cariño y por la vida que me han brindado y dedicado durante toda sus vidas, a los docentes que se esmeraron en brindarme lo mejor de ellos con mucho amor y paciencia mostrándome con sus acciones el significado de la palabra maestro, a los contribuyentes que hicieron posible mi educación y a todas las personas que trabajan en el proyecto GNU-Linux que nos brinda su sistema operativo y todos los programas openSOURCE y freeSOFTWARE como maravillosa herramienta de trabajo para los docentes y estudiantes.

Información de contacto. Diego Alberto Parra Garzón, Licenciatura en Física, Grupo de Física e Informática “FISINFOR”, Facultad de Ciencias y Educación, Universidad Distrital “Francisco Jose de Caldas”. Código estudiante: **20031135053**, diegoestudianteud1@gmail.com Bogotá D.C., Colombia 2016.

Director trabajo de grado Julián Andrés Salamanca Bernal, PhD. Profesor Universidad Distrital. Correo electrónico: jasalama@udistrital.edu.co

Índice	
INTRODUCCION	4
Experiments demonstrativos que ilustra el vehículo motorizado infraROSSI y su software FREEinfraROSSI	5
Calculo de la longitud de onda de la radiación de un diodo led infrarrojo	5
Decaimiento de la densidad de flujo de radiación infrarroja	6
Calculo del factor de perdida de la interacción radiación-materia	7
Vehículo motorizado infraROSSI	8
Modelo físico-matemático de los experimentos que ilustra el modulo infraROSSI y su software FREEinfraROSSI	11
Producción de fotones infrarrojos	11
Longitud de onda de un diodo led infrarrojo	12
Ley de decaimiento de la densidad de flujo de radiación	14
Factor de perdida de la interacción radiación-materia	14
Absorción de los fotones en el diodo receptor infrarrojo	15
Software de control FREEinfraROSSI	16
¿Qué es FREEinfraROSSI ?	16
Licencia	17
Instalación	17
Primer uso FREEinfraROSSI	18
Carga de firmware en el modulo infraROSSI	18
Diseño experimental del modulo infraROSSI	19
Disposición eléctrica	19
Control de hardware	19
Control de avance	21
Sensor de electro-recepción pasiva	22
Sensor de electro-recepción activa	23
Disposición mecánica	24
Test del modulo infraROSSI y su software FREEinfraROSSI	24
Difracción	24
Atenuación	26
Absorción	27
CONCLUSIONES	28
Referencias	30
Apéndice A. Diseño de una fuente emisora de fotones para el montaje del experimento ilustrativo de la difracción	31
Apéndice B. Calculo de la longitud de onda de la radiación de un diodo led infrarrojo	39
Apéndice C. Ilustración de la ley de decaimiento de la densidad de flujo de radiación con el cuadrado inverso de la distancia de separación a la fuente	49
Apéndice D. Calculo del factor de perdida de la densidad de flujo de radiación emitida por un diodo led emisor infrarrojo, cuando esta radiación interactúa con la materia	58
Apéndice E. Códigos del programa FREEinfraROSSI	68
Apéndice E1. LICENSE	68
Apéndice E2. copyright	81
Apéndice E3. README.md	83
Apéndice E4. instalador.py	85
Apéndice E5. INSTALADOR.py	88
Apéndice E6. unistall.py	90
Apéndice E7. infrarossi.py	92
Apéndice E8. infrarossi	97
Apéndice E9. c_Blue.sh	99
Apéndice E10. g_p_Abs.py	101
Apéndice E11. g_p_Ate.py	105
Apéndice E12. Estadistica1.m	108
Apéndice E13. Estadistica2.m	111
Apéndice E14. Atenuacion.py	114
Apéndice E15. Absorcion.py	117
Apéndice E16. Difraccion.py	120
Apéndice E17. c_p_Abs.py	123
Apéndice E18. c_p_Dif.py	126
Apéndice E19. c_p_Ate.py	129
Apéndice E20. Estadis2.py	132
Apéndice E21. o_Carpetas.py	136
Apéndice E22. o_Carpetas1.py	138
Apéndice E23. o_Carpetas2.py	140
Apéndice E24. m_Carpeta.sh	143
Apéndice E25. m_Carpetas1.sh	145
Apéndice E26. m_Carpetas2.sh	147
Apéndice E27. est_fuente.py	149
Apéndice E28. G_firmware.py	152
Apéndice E29. firmware_free_infrarossi.py	155
Apéndice E30. infrarossi.cpp	158
Apéndice E31. infrarossi.hex	168
Apéndice F. COTIZACIÓN COSTOS DE MATERIALES, PROYECTO INFRAROSSI	175
Índice de figuras	
1. Imagen del montaje para la difracción utilizando el modulo motorizado infraROSSI, una fuente emisora de fotones de $\lambda = 850\text{nm}$ y una red de difracción de 100 lineas por milímetro.	5
2. Interfaz gráfica software FREEinfraROSSI.	5

3.	Gráfica generada por el programa FREEinfraROSSI, para el laboratorio de difracción.	6	30.	Esquema tarjeta perforada con control de hardware del vehículo infraROSSI, cristal oscilador 12 MHz, dos condensadores cerámicos de 12 pF, resistencia de $1k\Omega$ y un botón; los cables de color negro son tierra, los de color rojo son voltaje, el cable azul y blanco son puentes. Figura generada en Fritzing.	19
4.	Imagen del montaje para la atenuación utilizando el modulo motorizado infraROSSI y un espejo.	6	31.	Esquema eléctrico microcontrolador atmegta 328 , cristal oscilador 12 MHz, dos condensadores cerámicos de 12 pF, resistencia de $1k\Omega$ y un botón. Figura generada en Fritzing.	19
5.	Gráfica generada por el software FREEinfraROSSI, para el laboratorio de atenuación.	7	32.	Esquema tarjeta perforada microcontrolador atmegta 328, transistor LM 7805 cv, dos condensadores electrolíticos de $10\mu F$, conectados a batería de 9 V. Figura generada en Fritzing.	20
6.	Montaje laboratorio absorción utilizando el modulo motorizado infraROSSI y un octavo de cartulina.	7	33.	Esquema eléctrico microcontrolador atmegta 328, transistor LM 7805 cv, dos condensadores electrolíticos de $10\mu F$, conectados a batería de 9V. Figura generada en Fritzing.	20
7.	Imagen generada por el programa FREEinfraROSSI, para el laboratorio de absorción.	8	34.	Esquema tarjeta perforada microcontrolador atmegta 328P-PU, modulo bluetooth hc-05, diodo led de 3mm y una resistencia de 500Ω . Figura generada en Fritzing.	20
8.	Imagen generada con la librería matplotlib de python 2.7, donde se aprecia las funciones estimadas por el programa y la función ideal de intensidad.	8	35.	Esquema eléctrico microcontrolador atmegta 328P-PU, modulo bluetooth hc-05, diodo led de 3mm y una resistencia de 500Ω . Figura generada en Fritzing.	20
9.	Imagen generada con la librería matplotlib de python 2.7, donde se aprecia una porción distingible de las funciones estimadas por el programa.	9	36.	Esquema en tarjeta perforada del microcontrolador atmegta 328P-PU, con las disposiciones del sistema de control de hardware. Figura generada en Fritzing.	21
10.	Vista superior del vehículo motorizado infraROSSI.	9	37.	Esquema eléctrico microcontrolador atmegta 328P-PU, con las disposiciones del sistema de control de hardware. Figura generada en Fritzing.	21
11.	Vista superior del vehículo motorizado infraROSSI y las bases para sus sistemas eléctricos.	9	38.	Esquema de tarjeta impresa para el microcontrolador atmegta 328P-PU, con las disposiciones del sistema de control de hardware. Figura generada en Fritzing.	21
12.	Vista superior del vehículo donde se observa el sistema de control de hardware, el sistema de control de avance del vehículo, el sensor de electro-recepción pasiva y el sensor de electro-recepción activa.	9	39.	Esquema tarjeta perforada del sistema de avance. Figura generada en Fritzing.	22
13.	Vista inferior del vehículo infraROSSI	10	40.	Esquema del circuito eléctrico del sistema de control de avance. Figura generada en Fritzing.	22
14.	Montaje en protoboard.	10	41.	Tarjeta de circuito impreso del sistema de control de avance. Figura generada en Fritzing.	22
15.	Esquema eléctrico equivalente del vehículo.	10	42.	Tarjeta perforada con el montaje del sensor de electro-recepción pasiva. Figura generada en Fritzing.	23
16.	Sistema control de hardware.	10	43.	Placa de circuito impreso para el sensor de electro-recepción pasiva. Figura generada en Fritzing.	23
17.	Sistema control de avance.	11			
18.	Sistema de electro-recepción pasiva.	11			
19.	Sistema de electro-recepción activa.	11			
20.	Transiciones con radiación entre una banda y los estados de impureza.	12			
21.	Lente doble convexa (Serway, 1997).	12			
22.	Gráfica que ofrece el software FREEinfraROSSI para el enfoque óptico de la fuente FUENinfraROSSI.	13			
23.	Reflexión y transmisión de un haz incidente.	15			
24.	Relación iradiana vs voltaje.	16			
25.	Instalación, comandos en terminal.	17			
26.	Instalación, vista navegador de archivos.	18			
27.	Instalación, propiedades FREEinfraROSSI.	18			
28.	Instalación, propiedades FREEinfraROSSI.	18			
29.	Ventana de instalación del firmware, que ofrece el software FREEinfraROSSI, para el vehículo motorizado infraROSSI.	18			

44.	Esquema tarjeta perforada del radar de electro-recepción activa sin barrido. Figura generada en Fritzing.	23
45.	Placa de circuito impreso para el sensor de electro-recepción activa. Figura generada en Fritzing.	24
46.	Chasis de madera con dimensiones en centímetros, la caja del eje delantero tiene dimensiones 2 centímetros de ancho, 3 centímetros de largo y 2.5 centímetros de alto; el eje delantero debe estar a 1 centímetro de separación del chasis.	24
47.	Montaje modulo motorizado infraROSSI para el laboratorio de difracción.	25
48.	Esquema de pines del sistema de control de hardware.	25
49.	Esquema de pines del sistema de control de avance.	25
50.	Esquema de pines del sensor de electro-recepción pasiva en infrarrojo.	25
51.	Montaje modulo motorizado infraROSSI para el laboratorio de atenuación y absorción.	26
52.	Esquema de pines del sensor de electro-recepción activa en infrarrojo.	26

Índice de tablas

1.	Análisis del experimento de difracción con el vehículo motorizado infraROSSI y su software de control FREEinfraROSSI.	6
2.	Análisis del experimento de difracción con el vehículo motorizado infraROSSI y su software de control FREEinfraROSSI.	7
3.	Análisis del experimento de absorción con el vehículo motorizado infraROSSI y su software de control FREEinfraROSSI.	8
4.	Análisis dispersión de la fuente FUENinfraROSSI con y sin sistema óptico.	13
5.	Análisis de los datos obtenidos en la calibración del sistema de control de avance.	22
6.	COTIZACIÓN COSTO DE MATERIALES, PROYECTO INFRAROSSI	175

INTRODUCCION

En la actualidad, una cantidad de instrumentos de laboratorio para la enseñanza en física y especialmente de fenómenos electromagnéticos, transporte e inyección de energía a los portadores de carga en los semiconductores, fenómenos ondulatorios, entre otros fenómenos electromagnéticos; son muy costosos, incluso estos instrumentos son exclusivos de los departamentos de ciencias en diferentes universidades y laboratorios, lo que dificulta el contacto de la población a estas manifestaciones físicas; un

caso especial es el estudio de las propiedades de las ondas electromagnéticas y su dualidad onda-partícula, en determinadas frecuencias caso clásico o en paquetes de energía discreta según su longitud de onda caso cuántico.

Específicamente en cuestión de frecuencias o longitudes de onda que corresponden al espectro electromagnético infrarrojo, concretamente cuando se habla de difracción, atenuación, absorbancia, transmitancia, reflectancia, ley de decaimiento de la densidad de radianza con el inverso del cuadrado de la distancia a la fuente; solo se tiene como marco de estudio de estos fenómenos ondulatorios infrarrojos a la espectrometría infrarroja, que es usada habitualmente por estudiantes de medicina, química, biología, entre otras; equipos muy especializados, con instrumentos muy costosos y precisos; lo que deja sin aproximarse a los demás estudiantes de ciencias exactas a estos tópicos de la física que están presentes en la vida diaria y que no son visibles al ojo humano.

El objetivo de este trabajo de grado es diseñar un instrumento de laboratorio que sea capaz de ilustrar tres fenómenos físicos de la radiación electromagnética y su dualidad onda-partícula; los cuales son: difracción en infrarrojo, reflectancia en el infrarrojo y ley del decaimiento de la irradianza con el inverso del cuadrado de la distancia a la fuente. Esto con el fin de dejar referencia que en la actualidad no se estudia de manera cómoda los fenómenos ondulatorios y de interacción de esta manifestación física de la energía con la materia, en el espectro de esta longitud de onda como lo es el infrarrojo.

Por esta razón se elabora muy cuidadosamente un instrumento que llene las expectativas de aprendices y docentes de carreras afines a estos asuntos; el cual es económico, con materiales de fácil acceso, que es capaz de aproximar al educando a fenómenos como lo es la difracción, la ley del decaimiento de la densidad de radianza con el inverso del cuadrado de la distancia a la fuente radiante, la reflectancia y transmitancia que ocurre debido a la interacción de esta radiación con la materia, en el rango infrarrojo del espectro.

A este instrumento se le denominó modulo motorizado infraROSSI, el cual es un vehículo de tracción electromagnética, con comunicación serial vía bluetooth y un ordenador GNU-Linux, equipado con un sensor infrarrojo lateral para ilustrar el fenómeno de difracción, un sensor infrarrojo frontal y un emisor infrarrojo frontal para ilustrar la ley del decaimiento de la densidad de radianza con el cuadrado de la distancia a la fuente radiante, la reflectancia y transmitancia que ocurre debido a la interacción de esta radiación con la materia; el modulo

EXPERIMENTOS DEMOSTRATIVOS QUE ILUSTRA EL VEHÍCULO MOTORIZADO INFRAROSSI Y SU SOFTWARE FREEINFRAROSSI

Calculo de la longitud de onda de la radiación de un diodo led infrarrojo

infraROSSI es capaz de acercar de manera cualitativa y cuantitativa al estudio de estos contenidos, pues aparte del instrumento de adquisición de datos, también cuenta con un software de control y análisis de datos obtenidos, a este software se le designó el nombre de software de control FREEinfraROSSI, el cual complementa el módulo motorizado infraROSSI, convirtiéndolo en una herramienta de laboratorio muy precisa, económica, fácil de utilizar, con su propio repositorio en github, con buena documentación, fácil de manipular e instalar.

Este escrito presenta en primera instancia lo que el vehículo motorizado FREEinfraROSSI puede hacer, en segunda instancia como se elaboró el vehículo y una breve documentación del software de control.

Experimentos demostrativos que ilustra el vehículo motorizado infraROSSI y su software FREEinfraROSSI

Debido a la naturaleza activa y pasiva de los sensores con los que cuenta el vehículo motorizado infraROSSI y su cualidad de moverse únicamente hacia adelante a un paso constante de 2 mm combinada con la comunicación bluetooth entre el dispositivo y un ordenador con GNU-Linux y el software de control FREEinfraROSSI pre-instalado, es posible realizar experimentos demostrativos de las propiedades de las ondas electromagnéticas en el espectro infrarrojo, los cuales se detallan a continuación.

Calculo de la longitud de onda de la radiación de un diodo led infrarrojo

En este experimento se calcula la longitud de onda propia emitida por un diodo led infrarrojo, utilizando la fuente fotónica en infrarrojo FUENinfraROSSI la cual se detalla en el anexo A. que se encuentra en la página 31, una rejilla de difracción de 100 líneas por mm, el vehículo motorizado infraROSSI y el software de control FREEinfraROSSI; se coloca el vehículo motorizado infraROSSI perpendicularmente a la fuente FUENinfraROSSI, separados una distancia de 45 cm de la fuente como se muestra en la figura 1 .

El proyecto ofrece en el anexo B. de la página 39, el artículo titulado “Calculo de la longitud de onda de la radiación de un diodo led infrarrojo, utilizando el módulo motorizado infraROSSI y su software de control FREEinfraROSSI”, el cual describe como se debe realizar las conexiones de los diferentes dispositivos de la interfaz de adquisición de datos, el uso del software FREEinfraROSSI, brindando una ayuda didáctica al lector sobre los principios físicos que suceden en el experimento, la captura y análisis de datos se realiza de una manera muy detallada y entendible para que el experimentador visualice los fenómenos físicos que ocurren.

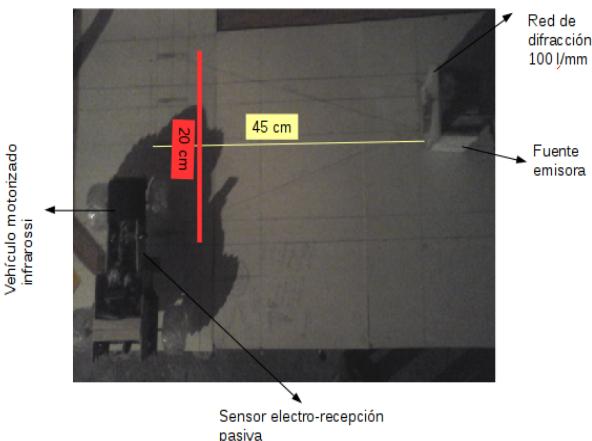


Figura 1. Imagen del montaje para la difracción utilizando el módulo motorizado infraROSSI, una fuente emisora de fotones de $\lambda = 850\text{nm}$ y una red de difracción de 100 líneas por milímetro.

Para comenzar con el experimento es necesario abrir una terminal de linux y escribir en ella `infrarossi`, dar enter y escribir la clave de superusuario; inmediatamente se abrirá la interfaz gráfica FREEinfraROSSI como se muestra en la figura 2 . Oprimir el botón `on`, y luego oprimir el botón de difracción.

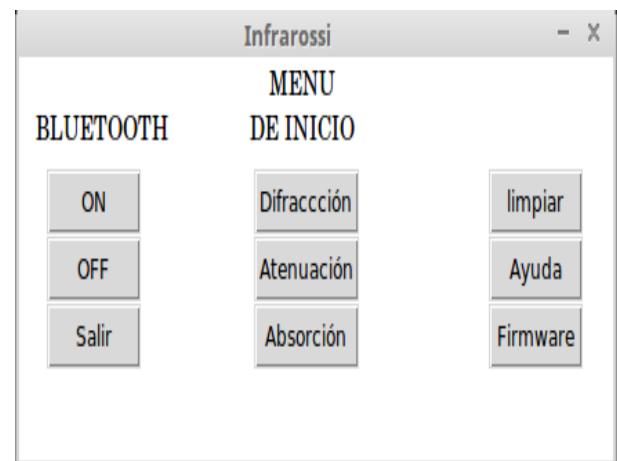


Figura 2. Interfaz gráfica software FREEinfraROSSI.

El software comienza la captura de datos del sensor de electro-recepción pasiva (Pedraja, 2012); enviando una señal vía bluetooth al vehículo infraROSSI, una vez ha recogido 140 datos el software FREEinfraROSSI le ordena al vehículo infraROSSI que avance 2mm, se detenga y capture 140 datos, esto lo realiza 94 veces hasta que halla avanzado 20 cm, luego de esto se detiene la captura de datos a través del vehículo infraROSSI, ahora el software FREEinfraROSSI realiza el análisis de datos y brinda una gráfica como se muestra en la figura 3 , con el análisis de datos del experimento de difracción.

EXPERIMENTOS DEMOSTRATIVOS QUE ILUSTRA EL VEHÍCULO MOTORIZADO INFRAROSSI Y SU SOFTWARE FREEINFRAROSSI

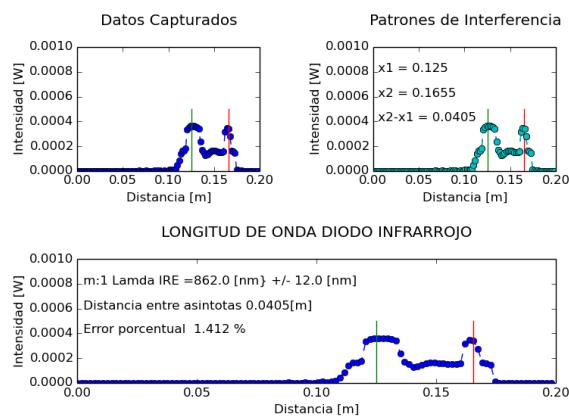


Figura 3. Gráfica generada por el programa FREEinfrarOSSI, para el laboratorio de difracción.

Este experimento se realizó 9 veces, obteniendo los resultados que se muestran en la tabla 1.

Tabla 1

Análisis del experimento de difracción con el vehículo motorizado infraROSSI y su software de control FREEinfrarOSSI.

Prueba	ΔX [m]	$\lambda_{exp.}$ [nm]	$\lambda_{teo.}$ [nm]	error %
1	0.0401	854	850	0.471
2	0.0405	862	850	1.412
3	0.0405	862	850	1.412
4	0.0405	862	850	1.412
5	0.0397	845	850	0.588
6	0.0393	837	850	1.529
7	0.0397	845	850	0.588
8	0.0393	845	850	1.529
9	0.0401	854	850	0.401

Nota. Datos obtenidos de nueve pruebas para medir la longitud de onda producida por un diodo emisor infrarrojo, con el modulo motorizado infraROSSI y su software de control FREEinfrarOSSI.

La distancia que hay entre pico y pico del patrón obtenido en la figura 3 es ΔX , esta viene en metros, $\lambda_{exp.}$ es la longitud de onda obtenida a través del experimento, viene en nanometros, $\lambda_{teo.}$ es la longitud de onda teórica del diodo emisor infrarrojo, la cual corresponde a 850nm suministrada por el fabricante¹, el error porcentual es la ultima columna de la tabla. La longitud de onda promedio de los datos en la tabla 1 es 851,77nm dando un error porcentual de 0,209 % de la longitud de onda suministrada por el fabricante.

Decaimiento de la densidad de flujo de radiación infrarroja

En este experimento se ilustra la ley de decaimiento de la densidad de flujo de radiación con el cuadrado inverso de

la distancia de separación a la fuente, utilizando un espejo, el vehículo motorizado infraROSSI y el software de control FREEinfrarOSSI; se coloca el espejo perpendicularmente al vehículo motorizado infraROSSI, separados una distancia de 30 cm como se muestra en la figura 4 .



*Figura 4.*Imagen del montaje para la atenuación utilizando el modulo motorizado infraROSSI y un espejo.

El proyecto ofrece en el anexo C. de la pagina 49, el articulo titulado “Ilustración de la ley de decaimiento de la densidad de flujo de radiación con el cuadrado inverso de la distancia de separación a la fuente; utilizando el modulo motorizado infraROSSI y su software de control FREEinfrarOSSI”, el cual describe como se debe realizar las conexiones de los diferentes dispositivos de la interfaz de adquisición de datos, el uso del software FREEinfrarOSSI, brindando una ayuda didáctica al lector sobre los principios físicos que suceden en el experimento, la captura y análisis de datos se realiza de una manera muy detallada y entendible para que el experimentador visualice los fenómenos físicos que ocurren.

Para comenzar con el experimento es necesario abrir una terminal de linux y escribir en ella **infrarossi**, dar enter y escribir la clave de superusuario; inmediatamente se abrirá la interfaz gráfica FREEinfrarOSSI como se muestra en la figura 2, oprimir el botón **on**, y luego oprimir el botón **atenuación**.

El software comienza la captura de datos del sensor de **electro-recepción activa** (Pedraja, 2012); enviando una señal vía bluetooth al vehículo **infraROSSI**, una vez ha recogido 140 datos el software FREEinfrarOSSI le ordena al vehículo **infraROSSI** que avance 2mm, se detenga

¹Enlace a la hoja de datos del diodo led emisor infrarrojo de 850nm suministrada por el fabricante online http://www.betlux.com/product/LED_lamp/infrared%20diode/BL-L314IR.PDF

y capture 140 datos, esto lo realiza 117 veces hasta que halla avanzado 30 cm, luego de esto se detiene la captura de datos a través del vehículo **infraROSSI**, ahora el software **FREEinfraROSSI** realiza el análisis de datos y brinda una gráfica como se muestra en la figura 5, con el análisis de datos del experimento de atenuación.

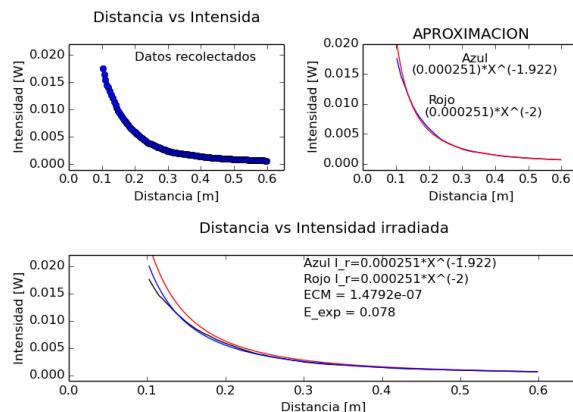


Figura 5. Gráfica generada por el software **FREEinfraROSSI**, para el laboratorio de atenuación.

Este experimento se realizó 6 veces, obteniendo los resultados que se muestran en la tabla 2.

Tabla 2

*Análisis del experimento de difracción con el vehículo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**.*

P	a ($\times 10^{-6}$)	b	T	E.C.M. ($\times 10^{-7}$)	E.E.
1	251	-1,922	-2	1.479	0.078
2	251	-1,922	-2	1.479	0.078
3	275	-1.874	-2	26.257	0.126
4	251	-1,922	-2	1.479	0.078
5	251	-1,922	-2	1.479	0.078
6	275	-1.874	-2	26.257	0.126

Nota. Datos obtenidos de seis pruebas para medir el exponente de atenuación, con el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**; *a* es la amplitud de la función, *b* es el exponente con el que se atenúa la función, *E.C.M.* es el error cuadrático medio, o también llamado bondad en el ajuste, *E.E.* es el error en el exponente.

De los datos que se muestran en la tabla 2 se obtiene la amplitud de la función densidad promedio de flujo fotónico reflejada sobre el espejo y que incide sobre el diodo led infrarrojo la cual tiene un valor de 0,000259 [W][m²], por lo que se considera este valor como la amplitud ideal para la función intensidad., también se obtiene el valor del exponente que acompaña a la distancia debido al decaimiento de esta densidad de energía sobre el espacio el cual tiene un valor promedio de -1,906 que esta muy cercano al predicho por la teoría el cual es -2, con un error

en su estimación de 4,7 %, lo cual indica que el decaimiento en la densidad de flujo electromagnético que atraviesa una superficie en la unidad de tiempo, que a su vez se refleja y por ende que llega al sistema detector del modulo motorizado **infraROSSI**, decrece con el cuadrado de la distancia.

El promedio del error cuadrático medio en los datos obtenidos por el experimentador en la tabla 2, indican que 11,286x10⁻⁵ % de estos no pueden ser explicados por este modelo físico-matemático, dejando solamente un 99,9998 % de fiabilidad en el ajuste estadístico, utilizado por el software **FREEinfraROSSI**.

Calculo del factor de perdida de la interacción radiación-materia

En este experimento se ilustra el factor de perdida de la densidad de flujo de radiación emitida por un diodo led emisor infrarrojo, cuando esta radiación interactúa con la materia; utilizando el modulo motorizado **infraROSSI**; se colocan octavos de cartulina perpendicularmente al vehículo motorizado **infraROSSI**, separados una distancia de 30 cm como se muestra en la figura 6.

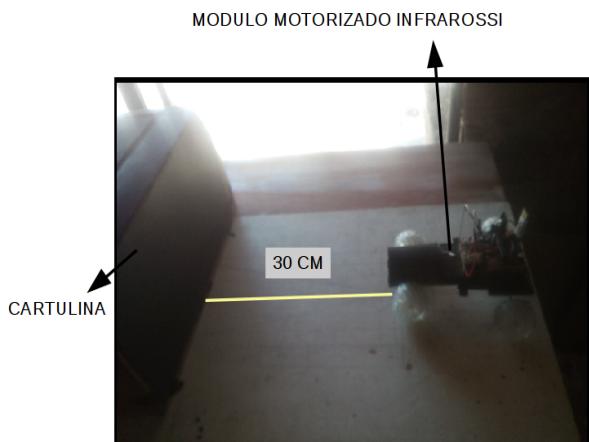


Figura 6. Montaje laboratorio absorción utilizando el modulo motorizado **infraROSSI** y un octavo de cartulina.

El proyecto ofrece en el anexo D. de la pagina 58, el articulo titulado “Calculo del factor de perdida de la densidad de flujo de radiación emitida por un diodo led emisor infrarrojo, cuando esta radiación interactúa con la materia; utilizando el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**”, el cual describe como se debe realizar las conexiones de los diferentes dispositivos de la interfaz de adquisición de datos, el uso del software **FREEinfraROSSI**, brindando una ayuda didáctica al lector sobre los principios físicos que suceden en el experimento, la captura y análisis de datos se realiza de una manera muy detallada y entendible para que el experimentador visualice los fenómenos físicos

que ocurren.

Para comenzar con el experimento es necesario abrir una terminal de linux y escribir en ella **infrarossi**, dar enter y escribir la clave de superusuario; inmediatamente se abrirá la interfaz gráfica **FREEinfraROSSI** como se muestra en la figura 2, oprimir el botón **on**, y luego oprimir el botón de absorción.

El software comienza la captura de datos del sensor de electro-recepción activa (Pedraja, 2012); enviando una señal vía bluetooth al vehículo **infraROSSI**, una vez ha recogido 140 datos el software **FREEinfraROSSI** le ordena al vehículo **infraROSSI** que avance 2mm, se detenga y capture 140 datos, esto lo realiza 117 veces hasta que halla avanzado 30 cm, luego de esto se detiene la captura de datos a través del vehículo **infraROSSI**.

El software **FREEinfraROSSI** realiza el análisis de datos y brinda una gráfica como se muestra en la figura 7, con el análisis de datos del experimento de absorción. Este experimento se realizó con 5 octavos de cartulina con superficie de diferentes pigmentos como se muestra en la tabla 3 .

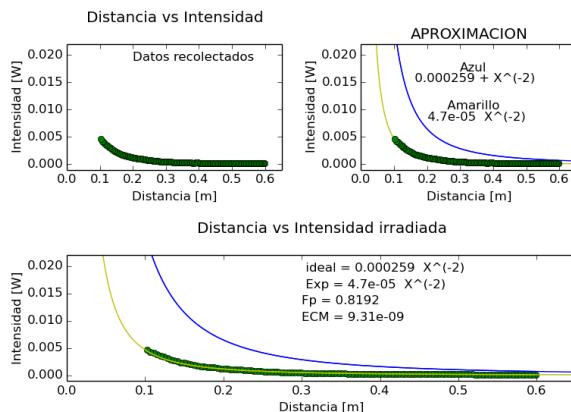


Figura 7. Imagen generada por el programa **FREEinfraROSSI**, para el laboratorio de absorción.

En la figura 7 se observan los datos capturados en el experimento, las funciones de color azul en la gráfica es intensidad ideal I_{ideal} , la función de color amarillo es la gráfica estimada $I^* = Exp$ y los puntos de color negro son los datos recolectados. En la tabla 3 y en las gráficas 8 y 9 , se muestra la función intensidad a en $[W][m^2]$, estimado por el software **FREEinfraROSSI**, para cada una de las diferentes muestras del mismo material de cartulina pero diferentes pigmentos; la cartulina en la cual hubo más transmitancia fue la de color negro, pues el 86,84 % de intensidad de la luz incidente sobre ella fue transmitida al material, llegando

solo un 13,16 % de esta al sensor.

La muestra que mayor reflectancia tuvo fue la de color rojo pues el 18,08 % de la intensidad lumínica llegó al sensor, indicando que solo el 81,92 % de esta fue transmitida al material de muestra.

Tabla 3

*Análisis del experimento de absorción con el vehículo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**.*

Color	$a (x 10^{-5})$	T	R	ECM ($x 10^{-7}$)
Azul	4.5	0.825	0.175	1.3896
Amarillo	4.3	0.8322	0.1678	0.0439
Verde	4.2	0.8392	0.1608	0.2399
Negro	3.4	0.8684	0.1316	0.332
Rojo	4.7	0.8192	0.1808	0.0931

Nota. Datos de amplitud, transmitancia y reflectancia de cinco muestras de cartulina de diferentes colores, recolectados con el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**.

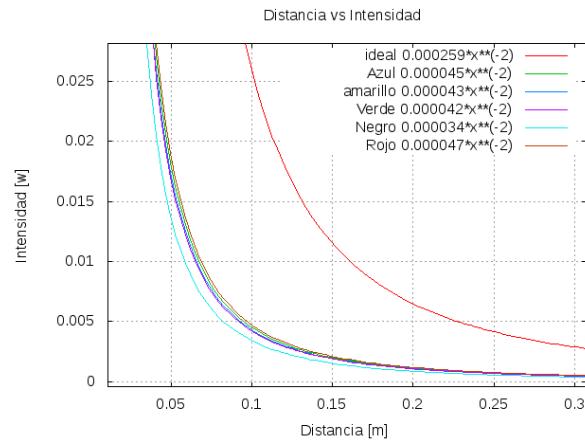


Figura 8. Imagen generada con la librería matplotlib de python 2.7, donde se aprecia las funciones estimadas por el programa y la función ideal de intensidad.

El error cuadrático medio indica que solo el 0,000004197 % del experimento no se explica con este modelo físico-matemático, dejando un 99,9999958 % de fiabilidad en el ajuste estadístico utilizado en este laboratorio por el software de control **FREEinfraROSSI** y el método estadístico que utiliza (Ostle, 1981).

Vehículo motorizado **infraROSSI**

El modulo motorizado **infraROSSI** ver figuras 10, 11, 12 y 13; es un vehículo con chasis de madera, batería de 9V del tipo cuadrada recargable, modulador de señal de 9V a 5V voltaje continuo, tracción electromagnética con

VEHÍCULO MOTORIZADO INFRAROSSI

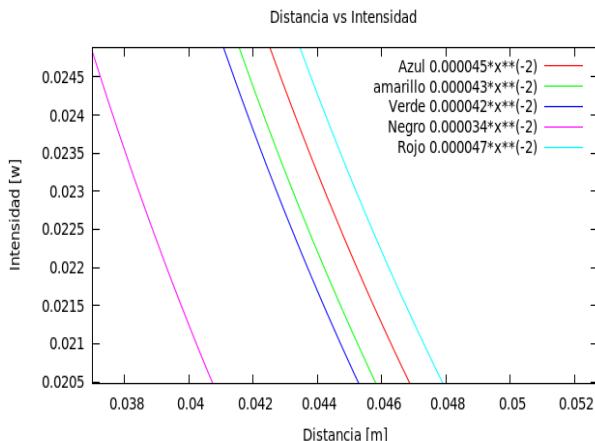


Figura 9. Imagen generada con la librería matplotlib de python 2.7, donde se aprecia una porción distinguible de las funciones estimadas por el programa.

transmisión de tipo engranaje trasera con diferencial cero de eje posterior fijo y eje delantero fijo, cuatro llantas de cinco centímetros de diámetro y dos centímetros de ancho, comunicación bluetooth entre vehículo y ordenador, dos sensores de intensidad lumínica en el rango infrarrojo del espectro electromagnético de estructura epoxy cinco milímetros, uno del tipo radar sin barrido basado en el principio de electro-recepción (Pedraja, 2012) activa con disposición espacial en la parte media del sector frontal, y en la zona lateral derecha a siete centímetros del sector frontal reposa el sensor del tipo radar sin barrido basado en el principio de electro-recepción pasiva, control de rapidez en el avance del vehículo y control de hardware del vehículo.

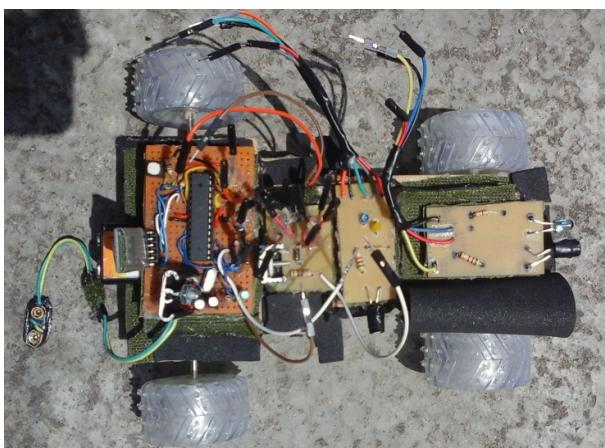


Figura 10. Vista superior del vehículo motorizado infraROSSI.

El modulo **infraROSSI** es un vehículo didáctico para la enseñanza experimental en física ver figuras 1, 4, 6, 14 y 15, los diferentes sistemas electrónicos del

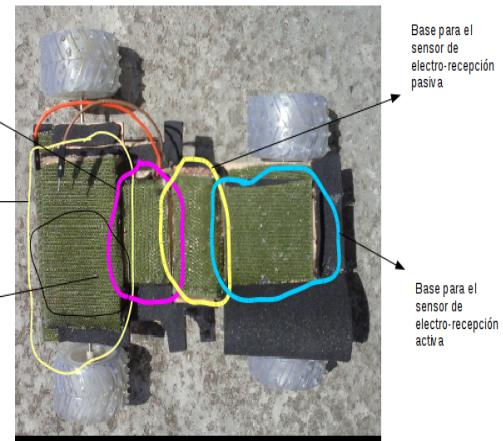


Figura 11. Vista superior del vehículo motorizado infraROSSI y las bases para sus sistemas eléctricos.

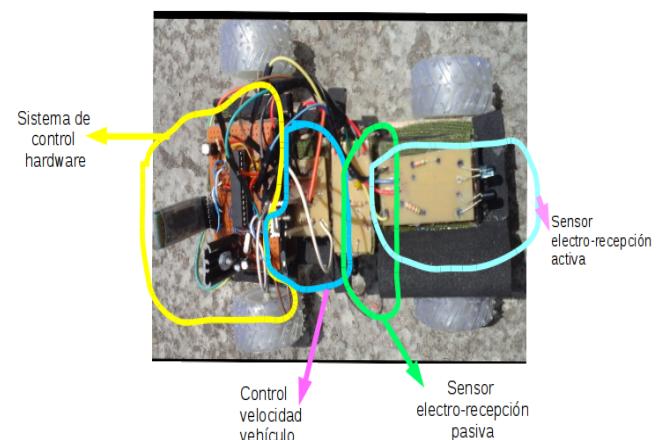


Figura 12. Vista superior del vehículo donde se observa el sistema de control de hardware, el sistema de control de avance del vehículo, el sensor de electro-recepción pasiva y el sensor de electro-recepción activa.

vehículo pueden removverse en cuatro sistemas: control de hardware y comunicación bluetooth con ordenador ver figura 16 , control rapidez del vehículo ver figura 17 , sensor electro-recepción pasiva ver figura 18 , y, un sensor de electro-recepción activa ver figura 19.

El modulo motorizado infraROSSI como se muestra en la figuras 13 y 14 , es un vehículo de tracción electromagnética, con comunicación serial vía bluetooth y un ordenador GNU-Linux, equipado con un sensor infrarrojo lateral para ilustrar el fenómeno de difracción, un sensor infrarrojo frontal y un emisor infrarrojo frontal para ilustrar la ley del decaimiento de la densidad de radianza con el cuadrado de la distancia a la fuente radiante, la reflectancia y transmitancia que ocurre debido a la interacción de esta radiación con la materia.

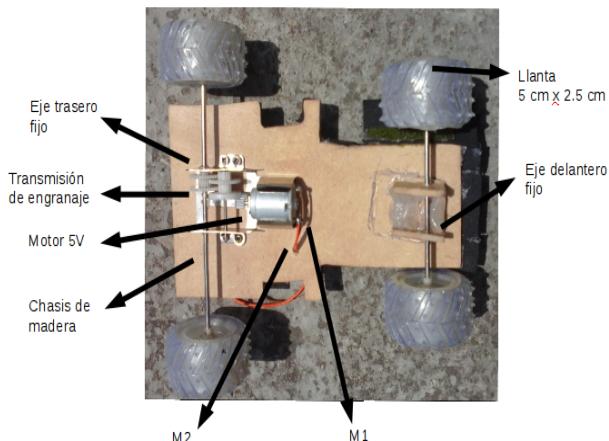


Figura 13. Vista inferior del vehículo infraROSSI

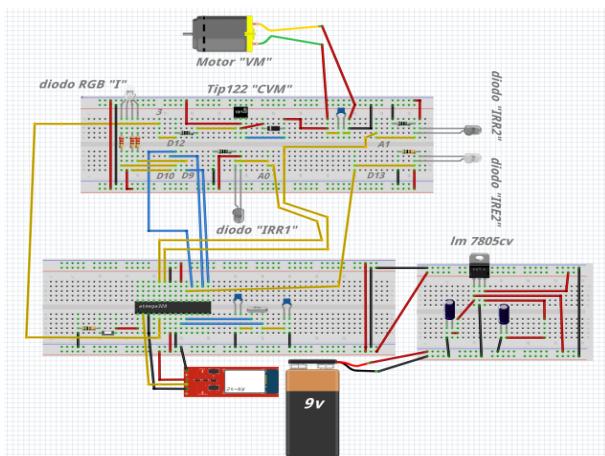


Figura 14. Montaje en protoboard.

Tanto los sistemas de control y los sensores del vehículo motorizado son móviles, la parte de abajo de las baquetas están cubiertas por velcro de color verde, esto facilita el poder desconectar y ubicar los diferentes elementos del proyecto, además con la ayuda de un voltímetro-amperímetro digital, una protoboard y los componentes eléctricos el instrumento infraROSSI sirve de modelo pedagógico para la enseñanza de electrónica digital y convencional según (Albella y Martínez-Duart, 1996).

En las figuras 10, 12, 16, 17, 18 y 19, todos los circuitos vienen con cables de conexión y en las figuras 14 y 15 están los esquemas eléctricos del proyecto, con los cuales podría demostrarse según (Albella y Martínez-Duart, 1996), el potencial eléctrico, capacitancia, corriente, resistencia, ley de Ohm, la potencia eléctrica, las leyes de Kirchhoff, generadores, motores eléctricos, teoría de diodos y circuitos integrados.

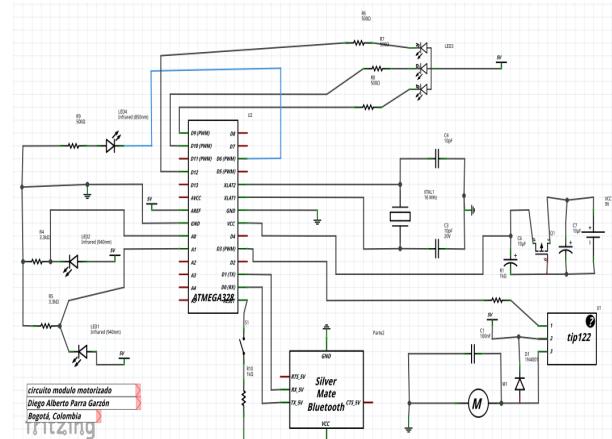


Figura 15. Esquema eléctrico equivalente del vehículo.

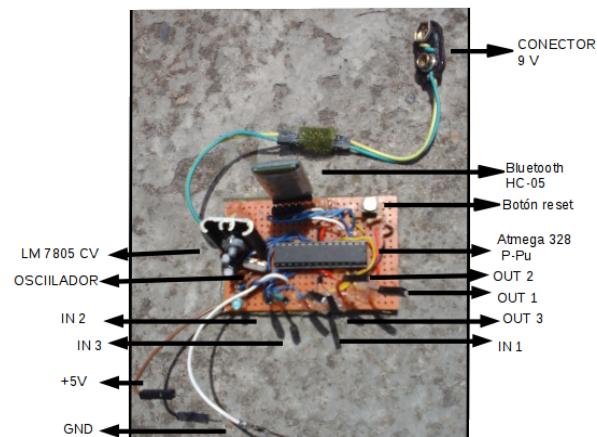


Figura 16. Sistema control de hardware.

Se elaboran cuatro artículos que detallan la manipulación del vehículo motorizado infraROSSI y las conexiones de los diferentes módulos eléctricos con los que cuenta.

El artículo titulado “Diseño de una fuente emisora de fotones en el espectro infrarrojo para el montaje del experimento ilustrativo de la difracción con el software FREEinfraROSSI y el vehículo motorizado infraROSSI” en el anexo A. página 31; describe detalladamente como se diseña una fuente fotónica en el infrarrojo de 850 nm con mínima dispersión de densidad de flujo fotónico, para poder ejecutar el laboratorio de difracción con el modulo motorizado infraROSSI y su software de control FREEinfraROSSI.

El artículo titulado “Calculo de la longitud de onda de la radiación de un diodo led infrarrojo, utilizando el modulo motorizado infraROSSI y su software de control FREEinfraROSSI” en el anexo B. página 39; se detalla

MODELO FÍSICO-MATEMÁTICO DE LOS EXPERIMENTOS QUE ILUSTRA EL MODULO INFRAROSSI Y SU SOFTWARE FREEINFRAROSSI

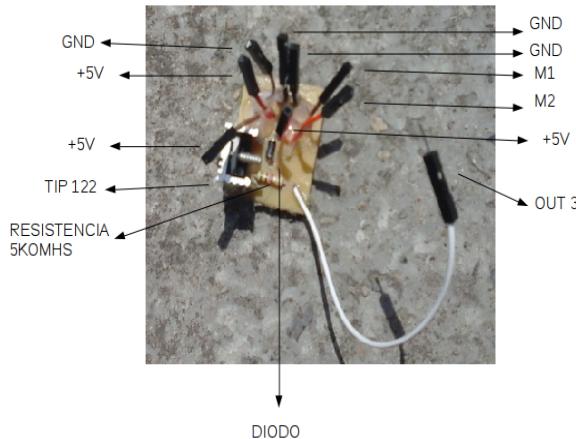


Figura 17. Sistema control de avance.

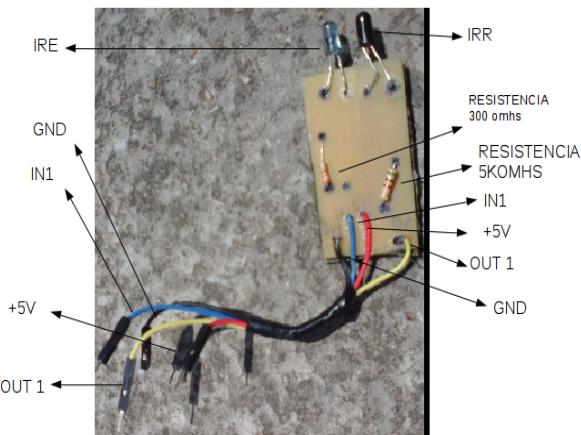


Figura 19. Sistema de electro-recepción activa.

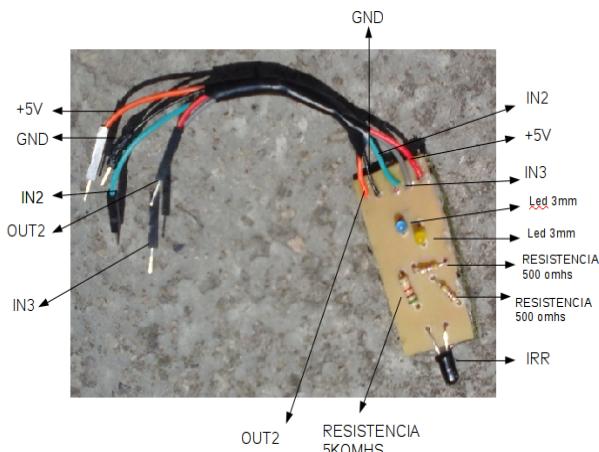


Figura 18. Sistema de electro-recepción pasiva.

el montaje de laboratorio para el experimento demostrativo de difracción con el vehículo motorizado **infraROSSI** y su software de control **FREEinfraROSSI** el cual calcula la longitud de onda de una rejilla de difracción de 100 líneas/mm.

El artículo titulado “Ilustración de la ley de decaimiento de la densidad de flujo de radiación con el cuadrado inverso de la distancia de separación a la fuente; utilizando el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**” en el anexo C. pagina 49; se detalla el montaje de laboratorio con el vehículo motorizado **infraROSSI** y el software de control **FREEinfraROSSI** para calcular el coeficiente de atenuación del flujo fotónico en el espacio el cual decrece con el inverso del cuadrado de la distancia de separación de este a la fuente, también calcular la amplitud de densidad de este flujo de fotones infrarrojos.

El artículo titulado “Calculo del factor de perdida de la

densidad de flujo de radiación emitida por un diodo led emisor infrarrojo, cuando esta radiación interactúa con la materia; utilizando el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**” anexo D. pagina 58, donde se detalla el montaje de laboratorio de absorción de radiación infrarroja debido a la interacción de esta radiación con la materia, las conexiones de los diferentes sistemas electrónicos del vehículo motorizado **infraROSSI** y con su software de control **FREEinfraROSSI** calcular la reflectancia y transmitancia de la interacción de esta radiación con la materia.

Modelo físico-matemático de los experimentos que ilustra el modulo **infraROSSI** y su software **FREEinfraROSSI**

Producción de fotones infrarrojos

Los fotones producidos por el diodo emisor de la fuente emisora² de radiación infrarroja, según (Shalímová y Grdiam, 1975) “se dan debido a que en los semiconductores, con estructura compleja de las bandas energéticas, son posibles las transiciones indirectas de los electrones de la banda de conducción a la de valencia acompañadas de la emisión de un fotón. En este caso la recombinación del electrón libre con el hueco se desarrolla con la aparición de un fonón, lo que asegura la conservación del cuasi impulso. Lo más probable es que el fonón sea emitido. Si en el semiconductor se desarrollan procesos de recombinación entre bandas tanto directa como indirectas, en el espectro de radiación se observan dos bandas de luminiscencia.

Las transiciones de los electrones de la banda de conducción a los niveles de los pequeños donadores (o de los huecos

²FUENinfraROSSI, la cual se detalla en el anexo A.A. pagina 31 y la fuente emisora del sensor de electro-recepción activa(Pedraja, 2012), el cual se muestra en la figura 19.

MODELO FÍSICO-MATEMÁTICO DE LOS EXPERIMENTOS QUE ILUSTRA EL MODULO INFRAROSSI Y SU SOFTWARE FREEINFRAROSSI

Longitud de onda de un diodo led infrarrojo

de la banda de valencia a los niveles de los pequeños aceptores), que hacen que estos últimos se neutralicen, pueden ser con radiación. En este caso es de esperar la aparición de luminosidad en la región infrarroja remota del espectro.

Pero los cálculos muestran que en estas transiciones lo más probable es que sea emitido un fonón y no un fotón, es decir, que el proceso se desarrolla sin radiación. La recombinación con radiación se produce por lo general como viene mostrado en la figura 20.

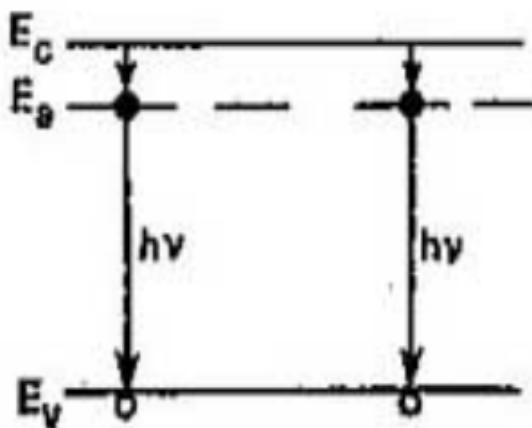


Figura 20. Transiciones con radiación entre una banda y los estados de impureza.

En la banda prohibida de los semiconductores reales existe una gran cantidad de estados localizados, que están ligados a los átomos de impureza, defectos de la estructura, infracciones de la periodicidad de la estructura en la superficie, entre otros. Estos estados localizados desempeñan un papel importante en los procesos de luminiscencia.

El estudio de los espectros de luminiscencia relacionados a diversas impurezas y defectos permite obtener información sobre estas infracciones de la estructura.

En los semiconductores se pueden producir pares electrón-hueco ligados por la atracción coulombiana, es decir, excitones. Si uno de estos pares se aniquila, se produce la emisión de un fotón. La energía de esta radiación es:

$$hv = E_g - E, \quad (1)$$

donde E es la energía de enlace del excitón."

Ahora se tiene un flujo de fotones de energía hv saliendo del diodo emisor infrarrojo, según (Gautreau, Savin, y Velazquez Valle, 2001) "como los fotones viajan a la velocidad de la luz deben, de acuerdo con la teoría de la relatividad, tener una masa en reposo igual a cero; de aquí

que su energía sea completamente cinética. Si un fotón existe, entonces se mueve a la velocidad de la luz, c ; si deja de moverse a velocidad c , deja de existir. Para $m_0 = 0$ la relación relativista momentum-energía se convierte en $E = pc$. de esta forma, cada fotón tiene un momentum de

$$p = \frac{E}{c} = \frac{hv}{c} = \frac{h}{\lambda}, \quad (2)$$

Desde el punto de vista cuántico, un haz de energía electromagnética se compone de fotones que se desplazan a la velocidad c . La intensidad del haz será proporcional al número de fotones que cruza un área unitaria por unidad de tiempo. Entonces, si el haz es monocromático (de una frecuencia), la intensidad I se obtendrá de

$$I = (hv) \times \left(\frac{N}{A \times t} \right), \quad (3)$$

h es la constante de Plank que tiene un valor de 6.626×10^{-34} J*s; N es el número de fotones que pasan por segundo a través de la superficie; A es la superficie; t es el tiempo en segundos."

Longitud de onda de un diodo led infrarrojo

Ahora los fotones de la fuente FUENinfraROSSI , salen del diodo led emisor infrarrojo y avanzan hacia el sistema óptico el cual se detalla en el anexo A. pagina 31 consiste en una lente doble convexa (Serway, 1997) y (Hecht, Dal Col, Talavera, y Pérez, 2000), la fuente emisora se coloca en el foco de la lente, pues como se muestra en la figura 21, cuando la fuente emite los fotones infrarrojos desde el punto focal, estos pasan por la lente y salen en forma paralela con menor dispersión que si no tuviera el sistema óptico como se observa en la tabla 4 y en la figura 22 .

Una vez producido los fotones infrarrojos con energía

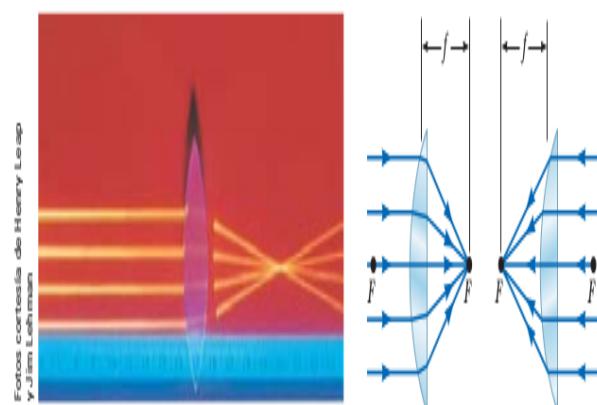


Figura 21. Lente doble convexa (Serway, 1997).

hv se hacen pasar por una red de difracción de 100 líneas por

MODELO FÍSICO-MATEMÁTICO DE LOS EXPERIMENTOS QUE ILUSTRAN EL MÓDULO INFRAROSSI Y SU SOFTWARE FREEINFRAROSSI

Longitud de onda de un diodo led infrarrojo

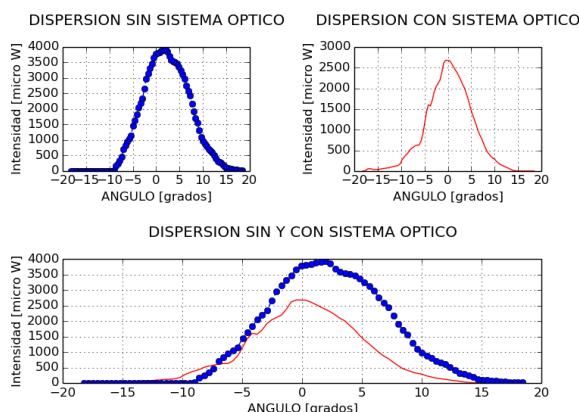


Figura 22. Gráfica que ofrece el software FREEinfraROSSI para el enfoque óptico de la fuente FUENinfraROSSI.

Tabla 4

Análisis dispersión de la fuente FUENinfraROSSI con y sin sistema óptico.

Prueba	Separación	Dispersión (Grados)
Sin sistema óptico	30 cm	10
Con sistema óptico	30 cm	5

Nota. Todos los datos son aproximados a partir de la gráfica que ofrece el software FREEinfraROSSI, la cual se observa en la figura 22.

milímetro, esta radiación en vez de comportarse como un corpúsculo como lo venia haciendo, se comporta como una onda y cada abertura de la red de difracción se convierte en una fuente, por lo que se tiene un gran numero de fuentes.

Según (Hecht et al., 2000) “Al reflejarse en esta clase de red, la luz esparcida por las varias características periódicas de la superficie llegarán a un punto P con una relación de fase definida. El patrón de interferencia correspondiente engendrado después de la reflexión es muy similar al que se produce por transmisión. Las redes diseñadas específicamente para funcionar de esta manera se denominan redes de reflexión de fase. Tradicionalmente, las redes de esta clase son rayadas sobre películas finas de aluminio que han sido evaporadas sobre bloques de vidrio ópticamente planos. Puesto que el aluminio es bastante blando, hay menos desgaste de la herramienta de rayar de diamante, siendo también mejor reflector en la región ultravioleta. Si miramos perpendicularmente a través de una red de transmisión hacia una fuente lineal paralela distante, los ojos servirían como lente de enfoque para la distribución de difracción. Como un puente simple aunque lógico entre los estudios de la interferencia y de la difracción se considera un conjunto de N osciladores puntuales coherentes (o antenas emisoras), todos ellos idénticos incluso en su polarización. Por ahora,

hay que suponer que los osciladores no tengan diferencia de fase intrínseca³, es decir, cada uno tiene el mismo ángulo de fase inicial.

Todos los rayos son casi paralelos, encontrándose en un punto P muy distante. Si la extensión espacial del conjunto es comparativamente pequeña, las amplitudes de onda individuales que lleguen a P serán esencialmente iguales, habiendo recorrido casi las mismas distancias, esto es:

$$E_0(r_1) = E_0(r_2) = \dots = E_0(r_n) = E_0(r) \quad (4)$$

La suma de los trenes de onda esféricos interferentes produce un campo eléctrico en P proporcionado por la parte real de

$$\vec{E} = E_0(r)e^{i(kr_1 - \omega t)} + E_0(r)e^{i(kr_2 - \omega t)} + \dots + E_0(r)e^{i(kr_N - \omega t)} \quad (5)$$

Por tanto ahora

$$\vec{E} = E_0(r)e^{(-i\omega t)}e^{ikr_1} \times [1 + e^{ik(r_2 - r_1)} + e^{ik(r_3 - r_1)} + \dots + e^{ik(r_N - r_1)}] \quad (6)$$

La diferencia de fases entre fuentes adyacentes se obtiene de la expresión $\delta = k_0\Lambda$, y puesto que $\Lambda = nd\sin(\theta)$, en un medio con índice n, $\delta = kds\sin(\theta)$, de esto se deduce que $\delta = k(r_2 - r_1)$, $2\delta = k(r_3 - r_1)$.

Entonces el campo de P puede escribirse como:

$$\vec{E} = E_0(r)e^{(-i\omega t)}e^{ikr_1} \times [1 + (e^{i\delta}) + (e^{i\delta})^2 + (e^{i\delta})^3 + \dots + (e^{i\delta})^{N-1}] \quad (7)$$

La serie geométrica entre paréntesis tiene el valor:

$$(e^{i\delta N} - 1)/(e^{i\delta} - 1)$$

que puede ordenarse así:

$$\frac{e^{i\delta N/2}[e^{i\delta N/2} - e^{-i\delta N/2}]}{e^{i\delta/2}[e^{i\delta/2} - e^{-i\delta/2}]} \quad (8)$$

o de manera equivalente

$$e^{i(N-1)\delta/2} * \left(\frac{\sin(N\delta/2)}{\sin(\delta/2)} \right) \quad (9)$$

Entonces el campo se transforma en:

$$\vec{E} = E_0(r)e^{(-i\omega t)}e^{i[kr_1 + (N-1)\delta/2]} * \left(\frac{\sin(N\delta/2)}{\sin(\delta/2)} \right) \quad (10)$$

Si se define R como la distancia desde el centro de la linea de los osciladores hasta el punto P, es decir:

$$R = \frac{1}{2}(N-1)ds\sin(\theta) + r_1 \quad (11)$$

³Este es el caso ideal, el cual en el experimento se da el caso que los fotones infrarrojos producidos por el diodo led infrarrojo no están en fase.

Entonces la ecuación 10 se convierte en:

$$\vec{E} = E_0(r) e^{i(kR - \omega t)} \left(\frac{\sin(N\delta/2)}{\sin(\delta/2)} \right) \quad (12)$$

Finalmente, la distribución de densidad de flujo dentro de la distribución de difracción debida a N fuentes puntuales distantes, idénticas y coherentes en una disposición lineal, es proporcional a $EE^*/2$ para E compleja o

$$I = I_0 \frac{\sin^2(N\delta/2)}{\sin^2(\delta/2)} \quad (13)$$

donde I_0 es la densidad de flujo que saliendo desde cualquier fuente puntual llegue a P.

La dependencia funcional de I con θ queda más clara en la forma

$$I = I_0 \frac{\sin^2[(Nk\delta/2)\sin(\theta)]}{\sin^2[(k\delta/2)\sin(\theta)]} \quad (14)$$

El término $\sin^2[(Nk\delta/2)\sin(\theta)]$ se somete a unas fluctuaciones rápidas, mientras que las fluctuaciones que la modula, $\sin^{-2}[(k\delta/2)\sin(\theta)]$, varía de manera relativamente lenta. La expresión combinada da lugar a una serie de picos principales agudos separados por picos pequeños complementarios.

La ecuación 15, es la que describe el fenómeno físico y se denomina ecuación de red para incidencia normal.

$$d * \sin(\theta_m) = m\lambda \quad (15)$$

Los valores de m especifican el orden de diversos máximos principales. Para una fuente que tenga un espectro continuo ancho, la imagen de orden cero, $m = 0$, corresponde a la imagen blanca de la fuente no desviada $\theta_0 = 0$.

La ecuación de red depende de λ y así, para cualquier valor de $m \neq 0$, las distintas imágenes coloreadas de la fuente correspondientes a ángulos ligeramente diferentes (θ_m), se dispersa en un espectro continuo.

Las regiones ocupadas por los débiles máximos secundarios aparecerán como bandas aparentemente desprovistas de luz. El espectro de primer orden $m = \pm 1$ aparece a cada lado de $\theta = 0$ y es seguido, junto con intervalos alternados de oscuridad, por los espectros de orden superior, $m = \pm 1, \pm 2, \dots$

Ahora se produce un patrón de difracción que alcanza al detector infrarrojo del modulo motorizado infraROSSI como se observa en la figura 1, el cual avanza en linea recta a 45 centímetros de la fuente emisora de fotones infrarrojos, perpendicular a la incidencia de los patrones de difracción.

Ley de decaimiento de la densidad de flujo de radiación

Esta radianza de fotones desde el diodo emisor como se muestra en la figura 19, avanza por el espacio proyectando un ángulo sólido hasta un espejo que se encuentra a 30 cm de la fuente como se observa en la figura 4, por lo que la irradiancia será igual al cociente de la radianza con el ángulo sólido proyectado sobre el espejo.

El flujo de fotones que interactúa con la superficie del material de muestra dan como resultado desde el punto de vista clásico varios fenómenos físicos como lo son la reflexión, refracción, absorción, atenuación, reflectancia, transmitancia. Ahora el flujo de fotones es visto como el flujo de ondas electromagnéticas, las que interactúan con la materia en donde parte de la energía es transmitida al material aumentando la energía cinética media de sus constituyentes, otra parte traspasa el material y el resto del flujo electromagnético avanza paralelamente contrario a la dirección de desplazamiento inicial, si no se considera interacción de las ondas electromagnéticas con el aire como medio disipativo y otras formas de perdida de energía.

En este trabajo solo se tiene en cuenta el fenómeno de reflexión de la luz con el espejo, en el caso ideal que no exista perdida de energía entre la luz incidente en el espejo; osea que la intensidad del rayo incidente sea igual a la intensidad de la luz emitida por la interfaz del espejo. Solo se tiene en cuenta la atenuación de energía debido al avance de la expansión geométrica por el espacio, debido al flujo de luz emitida por el diodo emisor infrarrojo.

Ahora la luz que sale del espejo tendrá que recorrer nuevamente 30 cm hasta el detector, mientras avanza seguirá decayendo su intensidad con el inverso del cuadrado de la distancia.

Factor de perdida de la interacción radiación-materia

Esta radianza de fotones desde el diodo emisor, avanza por el espacio proyectando un ángulo sólido, por lo que la irradiancia será igual al cociente de la radianza con el ángulo sólido proyectado.

La máxima distancia que se toma para la radianza de fotones como se observa en la figura 6, es de 30 centímetros desde el diodo emisor hasta el material de muestra⁴ colocado perpendicular al flujo de energía, por lo que habrá reducido su intensidad con el inverso del cuadrado de la distancia de la fuente a la muestra $\frac{1}{(0,30)^2}$.

⁴La muestra son octavos de cartulina de colores.

Ahora el flujo de fotones que interactua con la superficie del material de muestra dan como resultado desde el punto de vista clásico varios fenómenos físicos como lo son la reflexión, refracción, absorción, atenuación, reflectancia, transmitancia. Ahora el flujo de fotones es visto como el flujo de ondas electromagnéticas, las que interactúan con la materia en donde parte de la energía es transmitida al material aumentando la energía cinética media de sus constituyentes, otra parte traspasa el material y el resto del flujo electromagnético avanza paralelamente contrario a la dirección de desplazamiento inicial, si no se considera interacción de las ondas electromagnéticas con el aire como medio disipativo y otras formas de perdida de energía.

Según (Hecht et al., 2000) "Consideremos un haz de luz circular que incide en una superficie, tal como se muestra en la figura 23, de tal modo que se produzca una zona iluminada cuya área sea A . Recordemos que la potencia por unidad de área que cruza una superficie en el vacío cuya normal es paralela a \mathbf{S} , el vector de Poynting viene determinado por:

$$\mathbf{S} = c^2 \epsilon_0 \mathbf{E} \times \mathbf{B}. \quad (16)$$

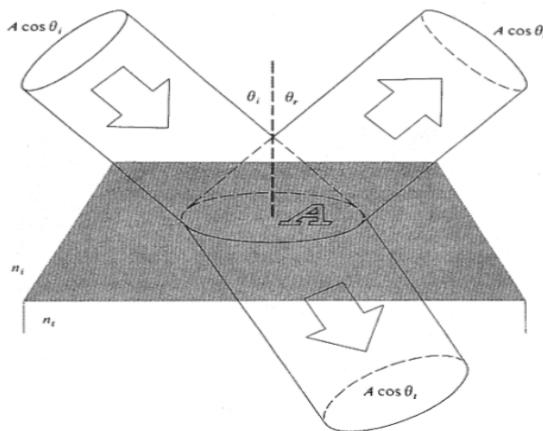


Figura 23. Reflexión y transmisión de un haz incidente.

A demás la densidad de flujo radiante (W/m^2) o irradiancia es entonces

$$I = \langle S \rangle_t = \frac{c\epsilon_0}{2} E_0. \quad (17)$$

Este el promedio de energía por unidad de tiempo que cruza un área unidad, normal a \mathbf{S} (en medios isótropos \mathbf{S} es paralela al vector de onda \mathbf{k}). En el caso que nos ocupa (figura 23) sean I_i , I_r y I_t las densidades de flujo incidente, reflejado y transmitido, serán respectivamente, $A\cos(\theta_i)$, $A\cos(\theta_r)$ y $A\cos(\theta_t)$.

De acuerdo con esto, la potencia incidente es $I_i A\cos(\theta_i)$,

esta es la energía por unidad de tiempo que fluye en el rayo incidente y, por consiguiente, la potencia que llega a la superficie de A . Del mismo modo, $I_r A\cos(\theta_r)$, es la potencia en el rayo reflejado, e $I_t A\cos(\theta_t)$, es la potencia que se transmite a través de A . Definimos la **reflectancia R** como el cociente entre la potencia (o flujo) reflejada y la potencia incidente, es decir:

$$R \equiv \frac{I_r A\cos(\theta_r)}{I_i A\cos(\theta_i)} = \frac{I_r}{I_i}, \quad (18)$$

Del mismo modo la **transmitancia T** se define como el cociente entre el flujo transmitido y el flujo incidente y viene dada por:

$$T \equiv \frac{I_t A\cos(\theta_t)}{I_i A\cos(\theta_i)} = \frac{I_t}{I_i}, \quad (19)$$

escribamos ahora una ecuación que represente la conservación de energía para la configuración que se muestra en la figura 23. Dicho de otro modo, la energía total que llega al área A por unidad de tiempo debe ser igual a la energía que fluye hacia fuera de ella por unidad de tiempo:

$$I_i A\cos(\theta_i) = I_r A\cos(\theta_r) + I_t A\cos(\theta_t), \quad (20)$$

multiplicando a ambos lados por c esta expresión queda:

$$n_i E_{0i}^2 \cos(\theta_i) = n_r E_{0r}^2 \cos(\theta_r) + n_t E_{0t}^2 \cos(\theta_t)$$

Dividiendo a ambos lados por $n_i E_{0i}^2 \cos(\theta_i)$ obtenemos:

$$1 = \left(\frac{E_{0r}}{E_{0i}} \right)^2 + \left(\frac{n_t \cos(\theta_t)}{n_i \cos(\theta_i)} \right) \left(\frac{E_{0t}}{E_{0i}} \right)^2, \quad (21)$$

Pero esto es simplemente:

$$R + T = 1, \quad (22)$$

En donde n_i es el índice de refracción del medio del que incide, n_t es el índice de refracción en el medio transmitido."

Absorción de los fotones en el diodo receptor infrarrojo

Al iluminar el diodo receptor infrarrojo con esta energía radiante, según (Albella y Martínez-Duart, 1996)"en el semiconductor por cada fotón absorbido se rompe un enlace y se crea un par electrón-hueco. Es importante destacar que no todos los portadores fotogenerados contribuyen a la conducción, ya que una fracción importante de ellos se recombinan antes de llegar al extremo correspondiente del semiconductor. El calculo del incremento de corriente ΔI_e , debida al exceso de electrones generados en la banda de conducción, Δn , es

$$\Delta I_e = q\mu_e(\Delta n)ES \quad (23)$$

siendo E el campo eléctrico aplicado, μ_e la movilidad de los electrones y S la sección transversal del fotoconductor.

En condiciones de iluminación, el estado estacionario se alcanza cuando la velocidad de generación de portadores en todo el volumen del semiconductor, G , se iguala a la velocidad de recombinación, R , es decir $R = G$. para un conductor intrínseco en el cual existe un exceso de portadores, $\Delta n = \Delta p$, la velocidad de recombinación de los portadores vendrá dada por:

$$R = \frac{\Delta n}{\tau} = \frac{\Delta p}{\tau} \quad (24)$$

siendo τ el tiempo de vida media de los portadores fotogenerados. En un semiconductor de longitud L en el que suponemos que el espesor es suficiente para que toda la luz que incide sobre el, sea absorbida en su interior, se tiene ahora para la velocidad de generación de portadores en la banda de conducción:

$$G = \eta n_{fot} = \eta \frac{P_i}{h\nu} \frac{1}{SL} \quad (25)$$

siendo n_{fot} el número de fotones incidentes en el semiconductor por unidad de volumen y de tiempo, y η la eficiencia de la conversión en la generación de portadores. El valor n_{fot} se calcula a través del cociente entre la potencia de la luz incidente, P_i , y la energía de la radiación, $h\nu$, dividido a su vez por el volumen del material.

Sabiendo que la velocidad de arrastre de los electrones por el campo eléctrico viene dada por: $v_e = \mu_e E$, las igualdades anteriores permiten escribir para la corriente de electrones fotogenerada entre los dos electrodos:

$$\Delta I_e = qv_e \eta \frac{P_i}{h\nu} \frac{\tau}{L} \quad (26)$$

si se tiene en cuenta que el cociente $t_r = L/v_e$, representa el tiempo de tránsito de los electrones entre los dos electrodos, resulta para ΔI_e :

$$\Delta I_e = q\eta \frac{P_i}{h\nu} \frac{\tau}{t_r} \quad (27)$$

con una expresión similar para la corriente de huecos en la banda de valencia. En la ecuación 27, el factor $q\eta(P_i/h\nu) = I_{fot}$ tiene dimensiones de corriente y representa la velocidad de generación de carga en el semiconductor. En función de este parámetro, se define el factor de ganancia del fotoconductor a través del cociente:

$$\frac{\Delta I}{I_{fot}} = \frac{\tau}{t_r} \quad (28)$$

ahora bien un diodo operando con cierto voltaje aplicado, V , en presencia de radiación electromagnética capaz de excitar portadores a través de la banda prohibida dejara pasar una intensidad I dada por:

$$I = I_0 [e^{(qV/kT)} - 1] - I_L \quad (29)$$

donde I_0 representa la corriente típica de un diodo, I_L representa la corriente debida a los portadores generados. El valor de I_L puede calcularse de la siguiente manera:

$$I_L = qGS(L_e - L_h) \quad (30)$$

siendo G el número de portadores generados por unidad de volumen y de tiempo y S el área de la sección transversal del diodo. L_e y L_h representan las longitudes de difusión de los electrones y huecos. El dispositivo funciona entonces como detector del nivel de iluminación convirtiendo una señal óptica en señal eléctrica."

Como la irradiancia de fotones infrarrojos es proporcional al inverso del cuadrado de la distancia de separación a la fuente, y la función voltaje que mide el modulo motorizado infraROSSI es proporcional al inverso del cuadrado de la distancia de separación de la fuente al vehículo infraROSSI; se realiza el cociente de la intensidad irradiada en el diodo led que es según el fabricante de $20 \text{ mW}/\Omega$, sobre el voltaje producido en el sensor que es de 5000mV ; como se muestra en la figura 24.

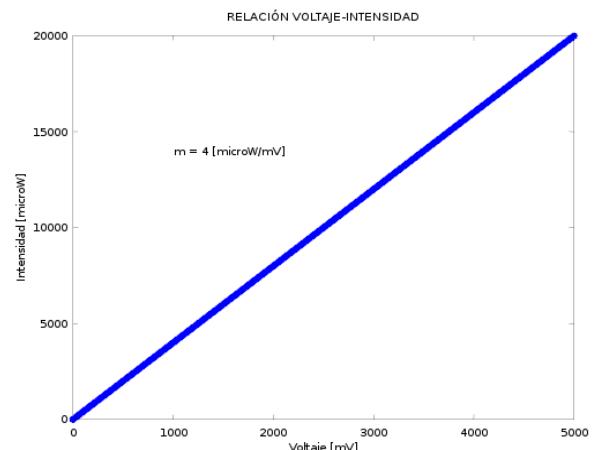


Figura 24. Relación irradianza vs voltaje.

Una vez absorbidos los fotones infrarrojos, el microcontrolador atmega 328 del vehículo motorizado infraROSSI, mide la relación de voltaje en el sensor de electro-recepción activa que es proporcional a la irradiancia del emisor infrarrojo la cual equivale a $4\mu\text{W}/\text{mV}$, este valor es la pendiente de la gráfica 24; lo que significa que por cada $4\mu\text{W}$ de radiación incidente el sensor esta arrojando en su salida 1mV .

Software de control FREEinfraROSSI

¿Qué es FREEinfraROSSI ?

FREEinfraROSSI es un programa creado en Colombia, en la ciudad de Bogotá, como tesis de grado para optar por el título de licenciado en física en la Universidad

Distrital Francisco José de Caldas, el cual fue desarrollado con software libre y en un entorno libre como lo es GNU-Linux, puede llegar a ser multiplataforma⁵ al ejecutar el programa en una maquina virtual⁶ con un sistema operativo GNU-Linux; se vale de lenguajes de programación, como C, C++ (Alcocer, Gomez, Prat, y Albareda, 2006), python 2.7 (McKinney, 2012), bash (Burtch, 2004); y de programas como arduino(Herrador, 2009), octave (Eaton, Bateman, y Hauberg, 1997) , gnuplot (Kathirvel y Srinivasan, 2009), blueman-manager (Hoile et al., 2014); entre otros, fusionados en una interfaz amigable y fácil de utilizar.

FREEinfraROSSI es el software de control del instrumento de laboratorio infraROSSI que ilustra la propiedad de difracción, atenuación y absorción de ondas electromagnéticas en el espectro infrarrojo, esta diseñado para ser utilizado tanto por estudiantes como docentes de muy diversas ramas de las ciencias y la ingeniería o como una herramienta muy útil para los educadores y alumnos de media vocacional.

Licencia

Programa de control de hardware e ilustración física de las propiedades de las ondas electromagnéticas en el espectro infrarrojo.

Copyright (C) 2016-01-01 Universidad Distrital Francisco Jose de Caldas, Diego Alberto Parra Garzón, PhD. Julian Andres Salamanca Bernal.

El programa FREEinfraROSSI es software libre; puedes redistribuirlo y / o modificarlo bajo los términos de la Licencia Pública General GNU publicada por la Fundación para el Software Libre; ya sea la versión 3 de la Licencia, o (a su elección) cualquier versión posterior. Este programa se distribuye con la esperanza de que sea útil, pero SIN NINGUNA GARANTÍA; ni siquiera la garantía implícita de COMERCIALIZACIÓN o IDONEIDAD PARA UN PROPÓSITO PARTICULAR. Vea la Licencia Pública General GNU para más detalles. Debería haber recibido una copia de la Licencia Pública General de GNU junto con este programa; si no, escriba a la Free Software Foundation, Inc., 51 Franklin Street, Quinto Piso, Boston, MA 02110-1301 EE.UU..

Si usted hace alguna modificación en esta aplicación, deberá siempre mencionar el autor original de la misma.

Instalación

La instalación de este software es el mismo para distribuciones basadas en Debian⁷ como lo es UBUNTU 14.04LTS⁸ y Linux MINT 17.2⁹.

Abrir una terminal¹⁰ como se observa en la figura 25 y escribir sin comillas “sudo su” y luego oprimir la tecla enter; escribir la contraseña de administrador, presionar enter; escribir sin comillas ”aptitude install -y git”, presionar enter; escribir sin comillas “cd Documentos” presionar enter; escribir sin comillas “git clone https://github.com/Diego-debian/Free-infrarossi”, oprimir enter; escribir sin comillas “chmod +777 Free-infrarossi”, presionar la tecla enter.

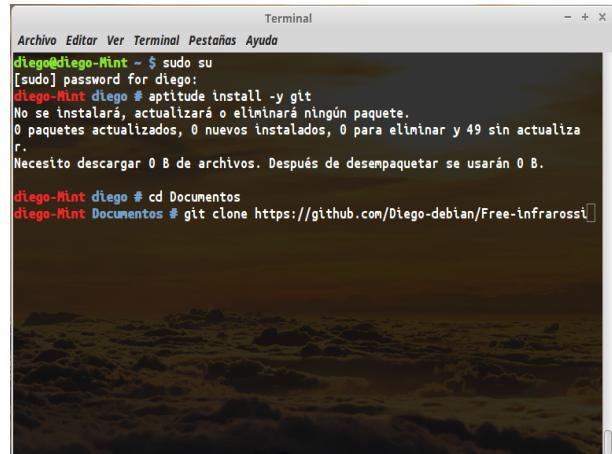


Figura 25. Instalación, comandos en terminal.

Cerrar la terminal y dirigirse al navegador de archivos en la carpeta Documentos/Free-infrarossi/install, como se observa en la figura 26; dar click derecho del mouse en el archivo instalador.py como se observa en las figuras 27 y 28 , click izquierdo en la opción propiedades; en la pestaña de general debe decir abrir archivo con python 2.7 y en la pestaña permisos debe estar seleccionada la casilla permitir ejecutar este archivo como programa, de no ser así, cambiar las opciones; cerrar la ventana y abrir el archivo INSTALADOR.py ; escribir la clave administrador y presionar enter; escribir 1 y presionar enter; escribir nuevamente 1 y enter; una vez finalizada la instalación hay que reiniciar el pc.

En caso de que en el sistema operativo UBUNTU 14.04 LTS

⁵Correr bajo cualquier sistema operativo como GNU-Linux, windows, MAC, Solaris, entre otros.

⁶Enlace online a Virtualbox <https://www.virtualbox.org/> y Vmware <http://www.vmware.com/>.

⁷Página oficial proyecto Debian <https://www.debian.org/>

⁸Enlace página oficial sistema operativo UBUNTU14.04LTS <http://www.ubuntu.com/>

⁹Enlace página oficial sistema operativo Linux MINT 17.2 <https://linuxmint.com/>

¹⁰Procedimiento: <http://www.comoinstalarlinux.com/como-abrir-una-terminal-en-ubuntu-linux-mint-centos-debian/>



Figura 26. Instalación, vista navegador de archivos.

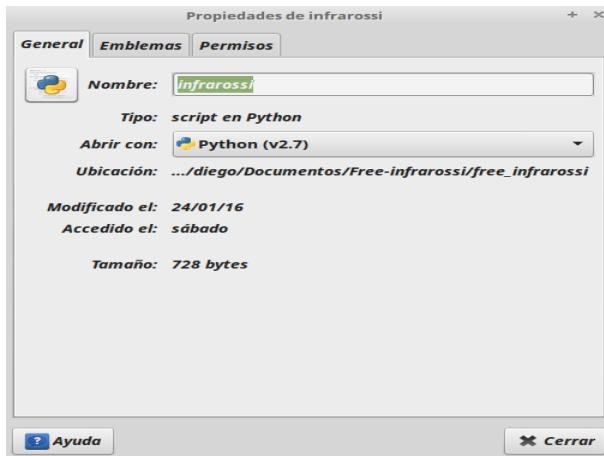


Figura 27. Instalación, propiedades FREEinfrarossi.

se produzca un error en la instalación deberá ejecutar en la terminal el siguiente comando sin comillas “`sudo python /Documentos/Free-infrarossi/install/installador.py`”, de esta manera comienza la instalación del programa.

Este software se testeó en diferentes sistemas operativos como UBUNTU 14.04 LTS, Linux MINT17.2, Debian 8, Kali Linux, Tails sin error alguno en el software, también se realizó el test sobre Windows xp, windows 7, windows vista, windows 8, windows 10 en máquina virtual con VirtualBox sin ningún error por parte del programa FREEinfrarossi.

Los diferentes scripts que componen este software se observan en el anexo E. pagina 68.

Primer uso FREEinfrarossi

Abrir una terminal del S.O. escribir en la terminal sin comillas '`infrarossi`', oprimir la tecla enter, escribir la

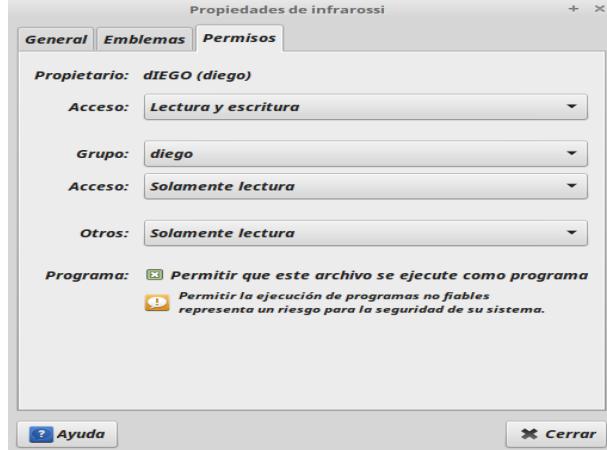


Figura 28. Instalación, propiedades FREEinfrarossi.

clave de administrador y se despliega la ventana del software FREEinfrarossi , como se aprecia en la figura 2 .

Carga de firmware en el modulo infraROSSI

La ventana del programa FREEinfrarossi , en la parte inferior derecha hay tres botones, oprimir el botón de firmware como se muestra en la figura 29 , y conectar la tarjeta micro controladora arduino uno al pc; oprimir el botón continuar y esperar que cargue el firmware en la tarjeta, una vez hecho esto retirar el micro controlador de la tarjeta y colocarlo en el montaje del vehículo.

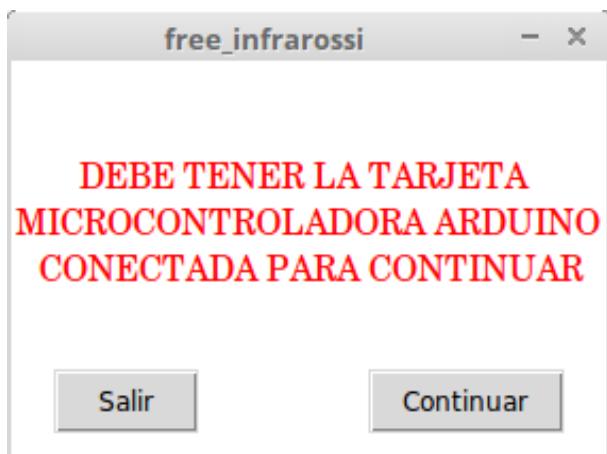


Figura 29. Ventana de instalación del firmware, que ofrece el software FREEinfrarossi , para el vehículo motorizado infraROSSI .

Diseño experimental del modulo infraROSSI

Disposición eléctrica

El modulo **infraROSSI** es un vehículo didáctico para la enseñanza experimental en física ver figuras 14 y 15 , los diferentes sistemas electrónicos del vehículo pueden removese en cuatro sistemas: control de hardware y comunicación bluetooth con ordenador ver figura 16, control rapidez del vehículo ver figura 17, sensor electro-recepción pasiva ver figura 18 , y un sensor de electro-recepción activa ver figura 19 .

Control de hardware. El modulo **infraROSSI** tiene un sistema de control de hardware ver figura 16 , que le indica al vehículo cuando debe avanzar, cuando debe detenerse, la rapidez con que debe avanzar, la distancia que debe avanzar, activar y detener el sensor de **electro-recepción pasiva** y enviar los datos vía bluetooth al ordenador, activar y detener el sensor de **electro-recepción activa** y enviar los datos vía bluetooth al ordenador, enviar señales del proceso que esta realizando con el encendido y apagado de diodos leds de colores y mantener una comunicación continua con el ordenador vía bluetooth.

El control de hardware ver figura 16 , está basado en un microcontrolador atmega 328P-PU según especificaciones del fabricante¹¹ , tiene 28 pines, con un voltaje de operación de 5 V, una corriente máxima de 40 mA por pin de salida, la suma de corriente en todas las salidas del microcontrolador no excede los 150 mA, trabaja en una frecuencia de 12 MHz. Este dispositivo consta de un modulador de frecuencia, un modulador de señal de 9V a 5V voltaje continuo, un modulo de comunicación tipo bluetooth, un botón de reinicio y un led indicador de procesos.

El modulador de frecuencia como se muestra en las figuras 30 y 31 , sincroniza el reloj interno del microcontrolador, calibrando los tiempos en los procesos del microcontrolador con los indicados por el fabricante y que reposan en la pagina oficial del proyecto arduino¹² .

El botón de reinicio o switch de reset como se muestra en las figuras 30 y 31, este interruptor reinicia el firmware cargado dentro del microcontrolador cuando surge algún conflicto interno en los procesos de este dispositivo.

El modulador de señal como se muestra en las figuras 32 y 33 , filtra la señal de entrada de 9 V voltaje continuo a la señal de trabajo óptima del microcontrolador y los diferentes sistemas electrónicos que es de 5 V voltaje continuo.

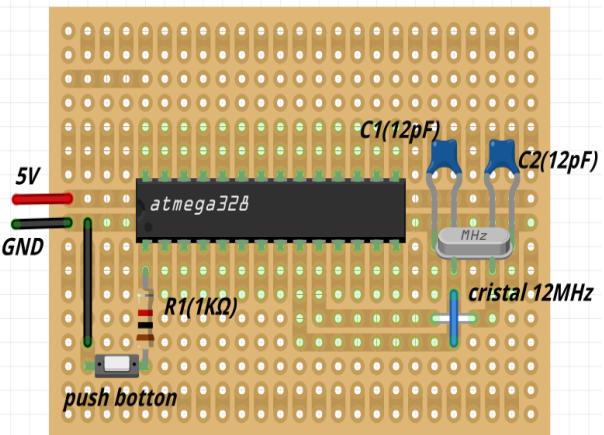


Figura 30. Esquema tarjeta perforada con control de hardware del vehículo **infraROSSI**, cristal oscilador 12 MHz, dos condensadores cerámicos de 12 pF, resistencia de 1kΩ y un botón; los cables de color negro son tierra, los de color rojo son voltaje, el cable azul y blanco son puentes. Figura generada en Fritzing.

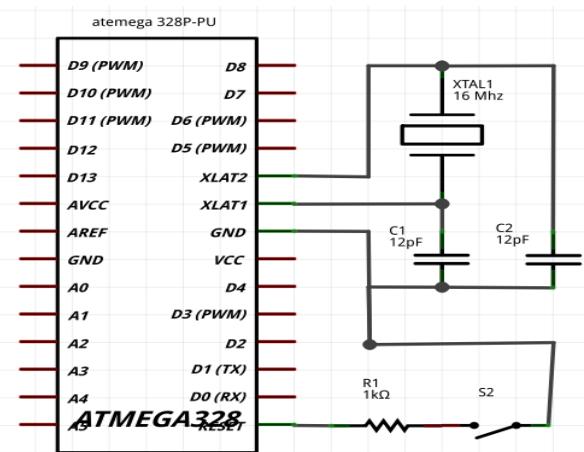


Figura 31. Esquema eléctrico microcontrolador **atmetga 328**, cristal oscilador 12 MHz, dos condensadores cerámicos de 12 pF, resistencia de 1kΩ y un botón. Figura generada en Fritzing.

La comunicación del microcontrolador con el ordenador como se muestra en las figuras 34 y 35 la ejecuta el modulo **bluetooth hc-05**; de esta manera el software de control **FREEinfraROSSI** tiene completo dominio del hardware del vehículo motorizado **infraROSSI**. El avance espacial del vehículo como el encendido y apagado de los diferentes sensores del dispositivo, junto con la

¹¹Enlace online a la hoja de datos del microcontrolador atmega328P-PU http://www.atmel.com/images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P_datasheet_Complete.pdf

¹²Página oficial proyecto arduino, [on line] <https://www.arduino.cc/>

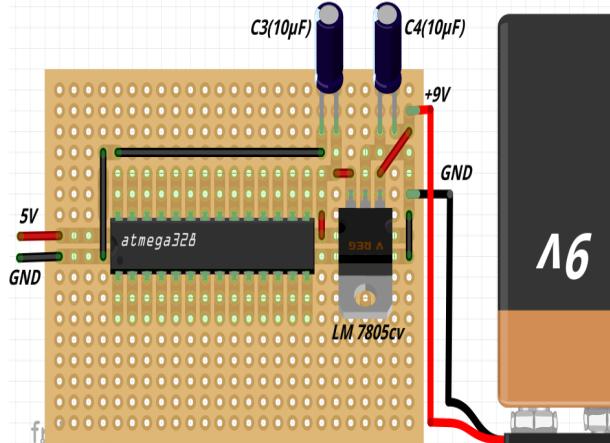


Figura 32. Esquema tarjeta perforada microcontrolador atmetga 328, transistor LM 7805 cv, dos condensadores electrolíticos de $10\mu F$, conectados a batería de 9 V. Figura generada en Fritzing.

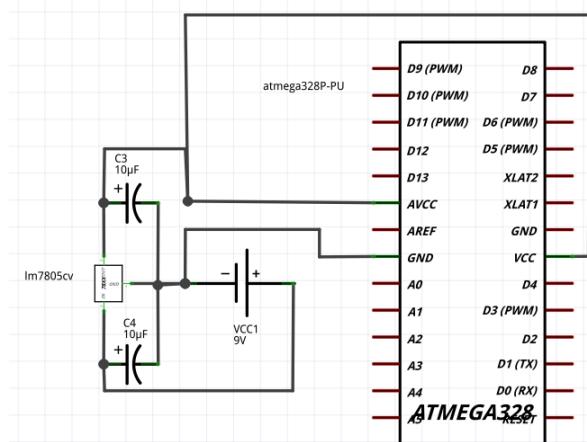


Figura 33. Esquema eléctrico microcontrolador atmetga 328, transistor LM 7805 cv, dos condensadores electrolíticos de $10\mu F$, conectados a batería de 9V. Figura generada en Fritzing.

recolección de datos de los sensores y la verificación del proceso que realiza la tarjeta, son enviados por el modulo hc-05 al ordenador, para su respectivo análisis; de igual forma las ordenes provenientes del software de control vía bluetooth son recibidas por el modulo hc-05 y enviadas al microcontrolador para su inmediata ejecución.

El diodo led que se observa en las figuras 34 y 35 es un indicador lumínico de los procesos que esta realizando el microcontrolador. De esta manera se tiene un indicador visual que muestra el proceso esta llevando a cabo el microcontrolador, si la intensidad de radiación que percibe el diodo receptor en cualquiera de los dos dispositivos eléctricos de electro-recepción es mayor o menor $10\mu W/\Omega$, realzando el carácter visual y estético del modulo motorizado infraROSSI.

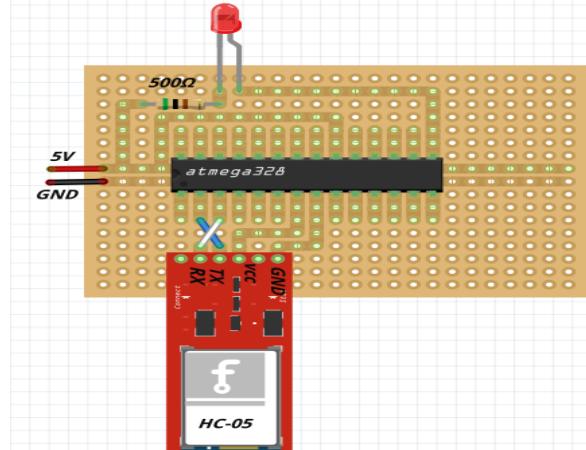


Figura 34. Esquema tarjeta perforada microcontrolador atmetga 328P-PU, modulo bluetooth hc-05, diodo led de 3mm y una resistencia de 500Ω . Figura generada en Fritzing.

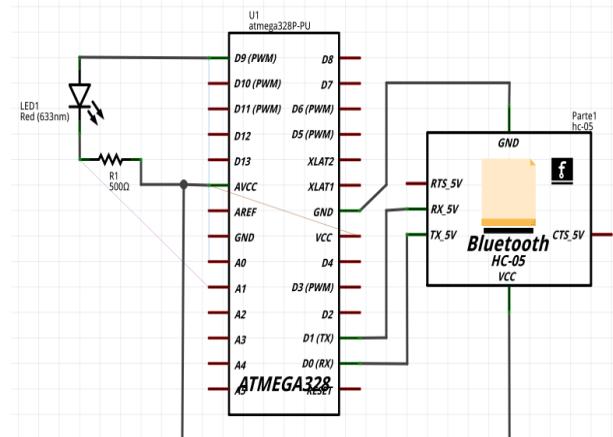


Figura 35. Esquema eléctrico microcontrolador atmetga 328P-PU, modulo bluetooth hc-05, diodo led de 3mm y una resistencia de 500Ω . Figura generada en Fritzing.

El control de hardware que se observa en las figuras 36, 37 y 38, consta de un modulador de señal LM 7805cv, modulador de frecuencia con oscilador 12 Mhz, el pin IN2 es la conexión al led de color 3mm, el pin IN3 es la conexión al led de color 3mm, pin de salida a +5v, pin de salida a GND, modulo bluetooth hc-05, botón de reinicio, microcontrolador atmega 328P-PU, el pin OUT1 es la salida al sensor de electro-recepción activa, el pin OUT2 es la salida al sensor de electro-recepción pasiva, el pin OUT3 es el pin de salida al sistema de control de velocidad, el pin IN1 es la salida al diodo emisor infrarrojo; la placa perforada tiene dimensiones de $6x4,5cm^2$, con un acabado en velcro, lo que facilita el montaje de la misma en el vehículo.

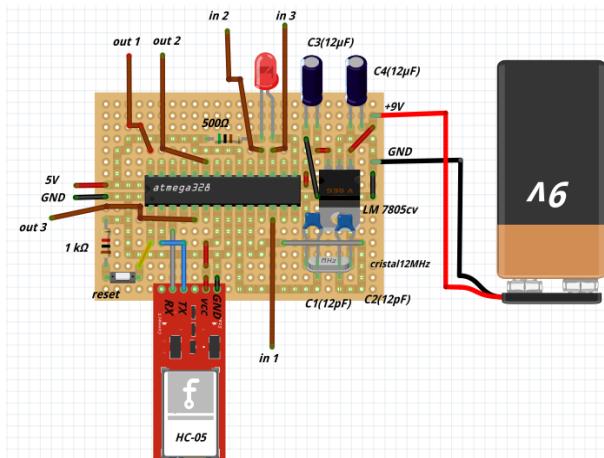


Figura 36. Esquema en tarjeta perforada del microcontrolador atmega 328P-PU, con las disposiciones del sistema de control de hardware. Figura generada en Fritzing.

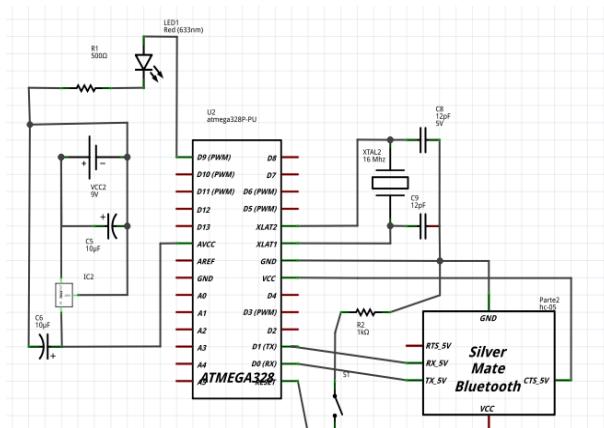


Figura 37. Esquema eléctrico microcontrolador atmega 328P-PU, con las disposiciones del sistema de control de hardware. Figura generada en Fritzing.

Cables conectores tipo hembra para que el usuario final NO manipule el microcontrolador atmega328 P-PU o podrían dañarlo; su parte inferior esta forrada en velcro lo que facilita remover el sistema de control en cualquier momento bien sea para mostrar sus dimensiones o explicar el funcionamiento eléctrico del sistema y las operaciones que realiza el microcontrolador atmega 328P-PU.

Al semiconductor se le coloco un disipador térmico, para evitar el sobrecalentamiento del semiconductor y de esta manera garantizar el correcto funcionamiento de este dispositivo el cual alimenta al sistema de control de hardware y a los demás dispositivos eléctricos del modulo motorizado infraROSSI con un voltaje de 5V voltaje

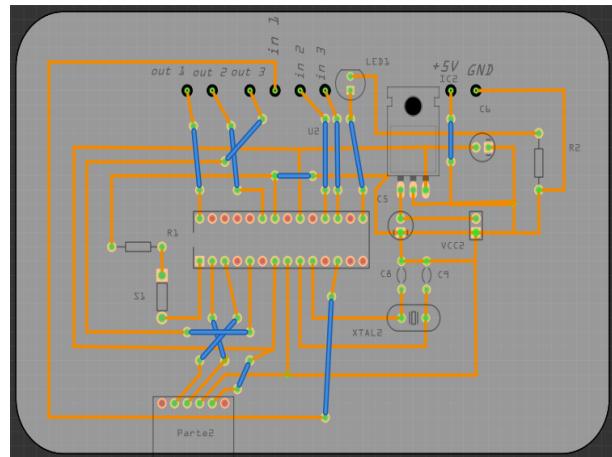


Figura 38. Esquema de tarjeta impresa para el microcontrolador atmega 328P-PU, con las disposiciones del sistema de control de hardware. Figura generada en Fritzing.

continuo.

Control de avance. El sistema de control de avance ver figura 17, es el que da el movimiento del vehículo únicamente hacia adelante, pues no consta de un sistema de reversa, esta peculiaridad del vehículo permite que este avance una cantidad mínima de distancia para la recolección de datos de los distintos sensores de electro-recepción que dispone el vehículo.

El sistema de control en el avance del vehículo infraROSSI como se observa en las figuras 39 , 40 y 41 , consta de dos salidas a GND, dos salidas a +5V, una entrada a GND, una entrada a +5V, un tip 122, una resistencia de $5k\Omega$, diodo 1N4001, el pin OUT3 es la salida al pin D3(PWM) del microcontrolador atmega 328P-PU, salida M1 para el motor, salida M2 para el motor; un motor de corriente directa de medio vatio de potencia con sistema de transmisión de engranaje de eje fijo como se observa en la figura 13 .

El pin D3(PWM) del microcontrolador envía una señal de 24mA al pin base del tip 122¹³ , durante 37 ms , dando un avance de 2 mm al vehículo motorizado infraROSSI. El sistema de control de avance del vehículo motorizado infraROSSI tiene dimensiones de $3x4 cm^2$, forrado en velcro, lo que facilita retirar el sistema de control para mostrar sus conexiones o explicar el funcionamiento del semiconductor TIP 122 para el control de velocidad en el avance espacial del vehículo.

El paso en el avance del vehículo motorizado infraROSSI

¹³Semiconductor, F. (2008). tip122 npn. [On line]. Disponible en: <http://pdf.datasheetcatalog.net/datasheet/fairchild/TIP122.pdf>.

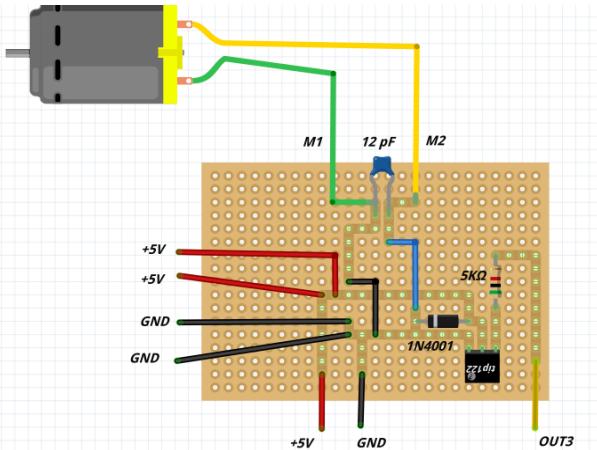


Figura 39. Esquema tarjeta perforada del sistema de control de avance. Figura generada en Fritzing.

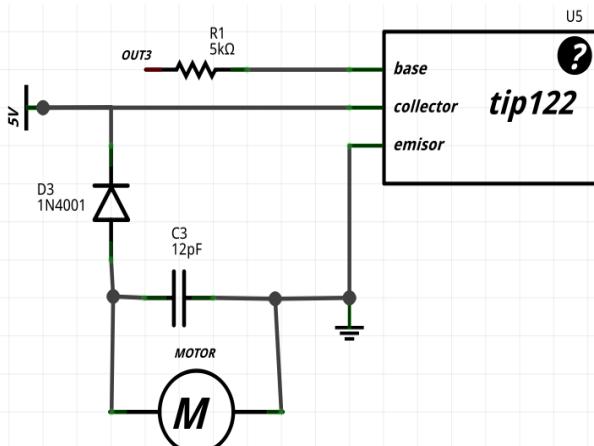


Figura 40. Esquema del circuito eléctrico del sistema de control de avance. Figura generada en Fritzing.

se calibra colocando el control de hardware que se muestra en la figura 16 y el control de avance que se muestra en la figura 17 en las bases del vehículo motorizado infraROSSI que se muestra en las figuras 11 y 46.

El vehículo motorizado infraROSSI cuenta con cinco modos de avance, en cada modo aumenta el avance en 0.4 mm, estos se encuentran reservados para el programador del software de control FREEinfraROSSI, dejando solo el avance de 2 mm solamente hacia adelante fijo y no cuenta con sistema de avance en reversa.

El vehículo motorizado infraROSSI se coloca en una superficie de apoyo, con la ayuda de un metro para medir la distancia de avance, un bolígrafo para marcar el avance y el software de control FREEinfraROSSI, se hace avanzar 94 veces y se mide la distancia de avance, se realizarán 8 pruebas de ajuste en el avance y los datos obtenidos se

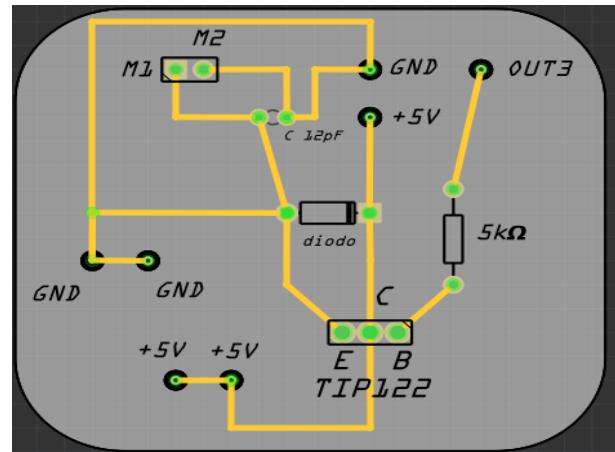


Figura 41. Tarjeta de circuito impreso del sistema de control de avance. Figura generada en Fritzing.

Tabla 5

Análisis de los datos obtenidos en la calibración del sistema de control de avance.

Prueba	Distancia [cm]	paso [cm]	error
1.	20.2	0.214	$\pm 1\text{mm}$
2.	20	0.212	$\pm 1\text{mm}$
3.	20.1	0.213	$\pm 1\text{mm}$
4.	19.8	0.210	$\pm 1\text{mm}$
5.	19.5	0.207	$\pm 1\text{mm}$
6.	20.4	0.217	$\pm 1\text{mm}$
7.	20	0.212	$\pm 1\text{mm}$
8.	20	0.212	$\pm 1\text{mm}$
DISTANCIA PROMEDIO		20 cm	
PASO PROMEDIO		0.212 cm	

Nota. Datos del modulo infraSOSSI con sistema de avance; la distancia de avance esta en cm, el avance es el cociente de la distancia avanzada y la cantidad de pasos, el error de la medida es $\pm 1\text{mm}$.

muestran en la tabla 5.

Sensor de electro-recepción pasiva. El sistema de electro-recepción pasiva como se observa en la figura 18, es un sistema de adquisición de información energética de radiación electromagnética en el espectro infrarrojo de manera pasiva, solo recibe señales electromagnéticas en estas longitudes de onda; la información que recibe el sensor es enviada al microcontrolador y este la envía al ordenador a través del dispositivo bluetooth para el análisis de estos datos por la interfaz gráfica FREEinfraROSSI.

El sistema de electro-recepción pasiva como se observa en las figuras 42 y 43, consta de un diodo receptor infrarrojo o IRR, dos led de 3mm, dos resistencias de 500Ω, resistencia de 5kΩ, el pin IN2 es salida al pin D10(PWM)

del microcontrolador, el pin IN3 es salida al pin D11(PWM) del microcontrolador, el pin OUT2 es salida al pin (A0) del microcontrolador, pin de entrada +5V y pin de entrada a GND.

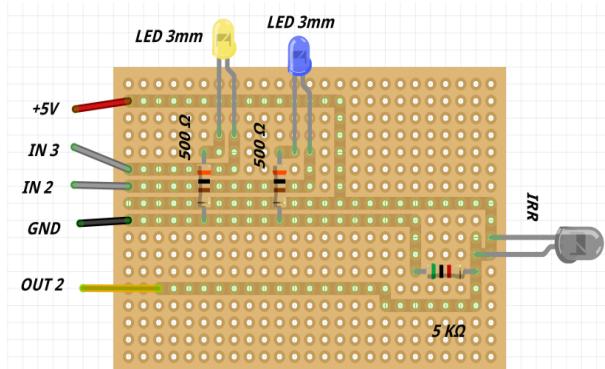


Figura 42. Tarjeta perforada con el montaje del sensor de electro-recepción pasiva. Figura generada en Fritzing.

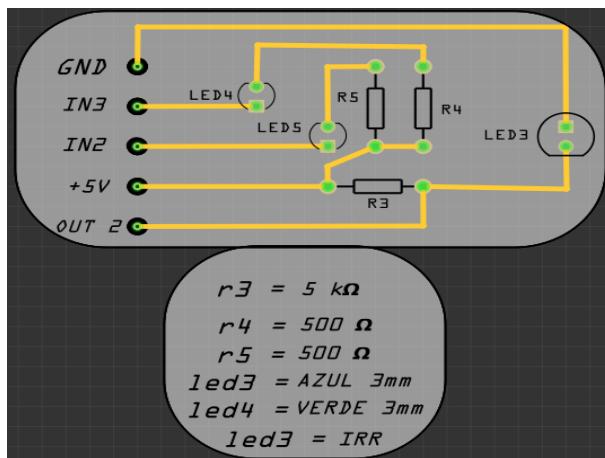


Figura 43. Placa de circuito impreso para el sensor de electro-recepción pasiva. Figura generada en Fritzing.

El sensor esta calibrado como se muestra en la gráfica 24, para medir $4\mu W/mV$, recolectando 70 datos por segundo, los diodos led son solo indicadores del proceso que esta realizando el microcontrolador y de la intensidad de señal recibida por el sensor, cantidad máxima medible $20mW/\Omega$.

Sensor de electro-recepción activa. El sensor de electro-recepción activa como se observa en la figura 19 , esta basado en el principio de electro-recepción activa sin barrido, el cual consiste en un sensor de recepción por reflexión, donde una fuente fotónica ubicada al lado del sensor envía un flujo de energía electromagnética en la longitud de onda de 940 nm por el espacio y al encontrar una barrera que le impide el paso, esta interactúa con el obstáculo dando una reflexión del flujo irradiado en sentido

contrario, fruto de esta interacción es la reducción de la intensidad de la señal.

El sensor infrarrojo percibe esta señal transformando la energía lumínica en energía eléctrica, esta información es enviada al microcontrolador quien a su vez la envía por el bluetooth al software de control FREEinfraROSSI para su análisis.

El sistema de electro-recepción activa como se observa en las figuras 44 y 45 , consta de un diodo receptor infrarrojo o IRR, un diodo emisor infrarrojo o IRE, una resistencia de 300Ω, una resistencia de 5kΩ, un pin de entrada +5V y un pin de entrada a GND , el pin IN1 es la conexión al pin D6(PWM) del microcontrolador, el pin OUT1 es la salida al pin A5 del microcontrolador. El emisor infrarrojo de longitud de onda 940 nm tiene una densidad de flujo radiante sobre ángulo sólido de $20mW/\Omega$.

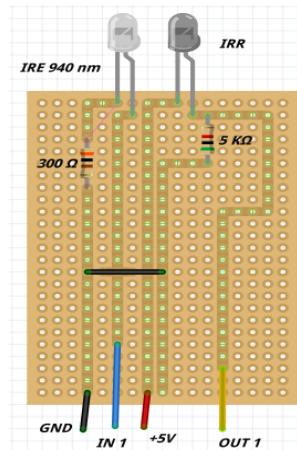


Figura 44. Esquema tarjeta perforada del radar de electro-recepción activa sin barrido. Figura generada en Fritzing.

Para el esquema de tarjeta perforada como se muestra en la figura 44 del sensor de electro-recepción activa, el cable de color negro es el pin de salida a tierra (GND), el cable de color azul es el pin de salida al pin D6(PWM) del microcontrolador, el cable de color rojo es el pin de conexión a (+5V) , el cable de color naranja es la conexión al pin A5 del microcontrolador. El sensor de electro-recepción activa como se muestra en la gráfica 24 esta calibrado para medir $4\mu W/mV$, recolectando 70 datos por segundo, máxima intensidad medible $20mW/\Omega$, con máximo alcance en la distancia de detección de 60 centímetros a la fuente radiante.

El sensor de electro-recepción activa envía la información de que cantidad de radiación infrarroja esta incidiendo sobre el, al microcontrolador atmega 328P-PU quien a su vez le comunica al modulo bluetooth hc-05 que

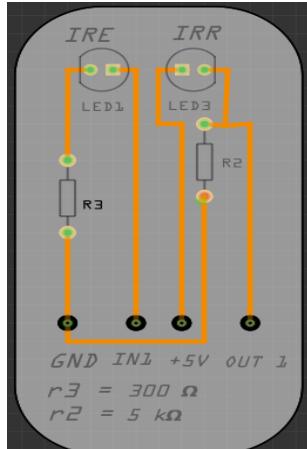


Figura 45. Placa de circuito impreso para el sensor de electro-recepción activa. Figura generada en Fritzing.

inicie la transmisión de datos al ordenador para su análisis con el software *FREEinfraROSSI*.

Disposición mecánica

La disposición mecánica del vehículo motorizado free infrarossi como se observa en las figuras 10, 11, 12, 13 y 46, esta compuesta por un chasis de madera, dos pares de llantas, dispone de tres bases macizas forradas en *velcro*, para facilitar el montaje de los diferentes sensores y sistemas de control, la base del sistema de control es hueca con espacio suficiente para la batería de 9 V de tipo cuadrada recargable, el montaje del vehículo, un capó de material fommy negro para aislar el ruido en el sensor de electro-recepción activa, dos ejes fijos, un sistema de transmisión de tipo engranaje y sistema de avance de 2 mm (únicamente hacia adelante), los sistemas de control se puedan quitar y poner en el vehículo debido al material de *velcro*, el sistema de control aparte de ser móvil, puede ser removido el microcontrolador y programado con futuras actualizaciones del firmware que provee el software *FREEinfraROSSI* para el vehículo motorizado *infraROSSI* y ampliar su funcionalidad como instrumento de laboratorio, estos sensores de adquisición de datos no solo sirven para ilustrar la propiedad de difracción, atenuación y absorción de las ondas electromagnéticas, sino que por su naturaleza activa y pasiva, podrían ser utilizados para ilustrar la magnitud de la aceleración gravitacional terrestre, las características en el movimiento de un péndulo, entre otras; como se piensa sacar en la versión 1.1 del software *FREEinfraROSSI*.

Puede revisar la pagina donde se aloja el proyecto *FREEinfraROSSI*¹⁴, como se observa en los proyectos de la pagina esta el proyecto *FREE_PoPs_1.0*,¹⁵ el cual es un programa que ilustra el periodo de oscilación de un

péndulo simple, que ha su vez es el prototipo de un modulo que va hacer adherido al software *FREEinfraROSSI* en su versión 1.1 .

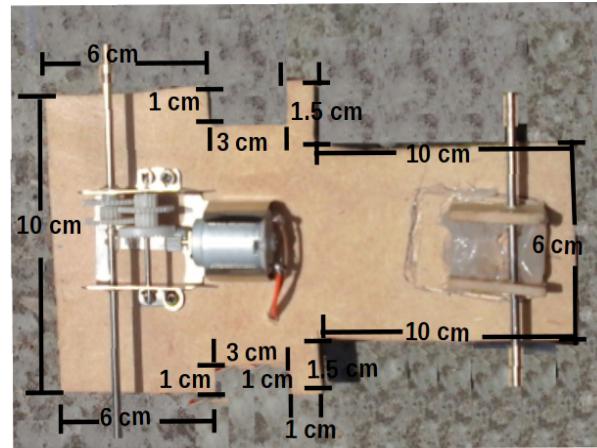


Figura 46. Chasis de madera con dimensiones en centímetros, la caja del eje delantero tiene dimensiones 2 centímetros de ancho, 3 centímetros de largo y 2.5 centímetros de alto; el eje delantero debe estar a 1 centímetro de separación del chasis.

Test del modulo *infraROSSI* y su software *FREEinfraROSSI*

En esta sección se realiza una verificación y testeo del software de control *FREEinfraROSSI* y el vehículo motorizado *infraROSSI*; detallando las conexiones electrónicas del modulo *infraROSSI* para los diferentes laboratorios que ofrece el software de control *FREEinfraROSSI*.

Difracción. Colocar el sensor de electro-recepción pasiva junto con el control de avance y el control de hardware en el vehículo motorizado *infraROSSI* como se observa en la figura 47.

Los esquemas de pines de las partes utilizadas para el montaje se muestran en las figuras 48 , 49 y 50 ; se debe conectar el pin ch2 con el pin ep5, el pin ch3 con el pin ca9, el pin de ch5 con el pin ep2, el pin de ch6 con el pin ep3, el pin ch7 con el pin ca2, el pin ch8 con el pin ca1, el pin ca4 con el pin ep4, el pin ca3 con el pin ep1, por ultimo el pin ca7 y el pin ca8 son las conexiones al motor.

Colocar la fuente¹⁶ emisora de fotones infrarrojos con

¹⁴Para mayor información [online] <https://github.com/Diego-debian/Free-infrarossi>

¹⁵Para mayor información [on line] https://github.com/Diego-debian/FREE_PoPs_1.0

¹⁶Esta fuente se elaboro con un diodo led infrarrojo de 850 nm y un encapsulado epoxi de 3 mm.

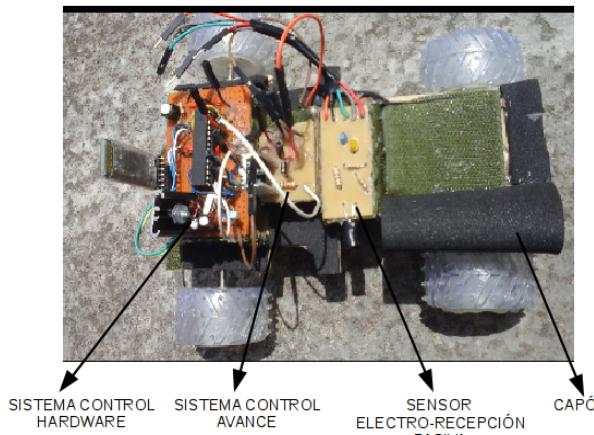


Figura 47. Montaje modulo motorizado *infraROSSI* para el laboratorio de difracción.

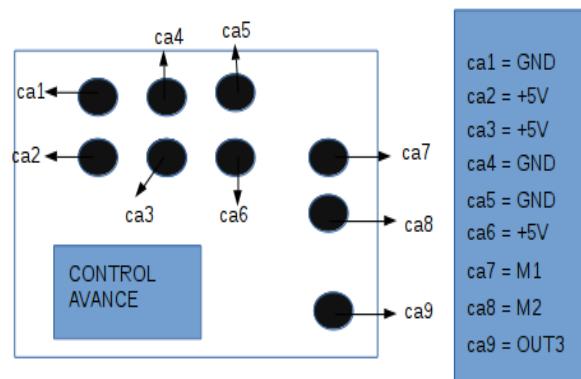


Figura 49. Esquema de pines del sistema de control de avance.

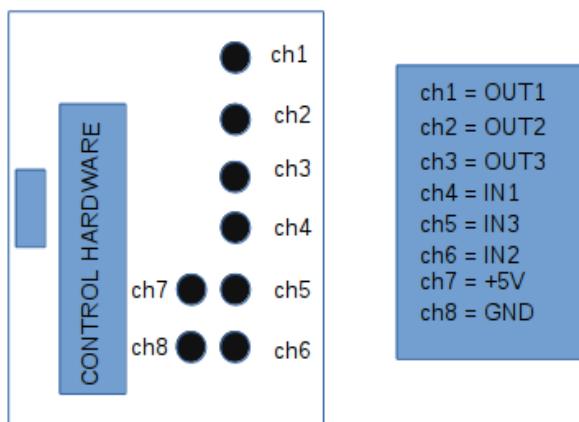


Figura 48. Esquema de pines del sistema de control de hardware.



Figura 50. Esquema de pines del sensor de electro-recepción pasiva en infrarrojo.

la red de difracción de 100 líneas por milímetro, frente a ella colocar el modulo motorizado *infraROSSI* a 45 cm de la fuente emisora, como se muestra en la figura 1.

Abrir una terminal de GNU-Linux y escribir *infrarossi*, oprimir enter y la clave de superusuario, luego de abrir el programa debe oprimir el botónon, esperar que se empareje el bluetooth, una vez emparejado el bluetooth el programa desplegará un tercer menú ver figura 2, oprimir el botón de difracción y esperar que el programa tome los datos necesarios.

Luego de capturar los datos aparecerá la gráfica de los datos, oprima doble click izquierdo en el máximo de interferencia y sin soltar el cursor lleve la linea al siguiente máximo de interferencia, suelte el botón del cursor e inmediatamente aparecerá la gráfica con el análisis de longitud de onda infrarroja del diodo, tal como se muestra en la figura 3.

Ahora el programa *FREEinfraROSSY*, llama al script *Estadis2.py* que se encuentra en el anexo E20 pagina 132 ; el cual realiza un análisis estadísticos de los datos midiendo la distancia que hay entre cada patrón de interferencia constructiva como se observa en la gráfica 3 ; y, predice el error porcentual en la aproximación realizando la diferencia del dato suministrado por el fabricante¹⁷ del diodo led emisor infrarrojo y la encontrada en el experimento de difracción con una rejilla de 100 líneas/mm.

Este experimento se realizó 9 veces, obteniendo los resultados que se muestran en la tabla 1. La longitud

¹⁷Enlace a la hoja de datos del diodo led emisor infrarrojo de 850 nm suministrada por el fabricante online http://www.betlux.com/product/LED_lamp/infrared%20diodo/BL-L314IR.PDF

de onda promedio de los datos en la tabla 1 es 851,77nm dando un error porcentual de 0,209 % de la longitud de onda suministrada por el fabricante.

Atenuación. Colocar el sensor de electro-recepción activa junto con el control de avance y el control de hardware en el vehículo motorizado *infraROSSI* como se observa en la figura 51.

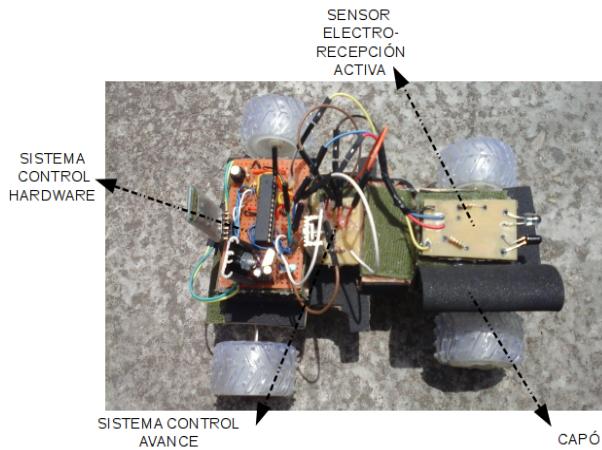


Figura 51. Montaje modulo motorizado *infraROSSI* para el laboratorio de atenuación y absorción.

Los esquemas de pines de las partes utilizadas para el montaje se muestran en las figuras 48 , 49 y 52 ; se debe conectar el pin ch1 con el pin ea4, el pin ch3 con el pin ca9, el pin de ch4 con el pin ea2, el pin ch7 con el pin ca2, el pin ch8 con el pin ca1, el pin ca5 con el pin ea1, el pin ca6 con el pin ea3, por ultimo el pin ca7 y el pin ca8 son las conexiones al motor.

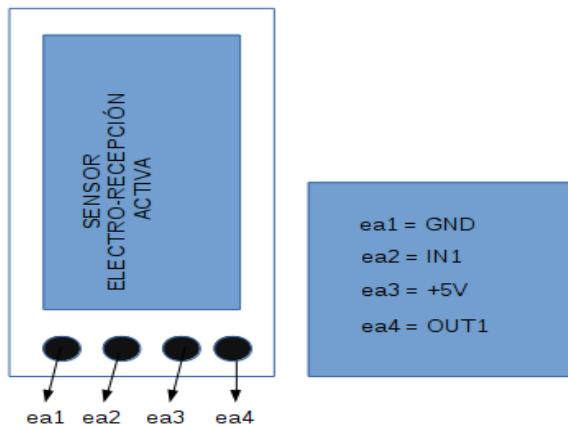


Figura 52. Esquema de pines del sensor de electro-recepción activa en infrarrojo.

Situar el vehículo motorizado *infraROSSI* a una distancia

de 30 cm del espejo como se muestra en la figura 4; el espejo debe estar perpendicular a la parte frontal del vehículo. Abrir una terminal de GNU-Linux y escribir *infrarossi*, oprimir enter y la clave de superusuario, luego de abrir el programa como se muestra en la figura 3, debe oprimir el botón on, esperar que se empareje el bluetooth, una vez emparejado el modulo bluetooth, el programa desplegará un tercer menú, oprimir el botón de atenuación y esperar que el programa tome los datos necesarios.

Ahora el software de control *FREEinfraROSSI* le envía vía bluetooth la señal de avanzar y capturar datos al vehículo motorizado *infraROSSI* el cual avanza 2 milímetros por cada paso, recolecta 140 datos por cada avance enviándolos vía bluetooth al ordenador en donde el software de control realiza un análisis estadístico de los mismos, cuando termina este análisis envía una señal al modulo motorizado *infraROSSI* vía bluetooth, indicándole que avance nuevamente y repita el proceso, esto lo realiza 117 veces hasta avanzar una distancia de 30 centímetros; una vez terminado de recoger todos los datos realiza una estadística sobre toda la muestra de estos datos capturados.

El programa *FREEinfraROSSI* después de recoger los datos realiza un análisis estadístico de los mismos, según (Ostle, 1981) “un ajuste lineal de una función exponencial de la forma $Y^* = aX^b$, siendo Y la intensidad irradiada, a la amplitud, X la distancia de avance, b el valor del exponente de atenuación, Y^* es el valor esperado de la variable física; aplicando logaritmo natural para linealizar se obtiene:

$$\ln(Y^*) = \ln(a) + b\ln(X) \dots \Rightarrow V^* = A + bU, \quad (31)$$

donde V^* es $\ln(Y^*)$, A es el $\ln(a)$ y U es igual al $\ln(X)$.

La suma de todos los errores debe ser diferentes a cero $\sum e \neq 0$.

El exponente de ajuste b se halla con la varianza de U sobre V dividida entre la varianza de U sobre U , obteniendo

$$b = \frac{S_{UV}}{S_U^2} = \frac{\frac{1}{n} \sum_{i=1}^n UV - \bar{U}\bar{V}}{\frac{1}{n} \sum_{i=1}^n U^2 - \bar{U}^2}, \quad (32)$$

ahora el valor de A sera igual al valor medio del logaritmo de la variable física \bar{V} menos el valor de la multiplicación entre el exponente b y el valor medio del logaritmo de la variable experimental:

$$A = \bar{V} - b\bar{U}, \quad (33)$$

deshaciendo el logaritmo de A se obtiene el valor de la amplitud a:

$$a = \text{antiln}(A) = \text{antiln}(\bar{V} - b\bar{U}), \quad (34)$$

de modo que el ajuste efectuado es:

$$Y^* = aX^b = [\text{antiln}(\bar{V} - b\bar{U})] * X^{\left(\frac{\frac{1}{n} \sum_{i=1}^n UV - \bar{U}\bar{V}}{\frac{1}{n} \sum_{i=1}^n U^2 - \bar{U}^2}\right)}, \quad (35)$$

la bondad del ajuste es el error cuadrático medio o *ECM* y es igual a:

$$ECM = \frac{\sum_{i=1}^n e_i^2}{n}, \quad (36)$$

siendo e_i cada una de las diferencias entre las variables físicas y los valores estimados para las variables físicas $e_i = Y_i - Y_i^*$; al haber transformado la variable física ya no se minimiza $\sum e^2$ sino $\sum (\ln(Y) - \ln(Y^*))^2$, de ahí que $\sum e \neq 0$.

Luego de este análisis estadístico, aparece la gráfica de estos, la cual se aprecia en la figura 5, esta ya contiene el análisis estadístico y arroja el valor del exponente que debe tener la distancia; los datos del experimento de atenuación están en puntos azules, la gráfica estimada estadísticamente está en líneas azules, la gráfica teórica son las líneas de color rojo.

La gráfica de análisis y los datos capturados se almacenan dentro del archivo llamado Carpetas/Atenuacion con la fecha y hora del análisis de datos.

De los datos que se muestran en la tabla 2 se obtiene la amplitud de la función densidad promedio de flujo fotónico reflejada sobre el espejo y que incide sobre el diodo led infrarrojo la cual tiene un valor de 0,000259 [W][m²], por lo que se considera este valor como la amplitud ideal para la función intensidad, también se obtiene el valor del exponente que acompaña a la distancia debido al decaimiento de esta densidad de energía sobre el espacio el cual tiene un valor promedio de -1,906 que esta muy cercano al predicho por la teoría el cual es -2, con un error en su estimación de 4,7 %, lo cual indica que el decaimiento en la densidad de flujo electromagnético que atraviesa una superficie en la unidad de tiempo, que a su vez se refleja y por ende que llega al sistema detector del modulo motorizado *infraROSSI*, decrece con el cuadrado de la distancia.

El promedio del error cuadrático medio en los datos obtenidos por el experimentador en la tabla 2, indican que 11,286x10⁻⁵ % de estos no pueden ser explicados por este modelo físico-matemático, dejando solamente un 99,9998 % de fiabilidad en el ajuste estadístico, utilizado por el software FREE*infraROSSI*.

Absorción. Colocar el sensor de electro-recepción activa junto con el control de avance y el control de hardware en el vehículo motorizado *infraROSSI* como se observa en la figura 51. Los esquemas de pines de

las partes utilizadas para el montaje se muestran en las figuras 48, 49 y 52; se debe conectar el pin ch1 con el pin ea4, el pin ch3 con el pin ca9, el pin de ch4 con el pin ea2, el pin ch7 con el pin ca2, el pin ch8 con el pin ca1, el pin ca5 con el pin ea1, el pin ca6 con el pin ea3, por ultimo el pin ca7 y el pin ca8 son las conexiones al motor.

Situar el vehículo motorizado *infraROSSI* a una distancia de 30 cm de la muestra (octavo de cartulina) tal como se muestra en la figura 6, la muestra debe estar perpendicular a la parte frontal del vehículo. Abrir una terminal de GNU-Linux y escribir *infrarossi*, oprimir enter y la clave de superusuario, luego de abrir el programa debe oprimir el botón, esperar que se empareje el bluetooth, una vez emparejado el modulo bluetooth, el programa desplegará un tercer menú como se observa en la figura 2, oprimir el botón de absorción y esperar que el programa tome los datos necesarios.

Ahora el software de control FREE*infraROSSI* le envía vía bluetooth la señal de avanzar y capturar datos al vehículo motorizado *infraROSSI* el cual avanza 2 milímetros por cada paso, recolecta 140 datos por cada avance enviándolos vía bluetooth al ordenador en donde el software de control realiza un análisis estadístico de los mismos, cuando termina este análisis envía una señal al modulo motorizado *infraROSSI* vía bluetooth, indicándole que avance nuevamente y repita el proceso, esto lo realiza 117 veces hasta avanzar una distancia de 30 centímetros; una vez terminado de recoger todos los datos realiza una estadística sobre toda la muestra de estos datos capturados.

El programa FREE*infraROSSI* después de recoger los datos realiza un análisis estadístico de los mismos, según (Ostle, 1981) utilizando las ecuaciones 34, 35 y 36.

Este análisis se realizó previamente con un espejo, el cual deja como patrón de referencia o de intensidad ideal, el valor medio de 0,000259X⁻²μW/m² que se muestra en la figura 5; el programa calcula un valor estimado para los datos de la siguiente manera: la intensidad transmitida I_t debe ser igual a la intensidad ideal I_{ideal} menos la intensidad que mide el sensor I_{real} .

$$I_t = I_{ideal} - I_{real}$$

El factor de perdida en la intensidad T debe ser igual al cociente de la intensidad I_t y la intensidad ideal I_{ideal}

$$T = \frac{I_t}{I_{ideal}}$$

por lo tanto el promedio en el factor de perdida debe ser igual a la transmitancia o mejor dicho a la energía que se transmitió a la muestra,

$$\bar{T} = \sum T_i/N$$

como la reflectancia es igual a $R = 1 - T$ (ecuación 22), el valor de la amplitud a , debe ser igual:

$$a = (1 - \bar{T})(0,000259)$$

por lo que ahora la intensidad estimada en la figura 5 con la etiqueta *Exp*, en función del inverso de la distancia, disminuye con la reflectancia R :

$$I^* = (1 - \bar{T})(0,000259)X^{-2}$$

después de este análisis el software FREEinfraROSSI realiza la gráfica mostrando en ella (figura 7) la intensidad ideal $I_{ideal}(x)$, la intensidad estimada debido a la reflectancia I^* y el factor de transmitancia o factor de caída en la intensidad F_p .

En la tabla 3 se muestra la función intensidad a en $[W][m^2]$, estimado por el software FREEinfraROSSI, para cada una de las diferentes muestras del mismo material de cartulina pero diferentes pigmentos; la cartulina en la cual hubo más transmitancia fue la de color negro, pues el 86.84% de intensidad de la luz incidente sobre ella fue transmitida al material, llegando solo un 13.16% de esta al sensor; ahora la muestra que mayor reflectancia tuvo fue la de color rojo pues el 18.08% de la intensidad lumínica llegó al sensor, indicando que solo el 81.92% de esta fue transmitida al material de muestra.

CONCLUSIONES

- * Se detalla el diseño y montaje de un vehículo motorizado con transmisión electromagnética, sensores de electro-recepción activa-pasiva, con comunicación bluetooth desde un ordenador equipado con un sistema operativo GNU-Linux y el modulo motorizado con hardware libre, para el estudio e ilustración de las propiedades de difracción, atenuación, absorción y fenómenos ondulatorios de las ondas electromagnéticas en el espectro infrarrojo de dicha radiación.
- * El software de este proyecto obtuvo la licencia GPL3 otorgada por la FREE SOFTWARE FOUNDATION para el desarrollo de software libre; por lo que cualquier persona puede tener acceso al software de control FREEinfraROSSI, y empezar a utilizarlo sin ninguna restricción o modificar los scripts citando siempre al autor original de los mismos.
- * Se detalla el uso del software de control FREEinfraROSSI, diseñado con herramientas libres del proyecto GNU-Linux¹⁸ a través de las distribuciones basadas en Debian¹⁹ como lo es UBUNTU 14.04LTS²⁰ y Linux MINT 17.2²¹ entre otras; para la manipulación del vehículo motorizado y sus diferentes sensores de electro-recepción,

con alojamiento en la red a través del enlace [en linea] <https://github.com/Diego-debian/Free-infrarossi>, el cual brinda al usuario del software el soporte técnico a través de actualizaciones del mismo como se piensa para la versión 1.1 del software FREEinfraROSSI donde se incluirá nuevos módulos y laboratorios para facilitar la enseñanza en física, en el aula de clase a través de las nuevas tecnologías de la información para la enseñanza en ciencias, tambien para ampliar el rango de cubrimiento del programa FREEinfraROSSI a distribuciones GNU-Linux brindando soporte a los sistemas operativos basados en ARCH Linux²² y RED HAT como lo es Fedora y CentOS; para mayor información, ver nuevas actualizaciones y proyectos futuros [online] <https://github.com/Diego-debian>.

* El vehículo motorizado infraROSSI y su software de control FREEinfraROSSI ilustran de manera cuantitativa y cualitativa fenómenos ondulatorios como la difracción e interferencia de las ondas electromagnéticas, calculando de manera aproximada su longitud de onda λ , con error inferior al 4%, como se muestra en la gráfica 3. La longitud de onda media del experimento que se muestra en la tabla 1, es de $\lambda_{exp} = 851.77 \text{ nm}$, la cual difiere en 1.77 nm del dato de la longitud de onda del diodo led emisor infrarrojo de 850 nm según especificaciones del fabricante²³.

* El vehículo motorizado infraROSSI y su software de control FREEinfraROSSI ilustran de manera cuantitativa y cualitativa fenómenos ondulatorios y corpusculares de la radiación electromagnética como la atenuación con el inverso del cuadrado de la distancia, la radiansa, la irradiansa, fenómenos de transporte e inyección y su análisis estadístico, calculando de una manera aproximada el exponente que acompaña a la atenuación debido a la distancia de propagación del flujo de energía radiante producida en el diodo emisor infrarrojo como se muestra en la gráfica 5; la amplitud del valor medio de la

¹⁸Enlace al proyecto GNU-Linux [online] <https://www.gnu.org/gnu/linux-and-gnu/>

¹⁹Debian <https://www.debian.org/>

²⁰UBUNTU14.04LTS [online] <http://www.ubuntu.com/>

²¹Linux MINT 17.2 [online] <https://linuxmint.com/>

²²Enlace [online] <https://www.archlinux.org/>

²³Hoja de datos del led emisor infrarrojo (850nm) suministrada por el fabricante [online] http://www.betlux.com/product/LED_lamp/infrared%20diodo/BL-L314IR.PDF

CONCLUSIONES

función intensidad como se muestra en la tabla 2 es **0.000259 [W][m²]**, por lo que se considera este valor como la amplitud ideal para la función intensidad; el valor medio en el factor de atenuación de la distancia es **-1.906** que esta muy cercano al predicho por la teoría el cual es **-2**, con un error en su estimación de **4.7%**, lo cual indica que el decaimiento en la densidad de flujo electromagnético que atraviesa una superficie en la unidad de tiempo, que a su vez se refleja y por ende que llega al sistema detector del modulo motorizado **infraROSSI**, decrece con el cuadrado de la distancia.

- * El vehículo motorizado **infraROSSI** y su software de control ilustran de manera cuantitativa y cualitativa fenómenos ondulatorios y corpusculares de la radiación electromagnética como la atenuación con el inverso del cuadrado de la distancia, la radianza, la irradiancia, fenómenos de transporte e inyección, calculando el factor de perdida de intensidad en el sensor de **electro-recepción** activa del vehículo motorizado **infraROSSI** y de esta manera predice los valores de reflectancia y transmitancia del material estudiado como se muestra en las gráficas 8 9 y 7. Según la tabla 3 la muestra en la cual hubo mayor transmitancia, es la cartulina de color negro, pues el **86.84%** de intensidad de la luz incidente sobre ella fue transmitida al material, llegando solo un **13.16%** de esta radiación al sensor de **electro-recepción** activa del modulo **infraROSSI**; según la tabla 3 la muestra que mayor reflectancia tuvo es la de color rojo pues el **18.08%** de la intensidad lumínica llegó al sensor, indicando que solo el **81.92%** de esta radiación fue transmitida al material de muestra; el error cuadrático medio indica que solo el **0.000004197%** del experimento no se explica con este modelo **físico-matemático**, dejando un **99.999958%** de fiabilidad en el ajuste estadístico utilizado en este trabajo.
- * Se elaboran cuatro artículos de carácter científico-didáctico titulados “Diseño de una fuente emisora de fotones en el espectro infrarrojo para el montaje del experimento ilustrativo de la difracción con el software FREEinfraROSSI y el vehículo motorizado **infraROSSI**” en el anexo A. pagina 31; “Calculo de la longitud de onda de la radiación de un diodo led infrarrojo, utilizando el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**” en el anexo B. pagina 39; “Ilustración de la ley de decaimiento de la densidad de flujo de radiación con el cuadrado inverso de la distancia de separación a la fuente; utilizando el modulo motorizado **infraROSSI** y su

software de control **FREEinfraROSSI**” en el anexo C. pagina 49; “Calculo del factor de perdida de la densidad de flujo de radiación emitida por un diodo led emisor infrarrojo, cuando esta radiación interactúa con la materia; utilizando el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**” anexo D. pagina 58; los cuales están en proceso de publicación en la revista brasileña de divulgación científica **ENSINO DE FÍSICA**²⁴ y se espera su pronta publicación.

- * El software de control **FREEinfraROSSI** es **freeSOFTWARE** y **openSOURCE** como se observa en el anexo E. pagina 68, pues según (Behlendorf et al., 1999) “el uso de software de fuentes abiertas en la educación contribuye a formar personas libres, independientes, críticas y autónomas; permite enseñar con herramientas adaptadas a la realidad del alumnado, crea una comunidad de conocimiento compartido; favorece en la persona la libertad de elección tecnológica; evoluciona rápidamente y permite una eficaz solución de los problemas; permite ahorrar costes en la implantación, el mantenimiento y la gestión de los centros educativos; facilita que los alumnos dispongan en su casa de las mismas herramientas educativas que utilizan en su centro educativo, y de forma **100%** legal; garantiza la seguridad; potencia la innovación de productos y servicios a través de empresas locales; en definitiva, el software de fuentes abiertas es un modelo educativo en sí mismo; libre, democrático, sostenible y tecnológicamente competitivo, y la opción ideal para el uso de la tecnología en el aula de clase y la enseñanza en ciencias.”
- * El modulo **infraROSSI** esta en estudio de viabilidad del trámite de patente a través de la superintendencia de industria y comercio de Colombia online <http://www.sic.gov.co/>, esperando poder obtener una patente de diseño del vehículo.
- * Los costos para la realización de este proyecto se detallan en la tabla 6 del anexo F. pagina 175, son de **89 dólares**, el software es gratuito y libre de ser operado por cualquier persona; estos costos convierten a este proyecto en el más económico en latino América para el estudio e **ilustración** de los fenómenos **físicos** de **difracción**, **atenuación** y **absorción** de las ondas **electromagnéticas** en el espectro infrarrojo; no se tiene referencia alguna de otro proyecto que ilustre estas propiedades, que sea

²⁴Para mayor información [on line] http://www.scielo.br/scielo.php?script=sci_serial&pid=1806-1117&lng=en&nrm=iso

software libre, hardware libre y de bajo costo.

Referencias

- Albella, J. M., y Martínez-Duart, J. M. (1996). *Fundamentos de electrónica física y microelectrónica*. Addison-Wesley Iberoamericana.
- Alcocer, P.-P. V., Gomez, J. M., Prat, A. M., y Albareda, X. M. (2006). *Programación en c++ para ingenieros*. Editorial Paraninfo.
- Behlendorf, B., Bradner, S., Hamerly, J., McKusick, T., Paquin, T., Perens, B., ... others (1999). Open sources: Voices from the open source revolution. *Retrieved February, 10, 2001*.
- Burtsch, K. O. (2004). *Linux shell scripting with bash*. Pearson Higher Education.
- Eaton, J. W., Bateman, D., y Hauberg, S. (1997). *Gnu octave*. Network theory London.
- Gautreau, R., Savin, W., y Velazquez Valle, D. (2001). *Física moderna*.
- Hecht, E., Dal Col, R., Talavera, R. W., y Pérez, J. M. G. (2000). *Óptica*. Addison Wesley.
- Herrador, R. E. (2009). Guía de usuario de arduino. *Universidad de Córdoba*, 8.
- Hoile, C., Bowman, C., Meijer, S. D., Corteil, B., Orsini, L., y Mott, T. (2014). *Raspberry pi and avr projects: Augmenting the pi's arm with the atmel atmega, ics, and sensors*. Maker Media, Inc.
- Kathirvel, A., y Srinivasan, R. (2009). Global mobile information system simulator in fedora linux. *ACM Computer Communication Review*, 1(1).
- McKinney, W. (2012). *Python for data analysis: Data wrangling with pandas, numpy, and ipython*. O'Reilly Media, Inc.
- Ostle, B. (1981). *Estadística aplicada*. Limusa.
- Pedraja, F. (2012). Modelo computacional de *gymnotus omarorum*: Un pez eléctrico de pulso con órgano distribuido.
- Serway, R. A. (1997). *física tomo ii*.
- Shalímová, K. V., y Grdiam, A. (1975). *Física de los semiconductores*.

**APÉNDICE A. DISEÑO DE UNA FUENTE EMISORA DE FOTONES PARA EL MONTAJE DEL EXPERIMENTO
ILUSTRATIVO DE LA DIFRACCIÓN**

**Apéndice A. Diseño de una fuente emisora de fotones
para el montaje del experimento ilustrativo de la
difracción**



Diseño de una fuente emisora de fotones en el espectro infrarrojo para el montaje del experimento ilustrativo de la difracción con el software FREEinfraROSSI y el vehículo motorizado infraROSSI

Diego Alberto Parra Garzón

diegoestudianteud1@gmail.com,

PhD. Julian Andrés Salamanca Bernal

jasalamanca@udistrital.edu.co

Universidad Distrital “Francisco José de Caldas”, Facultad de Ciencias y Educación
calle 3 No 26A-40 Bogotá-Colombia

Grupo de Física e Informática “FISINFOR”

Abstrac

This writing describes in detail the design of an external source of optoelectronic type in the infrared range of the electromagnetic spectrum, which feeds the infrared emitting diode and focuses electromagnetic radiation produced by this infrared LED; decreasing the scattering angle of photons radians on the diffraction grating 100 lines per millimeter, and providing a basis on which to rest the diffraction grating.

Keywords: Diffraction grating, optoelectronics source, dispersion, electromagnetic waves, infrared photons.

Resumen

El presente escrito, describe detalladamente el diseño de una fuente externa de tipo optoelectrónica en el rango infrarrojo del espectro electromagnético, la cual alimenta el diodo emisor infrarrojo y enfoca la radiación electromagnética producida por este led infrarrojo; disminuyendo el ángulo de dispersión de la radianza fotónica sobre la rejilla de difracción de 100 líneas por milímetro, y proporcionando una base en la que reposa esta rejilla de difracción.

Palabras clave: Rejilla de difracción, fuente optoelectrónica, dispersión, ondas electromagnéticas, infrarrojo, fotones.

Índice

INTRODUCCION	2
Diseno experimental	2
Diseño eléctrico	2
Montaje mecánico	2
Modelo físico matemático	3
Test del instrumento	5
Análisis de resultados	7
Conclusiones	7
Referencias	7

Índice de figuras

1. Vista lateral de la fuente emisora en infrarrojo FUENinfraROSSI.	2
2. Esquema del sistema de modulación de señal de (9 a 5)V de la fuente FUENinfraROSSI.	3
3. Esquema eléctrico del sistema de modulación de señal de (9 a 5)V de la fuente FUENinfraROSSI.	3
4. Esquema circuito impreso del sistema de modulación de señal de (9 a 5)V de la fuente FUENinfraROSSI.	3
5. Esquema eléctrico del emisor de fotones infrarrojos de la fuente FUENinfraROSSI. . .	3
6. Montaje mecánico, vista frontal de la fuente FUENinfraROSSI.	3
7. Montaje mecánico, vista superior de la fuente FUENinfraROSSI.	4

APÉNDICE A. DISEÑO DE UNA FUENTE EMISORA DE FOTONES PARA EL MONTAJE DEL EXPERIMENTO ILUSTRATIVO DE LA DIFRACCIÓN

Montaje mecánico

8.	Sistema óptico de la fuente FUENinfraROSSI.	4
9.	Transiciones con radiación entre una banda y los estados de impureza.	5
10.	Lente doble convexa (Serway, 1997).	5
11.	Montaje del modulo motorizado infraROSSI.	5
12.	Esquema de pines del sistema de control de hardware.	6
13.	Esquema de pines del sistema de control de avance.	6
14.	Esquema de pines del sensor de electro-recepción pasiva en infrarrojo.	6
15.	Montaje test fuente FUENinfraROSSI.	6
16.	Gráfica que ofrece el software FREEinfraROSSI para el enfoque óptico de la fuente FUENinfraROSSI.	7

Índice de tablas

1.	Análisis dispersión de la fuente FUENinfraROSSI con y sin sistema óptico.	7
----	---	---

INTRODUCCION

Para el experimento de difracción se necesita una fuente emisora de fotones, que sea monocromática y que su flujo este en fase, el instrumento más utilizado para este propósito es el láser que es distribuido por fabricas que surten de instrumentos científicos a las universidades y centros de investigación, marcas como PHYWE¹, LEYBOLD² entre otras, ofrecen estos productos de una excelente precisión en su longitud de onda, con una modulación de fase muy alta, pero también con precios no muy cómodos; por lo que este proyecto ofrece una alternativa económica para la fuente como se observa en la figura 1, a la que se le asignó el nombre de FUENinfraROSSI, esta cuenta con un enfoque óptico del tipo lente doble convexa (Giancoli, Association, et al., 1998); la densidad de flujo fotónico irradiado por el led es de $45mW/\Omega$, con una longitud de onda de 850 nm y una señal de voltaje de funcionamiento óptimo de (1.7 a 2)V voltaje continuo según parámetros del fabricante³.

Diseño experimental

La fuente FUENinfraROSSI tiene adaptado un modulador de señal de (9 a 5)V voltaje continuo, con un sistema de modulación de señal de salida de 2V voltaje continuo del tipo resistivo; un enfoque óptico del tipo lente doble convexa; una base para sostener firmemente la rejilla de difracción y un indicador led de encendido.

Diseño eléctrico

El diseño eléctrico consta de un sistema modulador de señal de (9 a 5)V voltaje continuo como se observa

DISEÑO EXPERIMENTAL

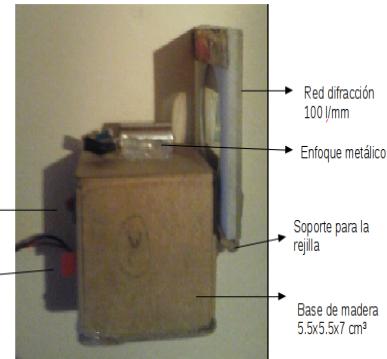


Figura 1. Vista lateral de la fuente emisora en infrarrojo FUENinfraROSSI.

en las figuras 2, 3 y 4, compuesto por el semiconductor lm7805cv⁴, dos condensadores electrolíticos de $10\ \mu F$, un interruptor, una fuente de alimentación del tipo batería de 9V, 170mAh, recargable de tipo cuadrada, un led de color rojo, una resistencia de $300\ \Omega$, un pin de salida a + 5V y otro pin a GND.

El sistema emisor de fotones como se observa en la figura 5, esta compuesto por un diodo led emisor infrarrojo con encapsulado epoxy de 3mm y una longitud de onda de 850 nm ; el cual crea una radianza de fotones infrarrojos los cuales tendrán que avanzar hasta el sistema de enfoque fotónico y de allí ha la rejilla de difracción de 100 líneas por milímetro, para irradiarse sobre el sistema de detección de electro-recepción pasiva (Pedraja, 2012), que posee el modulo motorizado infraROSSI para el estudio e ilustración del fenómeno físico ondulatorio de la difracción de ondas electromagnéticas en el espectro infrarrojo .

Montaje mecánico

El montaje de la fuente FUENinfraROSSI como se observa en las figuras 6 y 7, tiene dimensiones de $5.5x5.5x7.5\text{ cm}^3$ formando una caja hueca donde se coloca el sistema de alimentación de voltaje y sus elementos

¹Enlace a la sección online de láser PHYWE <https://www.phywe.com/en/experimentehierarchie/physics/age-16-19/optics/laser-optics/>

²Enlace a la sección online de láser LEYBOLD <http://www.photonics.1d-didactic.com/>

³Enlace a la hoja de datos del diodo led emisor infrarrojo suministrada por el fabricante online http://www.betlux.com/product/LED_lamp/infrared%20diode/BL-L314IR.PDF

⁴Enlace a la hoja de datos suministrada por el fabricante para el semiconductor lm7805cv <http://pdf1.alldatasheet.com/datasheet-pdf/view/9048/NSC/LM7805C.html>

MODELO FÍSICO MATEMÁTICO

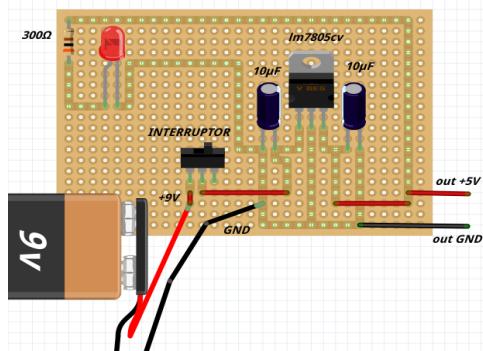


Figura 2. Esquema del sistema de modulación de señal de (9 a 5)V de la fuente FUENinfraROSSI.

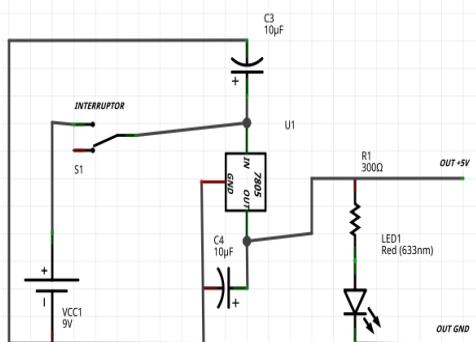


Figura 3. Esquema eléctrico del sistema de modulación de señal de (9 a 5)V de la fuente FUENinfraROSSI.

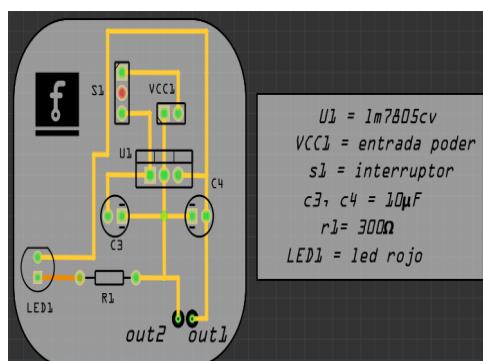


Figura 4. Esquema circuito impreso del sistema de modulación de señal de (9 a 5)V de la fuente FUENinfraROSSI.

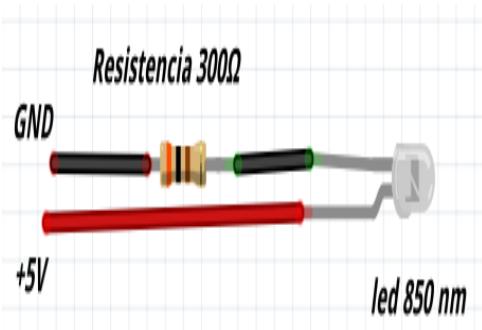


Figura 5. Esquema eléctrico del emisor de fotones infrarrojos de la fuente FUENinfraROSSI.

electrónicos junto a un interruptor de control de encendido y apagado , una base soporte donde se coloca la rejilla de difracción y una base soporte cilíndrica metálica donde se coloca el sistema de emisión fotónica. El sistema óptico

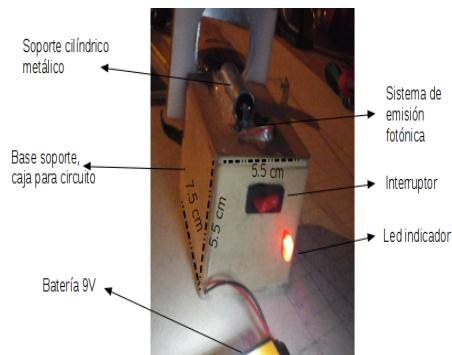


Figura 6. Montaje mecánico, vista frontal de la fuente FUENinfraROSSI.

de la fuente FUENinfraROSSI como se muestra en la figura 8, consta de una lente doble convexa, la cual se encarga de enfocar los fotones infrarrojos hasta llegar al obturador que permite el paso de los fotones enfocados por la lente doble convexa y a los que no están enfocados no les permite el paso, este obturador plástico tiene también la función de fijar la lente entre la base metálica circular de 0.5 cm de diámetro y el diodo emisor infrarrojo de 3 mm de diámetro.

Modelo físico matemático

Una vez se enciende el interruptor de la fuente como se observa en la figura 7, el semiconductor LM7805 cv modula la señal de entrada de voltaje de 9V voltaje continuo a una

MODELO FÍSICO MATEMÁTICO

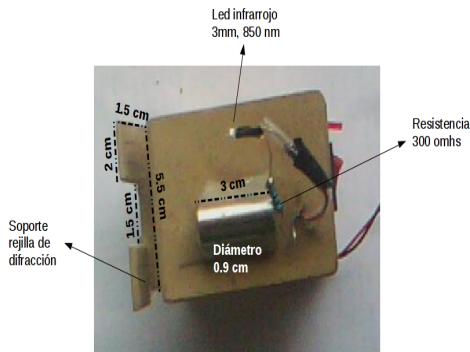


Figura 7. Montaje mecánico, vista superior de la fuente FUENinfraROSSI.

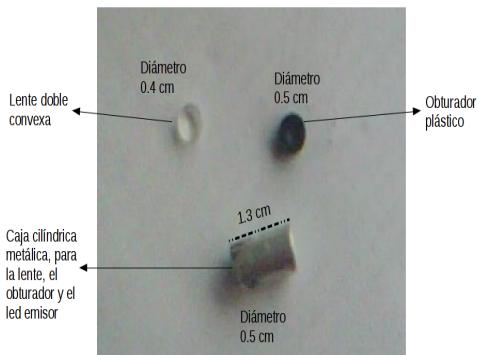


Figura 8. Sistema óptico de la fuente FUENinfraROSSI.

señal de 5V voltaje continuo esto con ayuda de los dos condensadores electrolíticos de $10\mu F$ que se encargan de filtrar el ruido de la señal tanto de entrada como de salida en el semiconductor (Alexander y Sadiku, 2002).

Al led⁵ indicador de encendido como se observa en las figuras 2 y 6, se le suministra $10mA$ para su corriente de trabajo, y 2V voltaje continuo para su voltaje de trabajo, por lo que la intensidad lumínica del led sera menor a 150 mcd .

El voltaje suministrado al led rojo de 2V, lo entrega la resistencia de 300Ω a medio vatio como se observa en las figuras 2 y 3. Utilizando la ley de ohm en la ecuación 1, en donde V es el voltaje disipado por la resistencia, I es la corriente que circula por la resistencia y el semiconductor led rojo según (Alexander y Sadiku, 2002); cambiando los valores de la ecuación 1,

$$V = RI. \quad (1)$$

R toma el valor de 300Ω e I toma el valor de 10 mA , el voltaje disipado por la resistencia es de 3V, de esta manera se obtiene el voltaje ideal para el indicador led.

Los fotones producidos por el diodo emisor de la fuente emisora de radiación infrarroja se dan según (Shalimova y Grdiam, 1975) (debido a que en los semiconductores, con estructura compleja de las bandas energéticas, son posibles las transiciones indirectas de los electrones de la banda de conducción a la de valencia acompañadas de la emisión de un fotón. En este caso la recombinación del electrón libre con el hueco se desarrolla con la aparición de un fonón, lo que asegura la conservación del cuasi impulso. Lo más probable es que el fonón sea emitido. Si en el semiconductor se desarrollan procesos de recombinación entre bandas tanto directa como indirectas, en el espectro de radiación se observan dos bandas de luminiscencia.

En la banda prohibida de los semiconductores reales existe una gran cantidad de estados localizados, que están ligados a los átomos de impureza, defectos de la estructura, infracciones de la periodicidad de la estructura en la superficie, entre otros. Estos estados localizados desempeñan un papel importante en los procesos de luminiscencia.

Las transiciones de los electrones de la banda de conducción a los niveles de los pequeños donadores (o de los huecos de la banda de valencia a los niveles de los pequeños aceptores), que hacen que estos últimos se neutralicen, pueden ser con radiación. En este caso es de esperar la aparición de luminosidad en la región infrarroja remota del espectro. Pero los cálculos muestran que en estas transiciones lo más probable es que sea emitido un fonón y no un fotón, es decir, que el proceso se desarrolle sin radiación. La recombinación con radiación se produce por lo general como viene mostrado en la figura 9.

Primero un electrón de la banda de conducción es capturado por un nivel local situado un poco más abajo que E_c^6 , y después se efectúa la recombinación de este electrón localizado con un hueco de la banda de valencia, la cual va acompañada de la emisión de un fotón. El electrón puede también realizar una transición con radiación de la banda de conducción y después recombinarse con un hueco. El estudio de los espectros de luminiscencia relacionados a diversas impurezas y defectos permite obtener información sobre estas infracciones de la estructura. En los semiconductores se pueden producir pares electrón-hueco ligados por la atracción coulombiana, es decir, excitones. Si uno de estos

⁵Hoja de datos del fabricante del diodo led rojo, online <https://www.sparkfun.com/datasheets/Components/LED/COM-09590-YSL-R531R3D-D2.pdf>

⁶ E_c es el nivel de energía de conducción.

TEST DEL INSTRUMENTO

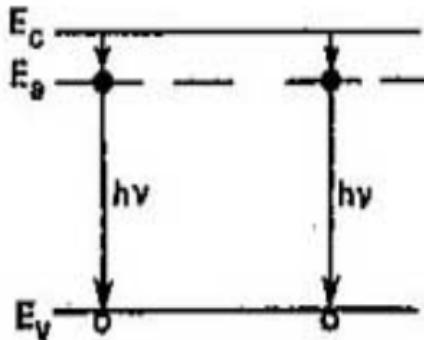


Figura 9. Transiciones con radiación entre una banda y los estados de impureza.

pares se aniquila, se produce la emisión de un fotón. La energía de esta radiación está dada por la ecuación 2,

$$hv = E_g - E_v \quad (2)$$

donde E es la energía de enlace del excitón.)

Ahora se tiene un flujo de fotones de energía hv saliendo del diodo emisor infrarrojo, según (Gautreau, Savin, y Velazquez Valle, 2001) (como los fotones viajan a la velocidad de la luz deben, de acuerdo con la teoría de la relatividad, tener una masa en reposo igual a cero; de aquí que su energía sea completamente cinética. Si un fotón existe, entonces se mueve a la velocidad de la luz, c ; si deja de moverse a velocidad c , deja de existir. Para $m_0 = 0$ la relación relativista momentum-energía se convierte en $E = pc$. de esta forma, cada fotón tiene un momentum como se muestra en la ecuación 3

$$p = \frac{E}{c} = \frac{hv}{c} = \frac{h}{\lambda}. \quad (3)$$

Desde el punto de vista cuántico, un haz de energía electromagnética se compone de fotones que se desplazan a la velocidad c . La intensidad del haz será proporcional al número de fotones que cruza un área unitaria por unidad de tiempo. Entonces, si el haz es monocromático (de una frecuencia), la intensidad I se obtendrá de la ecuación 4

$$I = (hv) \times \left(\frac{N}{A \times t} \right). \quad (4)$$

h es la constante de Plank que tiene un valor de $6,626 \times 10^{-34}(J * s)$; N es el número de fotones que pasan por segundo a través de la superficie; A es la superficie; t es el tiempo en segundos).

Ahora los fotones salen del diodo led emisor infrarrojo

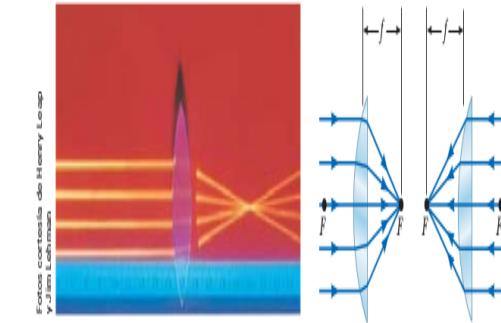


Figura 10. Lente doble convexa (Serway, 1997).

y avanzan hacia el sistema óptico el cual consiste en una lente doble convexa, la fuente emisora se coloca en el foco de la lente, pues como se muestra en la figura 10, cuando la fuente emite los fotones infrarrojos desde el punto focal, estos pasan por la lente y salen en forma paralela con menor dispersión que si no tuviera el sistema óptico.

Test del instrumento

Colocar el sensor de electro-recepción pasiva junto con el control de avance y el control de hardware en el vehículo motorizado infraROSSI como se observa en la figura 11. Los esquemas de pines de las partes utilizadas para el

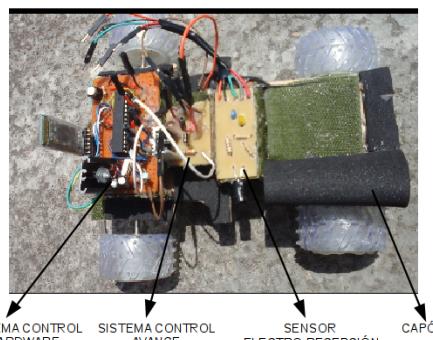


Figura 11. Montaje del modulo motorizado infraROSSI.

montaje se muestran en las figuras 12, 13 y 14; se debe conectar el pin ch2 con el pin ep5, el pin ch3 con el pin ca9, el pin de ch5 con el pin ep2, el pin de ch6 con el pin ep3, el pin ch7 con el pin ca2, el pin ch8 con el pin ca1, el pin ca4 con el pin ep4, el pin ca3 con el pin ep1,

por ultimo el pin ca7 y el pin ca8 son las conexiones al motor.

Se coloca el vehículo motorizado **infraROSSI** como

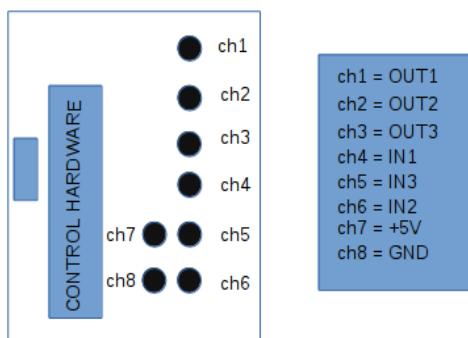


Figura 12. Esquema de pines del sistema de control de hardware.

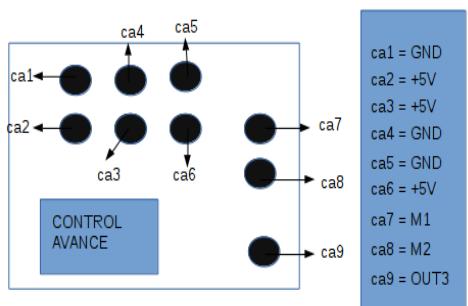


Figura 13. Esquema de pines del sistema de control de avance.

se muestra en la figura 15, a 30 cm de la fuente **FUENinfraROSSI** sin el sistema óptico, abrir una terminal de GNU-Linux y escribir en ella **infrarossi**, una vez abierta la ventana principal, dar click izquierdo en el botón **on**, una vez se establece la conexión bluetooth entre el ordenador y el vehículo **infraROSSI**, dar click izquierdo en el botón **difracción**, ahora el vehículo avanza hacia delante con un paso de 2 mm, recolectando datos con el sensor de electro-recepción pasiva; este paso lo realiza durante 20 cm.

Una vez termina de recoger los datos aparece una gráfica, no cerrar la gráfica, ni la ventana de color azul del programa, abrir el navegador de archivos y dirigirse a la carpeta Documentos/Free-infrarossi/free_infrarossi/

TEST DEL INSTRUMENTO

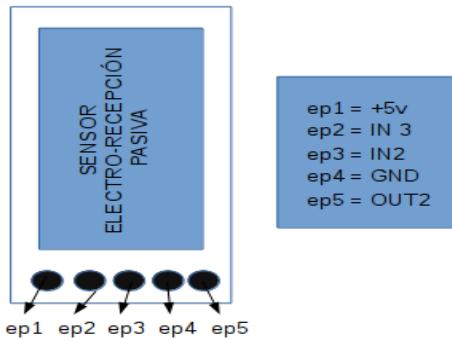


Figura 14. Esquema de pines del sensor de electro-recepción pasiva en infrarrojo.

datos; donde aparece un archivo con el nombre de dats1.dat, cambiar el nombre de este archivo a dater.dat y copiarlo a la carpeta Documentos/Free-infrarossi/free_infrarossi/Carpetas/FUENTE/.

Se coloca nuevamente el vehículo en la posición de



Figura 15. Montaje test fuente FUEInfraROSSI.

captura de datos como se muestra en la figura 15 pero esta vez con el sistema óptico; oprimir nuevamente el botón de difracción del software de control FREEinfrarossi, una vez termina de recoger los datos debe abrir el navegador de archivos y dirigirse a la carpeta Documentos/Free-infrarossi/free_infrarossi/datos; donde aparece un archivo con el nombre de dats1.dat, cambiar el nombre de este archivo a dats3.dat y copiarlo a la carpeta Documentos/Free-infrarossi/free_infrarossi/Carpetas/FUENTE/, hacer doble click sobre el programa est_fuente.py, esperar unos segundos mientras el programa hace los cálculos necesarios y una vez hecho esto

REFERENCIAS

mostrara la imagen que se observa en la figura 16.

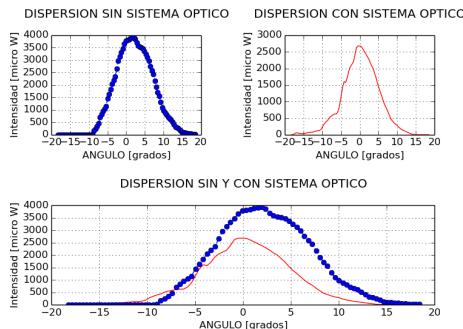


Figura 16. Gráfica que ofrece el software FREEinfraROSSI para el enfoque óptico de la fuente FUENinfraROSSI.

Análisis de resultados

Una vez termina de analizar los datos el programa se recomienda cerrar la ventana azul de software FREEinfraROSSI, el programa muestra la gráfica que se aprecia en la figura 16 en la cual se observa el flujo irradiado de fotones infrarrojos sobre el sistema de recolección del vehículo motorizado infraROSSI; en la tabla 1 se muestra el análisis de los datos con el sistema óptico colocado y sin colocar en la fuente, donde se observa que con el sistema óptico diseñado se obtiene una dispersión de 5 grados, esta dispersión se considera óptima para el experimento de difracción que ofrece el proyecto infraROSSI y su software de control FREEinfraROSSI.

Tabla 1

Analisis dispersión de la fuente FUENinfraROSSI con y sin sistema óptico.

Prueba	Separación	Dispersión (Grados)
Sin sistema óptico	30 cm	10
Con sistema óptico	30 cm	5

Nota. Todos los datos son aproximados a partir de la gráfica que ofrece el software FREEinfraROSSI, la cual se observa en la figura 16.

Conclusiones

- * Con el sistema óptico de la FUENinfraROSSI se logra disminuir la dispersión de la luz infrarroja emitida por el diodo led infrarrojo en 5 grados.
- * Al mejorar el enfoque óptico de la fuente FUENinfraROSSI, se disminuye la intensidad de emisión fotónica de la misma.
- * El experimento de difracción ofrece parámetros aceptables con la fuente FUENinfraROSSI, aunque se puede mejorar esta fuente remplazando el diodo emisor infrarrojo por un diodo láser infrarrojo.
- * El diseño de esta fuente es económico, pues el precio de fabricación no supera los 15 dólares.
- * El análisis físico, matemático que se observa en este proyecto puede servir de modelo pedagógico para el estudio de diversos tópicos de óptica y electrónica de profesionales y estudiantes de diversas carreras de ciencia e ingeniería.

Referencias

- Alexander, C., y Sadiku, M. (2002). *Fundamentos de circuitos eléctricos* mcgraw-hill. México.
 Gautreau, R., Savin, W., y Velazquez Valle, D. (2001). Física moderna.
 Giancoli, D. C., Association, B. D., y cols. (1998). *Physics: principles with applications* (Vol. 5). Prentice Hall Upper Saddle River, NJ, USA.:
 Pedraja, F. (2012). Modelo computacional de gymnotus omarorum: Un pez eléctrico de pulso con órgano distribuido.
 Serway, R. A. (1997). *física tomo ii*.
 Shalimova, K. V., y Grdiam, A. (1975). *Física de los semiconductores*.

**Apéndice B. Calculo de la longitud de onda de la
radiación de un diodo led infrarrojo**



Calculo de la longitud de onda de la radiación de un diodo led infrarrojo, utilizando el modulo motorizado infraROSSI y su software de control FREEinfraROSSI

Diego Alberto Parra Garzón

diegoestudianteud1@gmail.com,

PhD. Julian Andrés Salamanca Bernal

jasalamanca@udistrital.edu.co

Universidad Distrital “Francisco José de Caldas”, Facultad de Ciencias y Educación

calle 3 No 26A-40 Bogotá-Colombia

Grupo de Física e Informática “FISINFOR”

Abstrac

In this paper calculated the wavelength emitted by an infrared LED diode using a diffraction grating, infraROSSI motorized module and control software FREEinfraROSSI; a GNU-Linux environment, highlighting the functionality of this instrument in illustrating the diffraction property of electromagnetic waves and the duality wave - corpuscle.

Keywords: Motor module, infrared sensors, microcontroller module bluetooth, electromagnetic wave, diffraction.

Resumen

El presente trabajo calcula la longitud de onda emitida por un diodo led infrarrojo, utilizando una rejilla de difracción, el modulo motorizado infraROSSI y su software de control FREEinfraROSSI; en un entorno GNU-linux, resaltando la funcionalidad de este instrumento en la ilustración de la difracción como propiedad de las ondas electromagnéticas y su dualidad onda-corpúsculo.

Palabras clave: Modulo motorizado, sensores infrarrojos, microcontrolador, modulo bluetooth, ondas electromagnéticas, difracción.

Índice

Introducción	2	2. Transiciones con radiación entre una banda y los estados de impureza.	3
Marco teórico	2	3. Imagen del montaje para la difracción utilizando el modulo motorizado infraROSSI, una fuente emisora de fotones de $\lambda = 850nm$ y una red de difracción de 100 líneas por milímetro.	5
Montaje experimental	7	4. Absorción de la luz por un semiconductor.	5
Materiales del montaje	7	5. Relación irradiancia vs voltaje.	7
Montaje	7	6. Imagen del montaje para la difracción utilizando el modulo motorizado infraROSSI.	8
Análisis de resultados	8	7. Esquema de pines del sistema de control de hardware.	8
Conclusiones	9	8. Esquema de pines del sistema de control de avance.	8
Referencias	9	9. Esquema de pines del sensor de electro-recepción pasiva en infrarrojo.	8
Índice de figuras		10. Interfaz grafica software FREEinfraROSSI .	9
1. Vista lateral de la fuente emisora en infrarrojo FUENinfraROSSI.	3		

- | | |
|---|---|
| 11. Imagen generada por el programa FREEinfraROSSI. | 9 |
|---|---|

Índice de tablas

- | | |
|--|---|
| 1. Análisis del experimento de difracción con el vehículo motorizado infraROSSI y su software de control FREEinfraROSSI. | 9 |
|--|---|

Introducción

Los fenómenos de las ondas siempre han fascinado los pensamientos de los seres humanos; es allí donde la física con ayuda de la matemática muestran su majestuosidad, al explicar de manera muy detallada estos fenómenos de transporte; la difracción es una de estas propiedades, la cual está muy presente en la vida diaria y con la ayuda del modulo motorizado infraROSSI y su software de control FREEinfraROSSI se ilustra este fenómeno físico y se calcula la longitud de onda propia producida por un diodo infrarrojo.

Según (Hecht, Dal Col, Talavera, y Pérez, 2000) “Un cuerpo opaco colocado a medio camino entre una pantalla y una fuente puntual proyecta una sombra complicada hecha en regiones claras y oscuras muy diferentes de las que podría esperarse de los principios básicos de la óptica geométrica.

El trabajo de Francesco Grimaldi en el siglo XVII fue el primer estudio detallado que se publicó sobre esta desviación de la luz de su propagación rectilínea. A la que denomino difracción. El efecto es una característica general de los fenómenos ondulatorios que ocurren donde quiera que una parte de un frente de onda ya sea sonido, onda material o luz, esté obstruida de alguna manera.

Si al encontrar un obstáculo transparente u opaco se altera la amplitud o la fase de una región del frente de onda, esto produciría difracción. Los varios segmentos del frente de onda que se propagan más allá del obstáculo interfieren, produciendo aquella distribución de densidad de energía particular denominada figura de difracción. No hay distinción física significativa entre interferencia y difracción.

Sin embargo, se ha vuelto algo común, aunque no siempre apropiado, hablar de interferencia cuando se analiza la superposición de solamente unas pocas ondas y de difracción cuando se trata de un gran número de ondas.

La red de difracción es un conjunto repetitivo de elementos difractores de una onda emergente, bien sean aberturas u obstáculos, que tienen el efecto de producir alteraciones periódicas en la fase, amplitud o ambas. Uno de los más simples de tales conjuntos es la configuración de rendijas múltiples. Parece que fue inventado por el astrónomo

MARCO TEÓRICO

americano David Rittenhouse hacia 1785.

Algunos años más tarde Joseph Von Fraunhofer redescubrió, por su cuenta, este principio y siguió aportando un buen número de contribuciones importantes tanto a la teoría como a la tecnología de redes.

Los primeros dispositivos eran en realidad conjuntos de rendijas múltiples, que consistían por lo general en un retículo de alambre muy fino o hilo enrollado y extendido entre dos tornillos paralelos que servían como espaciadores.

Al pasar a través de semejante sistema, un frente de onda se encuentra con regiones opacas y transparentes alternadas, sufriendo una modulación en amplitud. Así mismo, una configuración múltiple de rendijas se denomina red de transmisión de amplitud.

Otra forma más corriente de red de transmisión se hace rayando o raspando unas hendiduras paralelas en la superficie de una lámina de cristal clara y plana. Cada raspadura sirve como fuente de luz esparsa, formando juntas un conjunto regular de fuentes lineales paralelas.

Cuando la red es totalmente transparente, de tal manera que la modulación en amplitud sea despreciable, las variaciones regulares del espesor óptico a través del retículo dan una modulación en fase y tenemos lo que se denomina red de transmisión de fase.

En la representación de Huygens-Fresnel podemos visualizar los trenes de onda como radiados con diferentes fases sobre la superficie de la red.

Un frente de onda emergente contiene, por consiguiente, unas variaciones periódicas en su forma más que en su amplitud lo cual, a su vez, equivale a una distribución angular de las ondas constitutivas”.

Marco teórico

El modulo motorizado infraROSSI necesita de una fuente emisora de fotones infrarrojos externa como se muestra en la figura 1.

Los fotones producidos por el diodo emisor de la fuente emisora de radiación infrarroja antes de pasar estos por la rejilla de difracción se dan según (Shalimova y Grdiam, 1975) "debido a que en los semiconductores, con estructura compleja de las bandas energéticas, son posibles las transiciones indirectas de los electrones de la banda de conducción a la de valencia acompañadas de la emisión de un fotón. En este caso la recombinación del electrón libre con el hueco se desarrolla con la aparición de un fonón,

MARCO TEÓRICO

lo que asegura la conservación del cuasi impulso. Lo más probable es que el fonón sea emitido.

Si en el semiconductor se desarrollan procesos de recombinación entre bandas tanto directa como indirectas, en el espectro de radiación se observan dos bandas de luminiscencia.

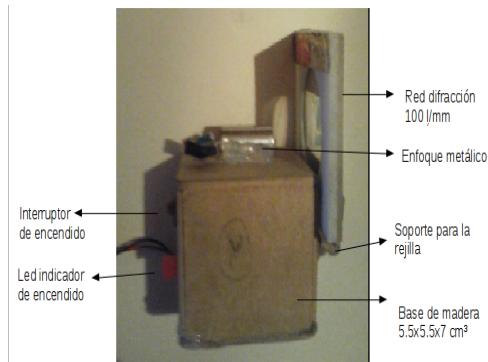


Figura 1. Vista lateral de la fuente emisora en infrarrojo FUENinfraROSSI.

En la banda prohibida de los semiconductores reales existe una gran cantidad de estados localizados, que están ligados a los átomos de impureza, defectos de la estructura, infracciones de la periodicidad de la estructura en la superficie, entre otros. Estos estados localizados desempeñan un papel importante en los procesos de luminiscencia.

Las transiciones de los electrones de la banda de conducción a los niveles de los pequeños donadores (o de los huecos de la banda de valencia a los niveles de los pequeños aceptores), que hacen que estos últimos se neutralicen, pueden ser con radiación. En este caso es de esperar la aparición de luminosidad en la región infrarroja remota del espectro.

Pero los cálculos muestran que en estas transiciones lo más probable es que sea emitido un fonón y no un fotón, es decir, que el proceso se desarrolle sin radiación. La recombinación con radiación se produce por lo general como viene mostrado en la figura 2.

Primero un electrón de la banda de conducción es capturado por un nivel local situado un poco más abajo que el nivel de energía de conducción, y después se efectúa la recombinación de este electrón localizado con un hueco de la banda de valencia, la cual va acompañada de la emisión de un fotón. El electrón puede también realizar una transición con radiación de la banda de conducción y

después recombinarse con un hueco.

El estudio de los espectros de luminiscencia relacionados a diversas impurezas y defectos permite obtener información sobre estas infracciones de la estructura. Durante la

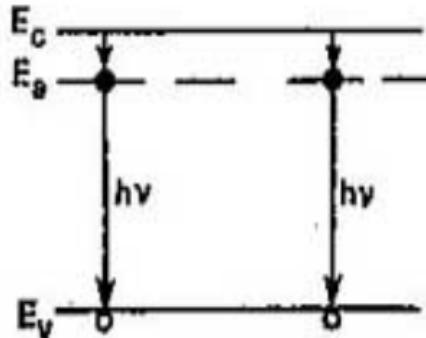


Figura 2. Transiciones con radiación entre una banda y los estados de impureza.

absorción de la luz puede surgir en los semiconductores pares electrón-hueco ligados por la atracción coulombiana, es decir, excitones. Si uno de estos pares se aniquila, se produce la emisión de un fotón. La energía de esta radiación es:

$$hv = E_g - E, \quad (1)$$

donde E es la energía de enlace del excitón. Como el excitón puede tener estados excitados, la radiación dada a la recombinación excitónica puede consistir en una serie de rayas estrechas correspondientes a las transiciones desde los estados excitados."

Una vez producido los fotones infrarrojos con energía hv se hacen pasar por una red de difracción de 100 líneas por milímetro, esta radiación en vez de comportarse como un corpúsculo como lo venía haciendo, se comporta como una onda y cada abertura de la red de difracción se convierte en una fuente, por lo que se tiene un gran número de fuentes.

Según (Hecht et al., 2000) "Al reflejarse en esta clase de red, la luz esparcida por las varias características periódicas de la superficie llegarán a un punto P con una relación de fase definida. El patrón de interferencia correspondiente engendrado después de la reflexión es muy similar al que se produce por transmisión. Las redes diseñadas específicamente para funcionar de esta manera se denominan redes de reflexión de fase. Tradicionalmente, las redes de esta clase son rayadas sobre películas finas de aluminio que han sido evaporadas sobre bloques de vidrio ópticamente planos.

MARCO TEÓRICO

Puesto que el aluminio es bastante blando, hay menos desgaste de la herramienta de rayar de diamante, siendo tambien mejor reflector en la región ultravioleta.

Si miramos perpendicularmente a través de una red de transmisión hacia una fuente lineal paralela distante, los ojos servirían como lente de enfoque para la distribución de difracción.

Como un puente simple aunque lógico entre los estudios de la interferencia y de la difracción se considera un conjunto de N osciladores puntuales coherentes (o antenas emisoras), todos ellos idénticos incluso en su polarización. Por ahora, hay que suponer que los osciladores no tengan diferencia de fase intrínseca¹, es decir, cada uno tiene el mismo ángulo de fase inicial.

Todos los rayos son casi paralelos, encontrándose en un punto P muy distante. Si la extensión espacial del conjunto es comparativamente pequeña, las amplitudes de onda individuales que lleguen a P serán esencialmente iguales, habiendo recorrido casi las mismas distancias, esto es:

$$E_0(r_1) = E_0(r_2) = \dots = E_0(r_n) = E_0(r) \quad (2)$$

La suma de los trenes de onda esféricos interferentes produce un campo eléctrico en P proporcionado por la parte real de

$$\vec{E} = E_0(r)e^{i(kr_1-\omega t)} + E_0(r)e^{i(kr_2-\omega t)} + \dots + E_0(r)e^{i(kr_N-\omega t)} \quad (3)$$

Por tanto ahora

$$\vec{E} = E_0(r)e^{(-i\omega t)}e^{ikr_1} \times [1 + e^{ik(r_2-r_1)} + e^{ik(r_3-r_1)} + \dots + e^{ik(r_N-r_1)}] \quad (4)$$

La diferencia de fases entre fuentes adyacentes se obtiene de la expresión $\delta = k_0\Lambda$, y puesto que $\Lambda = n dsin(\theta)$, en un medio con índice n, $\delta = kdsin(\theta)$, de esto se deduce que $\delta = k(r_2 - r_1)$, $2\delta = k(r_3 - r_1)$.

Entonces el campo de P puede escribirse como:

$$\vec{E} = E_0(r)e^{(-i\omega t)}e^{ikr_1} \times [1 + (e^{i\delta}) + (e^{i\delta})^2 + (e^{i\delta})^3 + \dots + (e^{i\delta})^{N-1}] \quad (5)$$

La serie geométrica entre paréntesis tiene el valor:

$$(e^{i\delta N} - 1)/(e^{i\delta} - 1)$$

que puede ordenarse así:

$$\frac{e^{i\delta N/2}[e^{i\delta N/2} - e^{-i\delta N/2}]}{e^{i\delta/2}[e^{i\delta/2} - e^{-i\delta/2}]} \quad (6)$$

o de manera equivalente

$$e^{i(N-1)\delta/2} * \left(\frac{\sin(N\delta/2)}{\sin(\delta/2)} \right) \quad (7)$$

Entonces el campo se transforma en:

$$\vec{E} = E_0(r)e^{(-i\omega t)}e^{i[kr_1+(N-1)\delta/2]} * \left(\frac{\sin(N\delta/2)}{\sin(\delta/2)} \right) \quad (8)$$

Si se define R como la distancia desde el centro de la linea de los osciladores hasta el punto P, es decir:

$$R = \frac{1}{2}(N-1)dsin(\theta) + r_1 \quad (9)$$

Entonces la ecuación 8 se convierte en:

$$\vec{E} = E_0(r)e^{i(kR-\omega t)} \left(\frac{\sin(N\delta/2)}{\sin(\delta/2)} \right) \quad (10)$$

Finalmente, la distribución de densidad de flujo dentro de la distribución de difracción debida a N fuentes puntuales distantes, idénticas y coherentes en una disposición lineal, es proporcional a $EE^*/2$ para E compleja o

$$I = I_0 \frac{\sin^2(N\delta/2)}{\sin^2(\delta/2)} \quad (11)$$

donde I_0 es la densidad de flujo que saliendo desde cualquier fuente puntual llegue a P.

La dependencia funcional de I con θ queda más clara en la forma

$$I = I_0 \frac{\sin^2[(Nk\delta/2)\sin(\theta)]}{\sin^2[(k\delta/2)\sin(\theta)]} \quad (12)$$

El término $\sin^2[(Nk\delta/2)\sin(\theta)]$ se somete a unas fluctuaciones rápidas, mientras que las fluctuaciones que la modula, $\sin^2[(k\delta/2)\sin(\theta)]$, varía de manera relativamente lenta. La expresión combinada da lugar a una serie de picos principales agudos separados por picos pequeños complementarios.

La ecuación 13, es la que describe el fenómeno físico y se denomina ecuación de red para incidencia normal.

$$d * \sin(\theta_m) = m\lambda \quad (13)$$

Los valores de m especifican el orden de diversos máximos principales. Para una fuente que tenga un espectro continuo ancho, la imagen de orden cero, $m = 0$, corresponde a la imagen blanca de la fuente no desviada $\theta_0 = 0$.

¹Este es el caso ideal, el cual en el experimento se da el caso que los fotones infrarrojos producidos por el diodo led infrarrojo no están en fase.

MARCO TEÓRICO

La ecuación de red depende de λ y así, para cualquier valor de $m \neq 0$, las distintas imágenes coloreadas de la fuente correspondientes a ángulos ligeramente diferentes (θ_m), se dispersa en un espectro continuo.

Las regiones ocupadas por los débiles máximos secundarios aparecerán como bandas aparentemente desprovistas de luz. El espectro de primer orden $m = \pm 1$ aparece a cada lado de $\theta = 0$ y es seguido, junto con intervalos alternados de oscuridad, por los espectros de orden superior, $m = \pm 1, \pm 2, \dots$

Ahora se produce un patrón de difracción que alcanza al detector infrarrojo del modulo motorizado **infraROSSI** como se observa en la figura 3, el cual avanza en linea recta a 45 centímetros de la fuente emisora de fotones infrarrojos, perpendicular a la incidencia de los patrones de difracción.

El detector del modulo motorizado **infraROSSI** es un diodo receptor infrarrojo; según (Shalimova y Grdiam, 1975) “cuando un haz de radiación monocromática u homogénea traspasa una sustancia, debido a la reflexión y absorción su intensidad disminuye”.

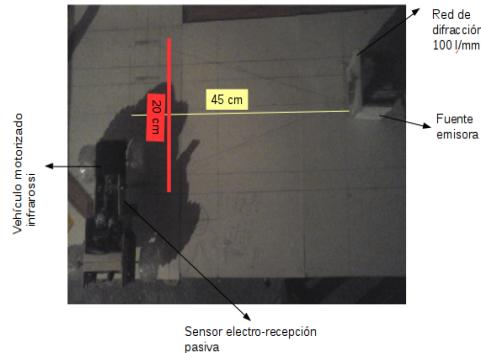


Figura 3. Imagen del montaje para la difracción utilizando el modulo motorizado **infraROSSI**, una fuente emisora de fotones de $\lambda = 850\text{nm}$ y una red de difracción de 100 líneas por milímetro.

Supongamos que la fracción de energía reflejada en el extremo del cuerpo sea R , magnitud que lleva el nombre de factor de reflexión. Si la intensidad de la luz incidente es I_0 y la reflejada I_R , entonces

$$R = \frac{I_R}{I_0} \quad (14)$$

La dependencia del factor de reflexión respecto a la

frecuencia $R(\omega)$ o de la longitud de onda $R(\lambda)$ se llama espectro de reflexión.

Designemos por I la intensidad de la luz que incide en la capa dx , cómo se muestra en la figura 4. En tal caso, debido a la absorción de la luz en esta capa la intensidad de radiación se reduce en la magnitud dI .

La cantidad de energía absorbida dI es proporcional a la cantidad de energía incidente en la capa y el espesor de la capa absorbente:

$$-dI = \alpha I dx \quad (15)$$

El coeficiente de proporcionalidad α , que expresa la cantidad de energía absorbida del haz de intensidad unidad por la capa de espesor unidad, se llama factor de absorción.

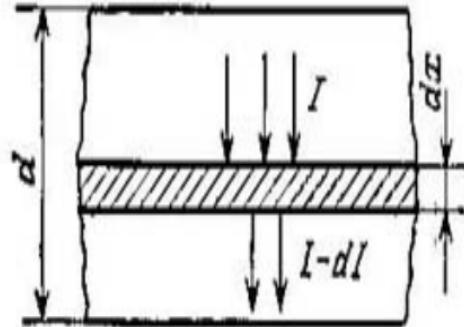


Figura 4. Absorción de la luz por un semiconductor.

Integrando la ecuación 15 sin tener en cuenta la reflexión

$$\int_{I_0}^I \frac{dI}{I} = - \int_0^d \alpha dx \quad (16)$$

se obtiene la expresión

$$I = I_0 e^{-\alpha d} \quad (17)$$

conocida con el nombre de ley de Buger-Lambert. La magnitud α es una característica del medio absorbente y depende la longitud de onda de la radiación. La dependencia del factor de absorción respecto de la frecuencia $\alpha(\omega)$ o de la longitud de onda $\alpha(\lambda)$ se llama espectro de absorción de la sustancia.

Suponiendo que se tiene N centros de absorción. Designamos por σ la probabilidad de absorción de un haz monofotón por un centro de absorción de un fotón en la

MARCO TEÓRICO

unidad de tiempo.

La sección eficaz σ depende de la energía del fotón y de la naturaleza de los centros absorbentes. De acuerdo con la ecuación $(\sigma N)^{-1}$ que es la longitud media de recorrido libre de un fotón l_f en un medio absorbente, es decir,

$$l_f = \frac{1}{\sigma N} \quad (18)$$

Mientras que el factor de absorción

$$\alpha = \frac{1}{l_f} \quad (19)$$

es la probabilidad de absorción de un fotón en la unidad de longitud. Suponiendo que en el semiconductor existen centros de absorción de diferente naturaleza. Si N_i centros de absorción se caracterizan por la sección eficaz σ_i , entonces

$$\alpha_i(\omega) = \sigma_i N_i \quad (20)$$

el factor de absorción total de la sustancia α es la suma de los factores de absorción parciales:

$$\alpha(\omega) = \sum_i \alpha_i(\omega) = \sum_i \sigma_i(\omega) N_i \quad (21)$$

Por lo tanto, el espectro de absorción total se compone de los espectros de absorción de los distintos centros de absorción.

Al interactuar los electrones del semiconductor con la radiación electromagnética deben cumplirse dos leyes: la ley de conservación de la energía E y la ley de conservación del casi impulso \mathbf{p} , y después de interactuar se tiene E' y \mathbf{p}' , estas leyes se escriben en la forma

$$E' = E + \hbar\omega \quad (22)$$

$$\mathbf{p}' = \mathbf{p} + \hbar\vec{\eta} \quad (23)$$

La absorción de la radiación en los semiconductores puede estar vinculada con la variación del estado energético de los electrones libres o enlazados con los átomos, así como la variación de la energía vibratoria (oscilante) de los átomos de la red.

Debido a esto, en los semiconductores se distinguen cinco tipos fundamentales de absorción óptica: intrínseca, excitónica, por portadores de carga libres, extrínseca y absorción de la luz por la red cristalina."

Al iluminar el diodo receptor infrarrojo con esta energía

radiante, según (Albella y Martínez-Duart, 1996) "en el semiconductor por cada fotón absorbido se rompe un enlace y se crea un par electrón-hueco. Es importante destacar que no todos los portadores fotogenerados contribuyen a la conducción, ya que una fracción importante de ellos se recombinan antes de llegar al extremo correspondiente del semiconductor.

El cálculo del incremento de corriente ΔI_e , debida al exceso de electrones generados en la banda de conducción, Δn , es

$$\Delta I_e = q\mu_e(\Delta n)ES, \quad (24)$$

siendo E el campo eléctrico aplicado, μ_e la movilidad de los electrones y S la sección transversal del fotoconductor. En condiciones de iluminación, el estado estacionario se alcanza cuando la velocidad de generación de portadores en todo el volumen del semiconductor, G , se iguala a la velocidad de recombinación, R , es decir $R = G$. para un conductor intrínseco en el cual existe un exceso de portadores, $\Delta n = \Delta p$, la velocidad de recombinación de los portadores vendrá dada por:

$$R = \frac{\Delta n}{\tau} = \frac{\Delta p}{\tau}, \quad (25)$$

siendo τ el tiempo de vida media de los portadores fotogenerados. En un semiconductor de longitud L en el que suponemos que el espesor es suficiente para que toda la luz que incide sobre él, sea absorbida en su interior, se tiene ahora para la velocidad de generación de portadores en la banda de conducción:

$$G = \eta n_{fot} = \eta \frac{P_i}{S L} \frac{1}{\hbar v}, \quad (26)$$

siendo n_{fot} el número de fotones incidentes en el semiconductor por unidad de volumen y de tiempo, y η la eficiencia de la conversión en la generación de portadores.

El valor n_{fot} se calcula a través del cociente entre la potencia de la luz incidente, P_i , y la energía de la radiación, $\hbar v$, dividido a su vez por el volumen del material.

Sabiendo que la velocidad de arrastre de los electrones por el campo eléctrico viene dada por: $v_e = \mu_e E$, las igualdades anteriores permiten escribir para la corriente de electrones fotogenerada entre los dos electrodos:

$$\Delta I_e = qv_e \eta \frac{P_i}{L} \frac{1}{\hbar v} \tau, \quad (27)$$

si se tiene en cuenta que el cociente $t_r = L/v_e$, representa el

MONTAJE EXPERIMENTAL

tiempo de tránsito de los electrones entre los dos electrodos, resulta para ΔI_e :

$$\Delta I_e = q\eta \frac{P_i}{hv} \frac{\tau}{t_r}, \quad (28)$$

con una expresión similar para la corriente de huecos en la banda de valencia. En la ecuación 28, el factor $q\eta(P_i/hv) = I_{fot}$ tiene dimensiones de corriente y representa la velocidad de generación de carga en el semiconductor. En función de este parámetro, se define el factor de ganancia del fotoconductor a través del cociente:

$$\frac{\Delta I}{I_{fot}} = \frac{\tau}{t_r}, \quad (29)$$

ahora bien un diodo operando con cierto voltaje aplicado, V , en presencia de radiación electromagnética capaz de excitar portadores a través de la banda prohibida dejara pasar una intensidad I dada por:

$$I = I_0[e^{(qV/kT)} - 1] - I_L, \quad (30)$$

donde I_0 representa la corriente típica de un diodo, I_L representa la corriente debida a los portadores generados. El valor de I_L puede calcularse de la siguiente manera:

$$I_L = qGS(L_e - L_h), \quad (31)$$

siendo G el número de portadores generados por unidad de volumen y de tiempo y S el área de la sección transversal del diodo. L_e y L_h representan las longitudes de difusión de los electrones y huecos. El dispositivo funciona entonces como detector del nivel de iluminación convirtiendo una señal óptica en señal eléctrica."

Como la irradiancia de fotones infrarrojos es proporcional al inverso del cuadrado de la distancia de separación a la fuente, y la función voltaje que mide el modulo motorizado **infraROSSI** es proporcional al inverso del cuadrado de la distancia de separación de la fuente al vehículo **infraROSSI**; se realiza el cociente de la intensidad irradiada en el diodo led que es según el fabricante de $20 \text{ mW}/\Omega$, sobre el voltaje producido en el sensor que es de 5000 mV ; como se muestra en la figura 5.

Una vez absorbidos los fotones infrarrojos, el microcontrolador atmega 328 del vehículo motorizado **infraROSSI**, mide la relación de voltaje en el sensor de electro-recepción pasiva que es proporcional a la irradiancia del emisor infrarrojo la cual equivale a $4 \mu\text{W}/\text{mV}$ lo que significa que por cada $4 \mu\text{W}$ de radiación incidente el sensor esta arrojando en su salida 1 mV .

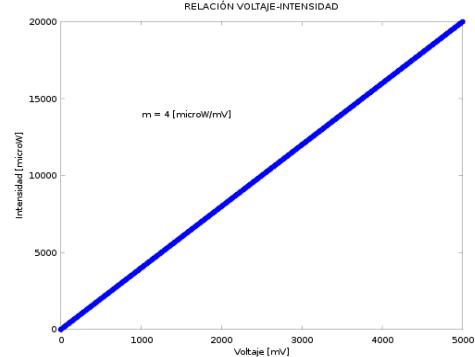


Figura 5. Relación irradiancia vs voltaje.

Montaje experimental

Materiales del montaje

Para la realización de este montaje se requiere los siguientes materiales:

- Ordenador con sistema operativo GNU-Linux.
- Modulo motorizado **infraROSSI**.
- Software de control **FREEinfraROSSI**.
- Fuente emisora de fotones infrarrojos.
- Modulo bluetooth para pc.
- Rejilla de difracción de 100 líneas por milímetro

Montaje

Colocar el sensor de electro-recepción pasiva junto con el control de avance y el control de hardware en el vehículo motorizado **infraROSSI** como se observa en la figura 6. Los esquemas de pines de las partes utilizadas para el montaje se muestran en las figuras 7 , 8 y 9 ; se debe conectar el pin ch2 con el pin ep5, el pin ch3 con el pin ca9, el pin de ch5 con el pin ep2, el pin de ch6 con el pin ep3, el pin ch7 con el pin ca2, el pin ch8 con el pin ca1, el pin ca4 con el pin ep4, el pin ca3 con el pin ep1, por ultimo el pin ca7 y el pin ca8 son las conexiones al motor.

Colocar la fuente² emisora de fotones infrarrojos con la red de difracción de 100 líneas por milímetro, frente a ella

²Esta fuente se elaboró con un diodo led infrarrojo de 850 nm y un encapsulado epoxi de 3 mm.

ANÁLISIS DE RESULTADOS

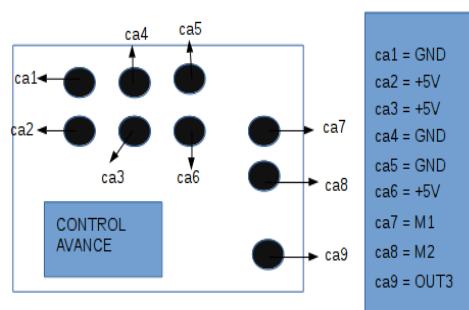
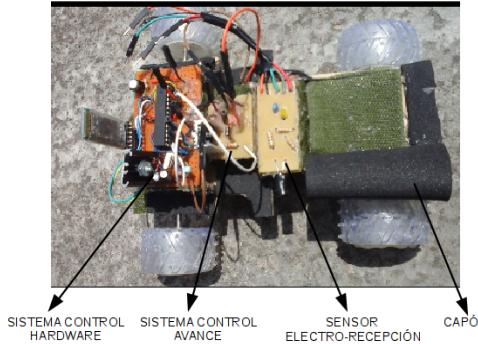


Figura 6. Imagen del montaje para la difracción utilizando el modulo motorizado **infraROSSI**.

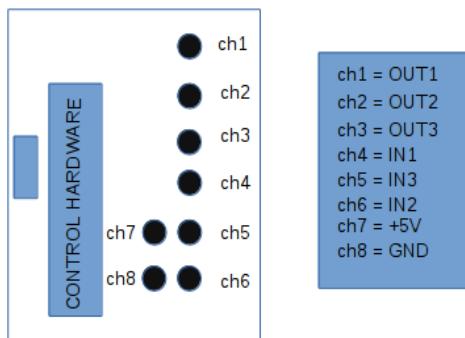


Figura 7. Esquema de pines del sistema de control de hardware.

colocar el modulo motorizado **infraROSSI** a 45 cm de la fuente emisora, como se muestra en la figura 3.

Abrir una terminal de GNU-Linux y escribir **infrarossi**, oprimir enter y la clave de superusuario, luego de abrir el programa debe oprimir el botón on, esperar que se empareje el bluetooth, una vez emparejado el bluetooth el programa desplegará un tercer menú ver figura 10, oprimir el botón de difracción y esperar que el programa tome los datos necesarios.

Luego de capturar los datos aparecerá la gráfica de los datos, oprima doble click izquierdo en el máximo de interferencia y sin soltar el cursor lleve la linea al siguiente máximo de interferencia, suelte el botón del cursor e inmediatamente aparecerá la gráfica con el análisis de longitud de onda infrarroja del diodo, tal como se muestra en la figura 11.

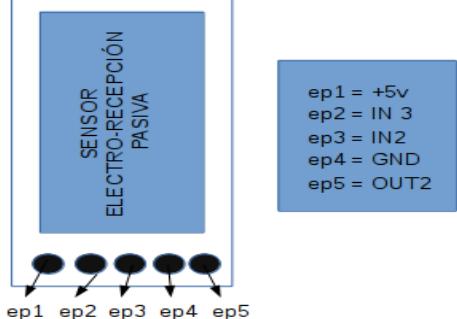


Figura 9. Esquema de pines del sensor de electro-recepción pasiva en infrarrojo.

Análisis de resultados

La gráfica de análisis y los datos capturados se almacenan dentro del archivo llamado Carpetas/Difraccion con la fecha y hora del análisis de datos. El programa **FREEinfraROSSI** cuando termina de recoger los datos realiza un análisis estadístico de los mismos como se muestra en la figura 11, en una ventana aparte realiza la gráfica de los datos y predice la longitud de onda del diodo emisor infrarrojo con un error en la medida aceptable, no superior al 4 % .

La tabla 1, muestra los resultados de nueve repeticiones del montaje para calcular la longitud de onda λ de la radiación producida por un diodo led emisor infrarrojo, fueron capturados y analizados con el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**.

REFERENCIAS

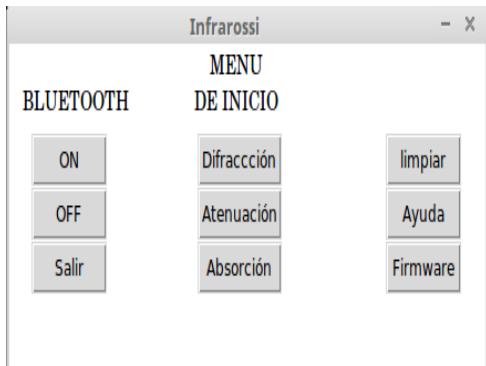


Figura 10. Interfaz grafica software FREEinfraROSSI

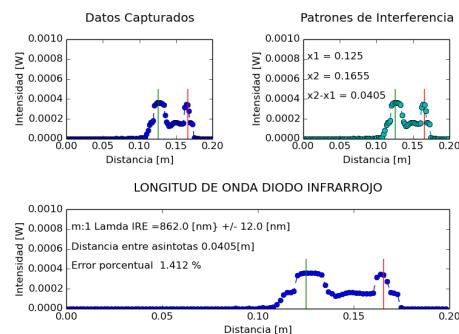


Figura 11. Imagen generada por el programa FREEinfraROSSI.

La distancia que hay entre pico y pico del patrón obtenido es ΔX , esta viene en metros, λ_{exp} . es la longitud de onda obtenida a través del experimento, viene en nanometros, λ_{teo} . es la longitud de onda teórica del diodo emisor infrarrojo, la cual corresponde a 850nm suministrada por el fabricante³, el error porcentual es la ultima columna de la tabla 1 es 851,77nm dando un error porcentual de 0,209 % de la longitud de onda suministrada por el fabricante.

Tabla 1

Análisis del experimento de difracción con el vehículo motorizado infraROSSI y su software de control FREEinfraROSSI.

Prueba	ΔX [m]	$\lambda_{exp}[\text{nm}]$	$\lambda_{teo}[\text{nm}]$	error %
1	0.0401	854	850	0.471
2	0.0405	862	850	1.412
3	0.0405	862	850	1.412
4	0.0405	862	850	1.412
5	0.0397	845	850	0.588
6	0.0393	837	850	1.529
7	0.0397	845	850	0.588
8	0.0393	845	850	1.529
9	0.0401	854	850	0.401

Nota. Datos obtenidos de nueve pruebas para medir la longitud de onda producida por un diodo emisor infrarrojo, con el modulo motorizado infraROSSI y su software de control FREEinfraROSSI.

Conclusiones

- * La longitud de onda media del experimento es de $\lambda_{exp} = 851,77\text{nm}$, la cual difiere en 1,77nm de la longitud de onda del diodo emisor infrarrojo.
- * El modulo motorizado infraROSSI y su software de control ilustran de manera cuantitativa y cualitativa fenómenos ondulatorios como la difracción e interferencia de las ondas electromagnéticas, calculando de manera aproximada su longitud de onda λ , con error inferior al 4 %.
- * El modulo motorizado infraROSSI y su software de control es una herramienta fácil de usar y muy precisa, capaz de ser utilizada para diversos propósitos en el aula de clase como modelo pedagógico, tanto de profesionales como estudiantes de diferentes ramas de la ciencia.

Referencias

- Albella, J. M., y Martínez-Duart, J. M. (1996). *Fundamentos de electrónica física y microelectrónica*. Addison-Wesley Iberoamericana.
- Hecht, E., Dal Col, R., Talavera, R. W., y Pérez, J. M. G. (2000). *Óptica*. Addison Wesley.
- Shalimova, K. V. y Grdiam, A. (1975). *Física de los semiconductores*.

³Enlace a la hoja de datos del diodo led emisor infrarrojo suministrada por el fabricante online http://www.betlux.com/product/LED_lamp/infrared%20diodo/BL-L314IR.PDF

APÉNDICE C. ILUSTRACIÓN DE LA LEY DE DECAIMIENTO DE LA DENSIDAD DE FLUJO DE RADIACIÓN CON EL CUADRADO INVERSO DE LA DISTANCIA DE SEPARACIÓN A LA FUENTE

Apéndice C. Ilustración de la ley de decaimiento de la densidad de flujo de radiación con el cuadrado inverso de la distancia de separación a la fuente



Ilustración de la ley de decaimiento de la densidad de flujo de radiación con el cuadrado inverso de la distancia de separación a la fuente; utilizando el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**

Diego Alberto Parra Garzón

diegoestudianteud1@gmail.com,

PhD. Julian Andrés Salamanca Bernal
jasalamanca@udistrital.edu.co

Universidad Distrital “Francisco José de Caldas”, Facultad de Ciencias y Educación
calle 3 No 26A-40 Bogotá-Colombia
Grupo de Física e Informática “FISINFOR”

Abstrac

The present article illustrates the attenuation due to the inverse square law of the flux of radiation emitted by a LED diode infrared emitter, using a mirror, the motor module **infraROSSI** and control software **FREEinfraROSSI**; environment GNU-linux, highlighting the functionality of this instrument illustration attenuation because to the irradiancia in point source is proportional to the inverse of the distance squared.

Keywords: Motor module, infrared sensors, microcontroller module bluetooth, electromagnetic wave, attenuation, inverse square law.

Resumen

El presente trabajo ilustra la atenuación debido a la ley del cuadrado inverso de la densidad de flujo de radiación emitida por un diodo led emisor infrarrojo, utilizando un espejo, el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**; en un entorno GNU-linux, resaltando la funcionalidad de este instrumento en la ilustración de la atenuación debido a que la irradiancia de una fuente puntual es proporcional al inverso de su distancia al cuadrado si no se consideran otros tipos de atenuación.

Palabras clave: Modulo motorizado, sensores infrarrojos, microcontrolador, modulo bluetooth, ondas electromagnéticas, atenuación, ley del cuadrado inverso, quantum.

Índice

INTRODUCCION	2
Marco teórico	2
Montaje experimental	5
Materiales del montaje	5
Montaje	5
Análisis de resultados	6
Conclusiones	7
Referencias	8

Índice de figuras

1. Transiciones con radiación entre una banda y los estados de impureza.	3
2. Relación irradianza vs voltaje.	4
3. Montaje modulo motorizado infraROSSI para el laboratorio de atenuación.	5
4. Esquema de pines del sistema de control de hardware.	5
5. Esquema de pines del sistema de control de avance.	5
6. Esquema de pines del sensor de electro-recepción activa en infrarrojo.	6

MARCO TEÓRICO

7.	Imagen del montaje para la atenuación utilizando el modulo motorizado infraROSSI y un espejo.	6
8.	Interfaz grafica software FREEinfraROSSI.	6
9.	Gráfica generada por el software FREEinfraROSSI.	7

Índice de tablas

1.	Ánalisis del experimento de difracción con el vehículo motorizado infraROSSI y su software de control FREEinfraROSSI.	7
----	---	---

INTRODUCCION

Los fenómenos de las ondas siempre han fascinado los pensamientos de los seres humanos; es allí donde la física con ayuda de la matemática muestran su majestuosidad, al explicar de manera muy detallada estos fenómenos de transporte; la atenuación es una de estas propiedades, la cual está muy presente en la vida diaria y con la ayuda del modulo motorizado infraROSSI y su software de control FREEinfraROSSI, se ilustra este fenómeno físico y se realiza un calculo del valor exponencial de la distancia a la cual se atenúa la irradiancia propia producida por un diodo infrarrojo emisor.

Según (Miáishev, 1995) “una fuente de luz casi siempre ilumina la superficie de los objetos desigualmente. Así, una lámpara suspendida sobre una mesa ilumina mejor el centro de ésta. Los bordes de la mesa están mucho menos iluminados. Esto no sólo se debe a que la intensidad de la luz de la lámpara eléctrica sea distinta en diferentes direcciones. Incluso en el caso un foco puntual corresponderá a la superficie del centro más potencia luminosa (flujo luminoso) que a una superficie igual en el borde.”

Se llama iluminación (o iluminancia) a la razón del flujo luminoso, que incide sobre una superficie determinada, al área de dicha superficie. Como unidad de iluminación se toma el lux (lx); un lx es la iluminación con la cual sobre 1 metro cuadrado de superficie se distribuye uniformemente un flujo luminoso de un lumen.

La dependencia de la iluminacion respecto de la distancia a la fuente se puede determinar colocando mentalmente una fuente puntual en el centro de una esfera, la iluminación sera igual al flujo luminoso total de la fuente distribuido sobre el área de la esfera. Es decir, la iluminación de una superficie es inversamente proporcional al cuadrado de su distancia a la fuente.”

Marco teórico

Los fotones producidos por el diodo emisor de la fuente emisora de radiación infrarroja se dan según (Shalimova y Grdiam, 1975) “debido a que en los semiconductores, con estructura compleja de las bandas energéticas, son posibles las transiciones indirectas de los electrones de la banda de conducción a la de valencia acompañadas de la emisión de un fotón. En este caso la recombinación del electrón libre con el hueco se desarrolla con la aparición de un fonón, lo que asegura la conservación del cuasi impulso. Lo más probable es que el fonón sea emitido. Si en el semiconductor se desarrollan procesos de recombinación entre bandas tanto directa como indirectas, en el espectro de radiación se observan dos bandas de luminiscencia.

En la banda prohibida de los semiconductores reales existe una gran cantidad de estados localizados, que están ligados a los átomos de impureza, defectos de la estructura, infracciones de la periodicidad de la estructura en la superficie, entre otros. Estos estados localizados desempeñan un papel importante en los procesos de luminiscencia.

Las transiciones de los electrones de la banda de conducción a los niveles de los pequeños donadores (o de los huecos de la banda de valencia a los niveles de los pequeños aceptores), que hacen que estos últimos se neutralicen, pueden ser con radiación. En este caso es de esperar la aparición de luminosidad en la región infrarroja remota del espectro. Pero los cálculos muestran que en estas transiciones lo más probable es que sea emitido un fonón y no un fotón, es decir, que el proceso se desarrolle sin radiación. La recombinación con radiación se produce por lo general como viene mostrado en la figura 1. Primero un electrón de la banda de conducción es capturado por un nivel local situado un poco más abajo que Ec¹, y después se efectúa la recombinación de este electrón localizado con un hueco de la banda de valencia, la cual va acompañada de la emisión de un fotón. El electrón puede también realizar una transición con radiación de la banda de conducción y después recombinarse con un hueco.

El estudio de los espectros de luminiscencia relacionados a diversas impurezas y defectos permite obtener información sobre estas infracciones de la estructura.

En los semiconductores se pueden producir pares electrón-hueco ligados por la atracción coulombiana, es decir, excitones. Si uno de estos pares se aniquila, se produce la emisión de un fotón. La energía de esta radiación es:

$$hv = E_g - E, \quad (1)$$

¹Ec es el nivel de energía de conducción.

MARCO TEÓRICO

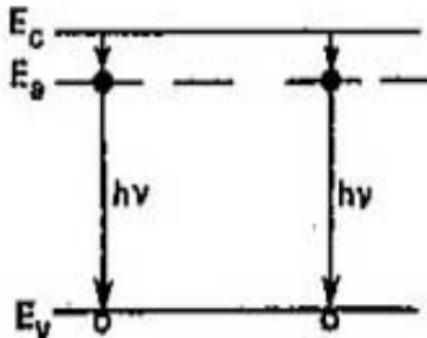


Figura 1. Transiciones con radiación entre una banda y los estados de impureza.

donde E es la energía de enlace del excitón."

Ahora se tiene un flujo de fotones de energía $h\nu$ saliendo del diodo emisor infrarrojo, según (Gautreau, Savin, y Velazquez Valle, 2001) "como los fotones viajan a la velocidad de la luz deben, de acuerdo con la teoría de la relatividad, tener una masa en reposo igual a cero; de aquí que su energía sea completamente cinética. Si un fotón existe, entonces se mueve a la velocidad de la luz, c ; si deja de moverse a velocidad c , deja de existir. Para $m_0 = 0$ la relación relativista momentum–energía se convierte en $E = pc$. de esta forma, cada fotón tiene un momentum de

$$p = \frac{E}{c} = \frac{h\nu}{c} = \frac{h}{\lambda}, \quad (2)$$

Desde el punto de vista cuántico, un haz de energía electromagnética se compone de fotones que se desplazan a la velocidad c . La intensidad del haz será proporcional al número de fotones que cruza un área unitaria por unidad de tiempo. Entonces, si el haz es monocromático (de una frecuencia), la intensidad I se obtendrá de

$$I = (h\nu) \times \left(\frac{N}{A \times t} \right), \quad (3)$$

h es la constante de Plank que tiene un valor de $6,626 \times 10^{-34}(J \cdot s)$; N es el número de fotones que pasan por segundo a través de la superficie; A es la superficie; t es el tiempo en segundos."

Esta radianza de fotones desde el diodo emisor, avanza por el espacio proyectando un ángulo sólido hasta un espejo que se encuentra a 30 cm de la fuente, por lo que la irradiancia sera igual al cociente de la radianza con el ángulo sólido proyectado sobre el espejo.

Ahora el flujo de fotones que interactúa con la superficie del material de muestra dan como resultado desde el punto de vista clásico varios fenómenos físicos como lo son la reflexión, refracción, absorción, atenuación, reflectancia, transmitancia. Ahora el flujo de fotones es visto como el flujo de ondas electromagnéticas, las que interactúan con la materia en donde parte de la energía es transmitida al material aumentando la energía cinética media de sus constituyentes, otra parte traspasa el material y el resto del flujo electromagnético avanza paralelamente contrario a la dirección de desplazamiento inicial, si no se considera interacción de las ondas electromagnéticas con el aire como medio disipativo y otras formas de perdida de energía.

En este trabajo solo se tiene en cuenta el fenómeno de reflexión de la luz con el espejo, en el caso ideal que no exista perdida de energía entre la luz incidente en el espejo; osea que la intensidad del rayo incidente sea igual a la intensidad de la luz emitida por la interfaz del espejo. Solo se tiene en cuenta la atenuación de energía debido al avance de la expansión geométrica por el espacio, debido al flujo de luz emitida por el diodo emisor infrarrojo.

Ahora la luz que sale del espejo tendrá que recorrer nuevamente 30 cm hasta el detector, mientras avanza seguirá decayendo su intensidad con el inverso del cuadrado de la distancia.

Al iluminar el diodo receptor infrarrojo con esta energía radiante, según (Albella y Martínez-Duart, 1996) "en el semiconductor por cada fotón absorbido se rompe un enlace y se crea un par electrón-hueco. Es importante destacar que no todos los portadores fotogenerados contribuyen a la conducción, ya que una fracción importante de ellos se recombinan antes de llegar al extremo correspondiente del semiconductor. El calculo del incremento de corriente ΔI_e , debida al exceso de electrones generados en la banda de conducción, Δn , es

$$\Delta I_e = q\mu_e(\Delta n)ES, \quad (4)$$

siendo E el campo eléctrico aplicado, μ_e la movilidad de los electrones y S la sección transversal del fotoconductor.

En condiciones de iluminación, el estado estacionario se alcanza cuando la velocidad de generación de portadores en todo el volumen del semiconductor, G , se iguala a la velocidad de recombinación, R , es decir $R = G$. para un conductor intrínseco en el cual existe un exceso de portadores, $\Delta n = \Delta p$, la velocidad de recombinación de los portadores vendrá dada por:

$$R = \frac{\Delta n}{\tau} = \frac{\Delta p}{\tau}, \quad (5)$$

MARCO TEÓRICO

siendo τ el tiempo de vida media de los portadores fotogenerados. En un semiconductor de longitud L en el que suponemos que el espesor es suficiente para que toda la luz que incide sobre él, sea absorbida en su interior, se tiene ahora para la velocidad de generación de portadores en la banda de conducción:

$$G = \eta n_{fot} = \eta \frac{P_i}{SL}, \quad (6)$$

siendo n_{fot} el número de fotones incidentes en el semiconductor por unidad de volumen y de tiempo, y η la eficiencia de la conversión en la generación de portadores. El valor n_{fot} se calcula a través del cociente entre la potencia de la luz incidente, P_i , y la energía de la radiación, $h\nu$, dividido a su vez por el volumen del material.

Sabiendo que la velocidad de arrastre de los electrones por el campo eléctrico viene dada por: $v_e = \mu_e E$, las igualdades anteriores permiten escribir para la corriente de electrones fotogenerada entre los dos electrodos:

$$\Delta I_e = qv_e \eta \frac{P_i}{L} \tau, \quad (7)$$

si se tiene en cuenta que el cociente $t_r = L/v_e$, representa el tiempo de tránsito de los electrones entre los dos electrodos, resulta para ΔI_e :

$$\Delta I_e = q\eta \frac{P_i}{h\nu} \frac{\tau}{t_r}, \quad (8)$$

con una expresión similar para la corriente de huecos en la banda de valencia. En la ecuación 8, el factor $q\eta(P_i/h\nu) = I_{fot}$ tiene dimensiones de corriente y representa la velocidad de generación de carga en el semiconductor. En función de este parámetro, se define el factor de ganancia del fotoconductor a través del cociente:

$$\frac{\Delta I}{I_{fot}} = \frac{\tau}{t_r}, \quad (9)$$

ahora bien un diodo operando con cierto voltaje aplicado, V , en presencia de radiación electromagnética capaz de excitar portadores a través de la banda prohibida dejara pasar una intensidad I dada por:

$$I = I_0 [e^{(qV/kT)} - 1] - I_L, \quad (10)$$

donde I_0 representa la corriente típica de un diodo, I_L representa la corriente debida a los portadores generados. El valor de I_L puede calcularse de la siguiente manera:

$$I_L = qGS(L_e - L_h), \quad (11)$$

siendo G el número de portadores generados por unidad de volumen y de tiempo y S el área de la sección transversal del diodo. L_e y L_h representan las longitudes de difusión de los electrones y huecos. El dispositivo funciona entonces como detector del nivel de iluminación convirtiendo una señal óptica en señal eléctrica."

Como la corriente I en el diodo es proporcional a la irradiancia de la superficie iluminada por fotones infrarrojos y esta ultima se atenúa con el inverso del cuadrado de la distancia del frente de energía a la fuente; en otras palabras como el voltaje que mide el microcontrolador es directamente proporcional a la corriente I en el diodo receptor infrarrojo, por consiguiente el voltaje medido en el semiconductor debe tener una relación inversamente proporcional al cuadrado de la distancia.

$$V(x) = \frac{A_0}{x^2}, \quad (12)$$

Como la irradiancia de fotones infrarrojos es proporcional al inverso del cuadrado de la distancia de separación a la fuente, y la función voltaje que mide el modulo motorizado **infraROSSI** es proporcional al inverso del cuadrado de la distancia de separación de la fuente al vehículo **infraROSSI**; se realiza el cociente de la intensidad irradiada en el diodo led que es según el fabricante de $20 \text{ mW}/\Omega$, sobre el voltaje producido en el sensor que es de 5000 mV ; como se muestra en la figura 2.

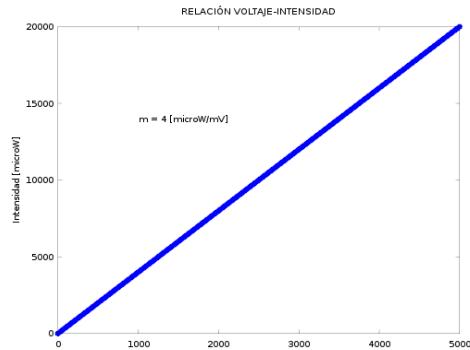


Figura 2. Relación irradiancia vs voltaje.

Una vez absorbidos los fotones infrarrojos, el microcontrolador atmega 328 del vehículo motorizado **infraROSSI**, mide la relación de voltaje en el sensor de electro-recepción activa que es proporcional a la irradiancia del emisor infrarrojo la cual equivale a $4\mu\text{W}/\text{mV}$ lo que significa que por cada $4\mu\text{W}$ de radiación incidente el sensor

MONTAJE EXPERIMENTAL



Figura 3. Montaje modulo motorizado infraROSSI para el laboratorio de atenuación.

esta arrojando en su salida 1mV.

$$I(x) = \frac{B_0}{x^2}, \quad (13)$$

Montaje experimental

Materiales del montaje

Para la realización de este montaje se requiere de los siguientes materiales:

- Ordenador con sistema operativo GNU-Linux.
- Modulo motorizado **infraROSSI**.
- Software de control **FREEinfraROSSI**.
- Espejo de 10x20 cm².
- Modulo bluetooth para pc.

Montaje

Colocar el sensor de electro-recepción activa junto con el control de avance y el control de hardware en el vehículo motorizado **infraROSSI** como se observa en la figura 3.

Los esquemas de pines de las partes utilizadas para el montaje se muestran en las figuras 4 , 5 y 6 ; se debe conectar el pin ch1 con el pin ea4, el pin ch3 con el pin ca9, el pin de ch4 con el pin ea2, el pin ch7 con el pin ca2, el pin ch8 con el pin ca1, el pin ca5 con el pin ea1, el pin ca6 con el pin ea3, por ultimo el pin ca7 y el pin ca8 son las conexiones al motor.

Situar el vehículo motorizado **infraROSSI** a una distancia de 30 cm del espejo como se muestra en la figura 7; el espejo debe estar perpendicular a la parte frontal del vehículo. Abrir

Montaje

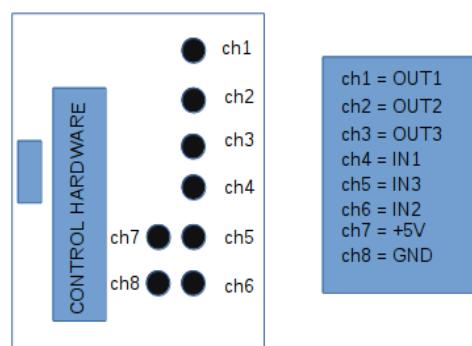


Figura 4. Esquema de pines del sistema de control de hardware.

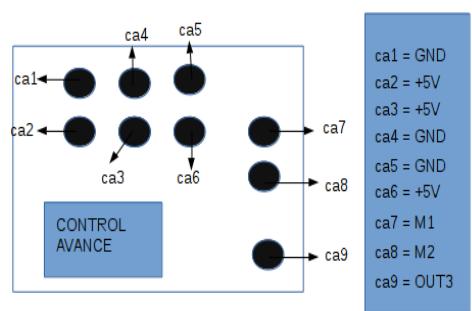


Figura 5. Esquema de pines del sistema de control de avance.

una terminal de GNU-Linux y escribir infrarossi, oprimir enter y la clave de superusuario, luego de abrir el programa como se muestra en la figura 8, debe oprimir el botón on, esperar que se empareje el bluetooth, una vez emparejado el modulo bluetooth, el programa desplegará un tercer menú, oprimir el botón de atenuación y esperar que el programa tome los datos necesarios.

Ahora el software de control **FREEinfraROSSI** le envía vía bluetooth la señal de avanzar y capturar datos al vehículo motorizado **infraROSSI** el cual avanza 2 milímetros por cada paso, recolecta 140 datos por cada avance enviándolos vía bluetooth al ordenador en donde el software de control realiza un análisis estadístico de los mismos, cuando termina este análisis envía una señal al módulo motorizado **infraROSSI** vía bluetooth, indicándole que avance nuevamente y repita el proceso, esto lo realiza 117 veces; una vez terminado de recoger todos los datos

ANÁLISIS DE RESULTADOS

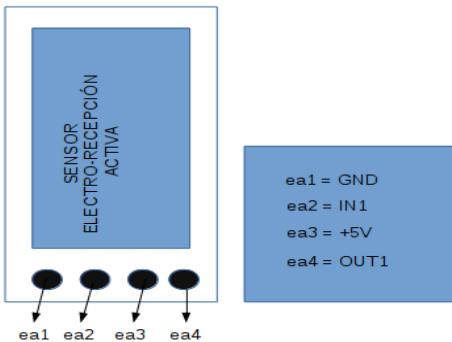


Figura 6. Esquema de pines del sensor de electro-recepción activa en infrarrojo.



Figura 7. Imagen del montaje para la atenuación utilizando el módulo motorizado infraROSSI y un espejo.

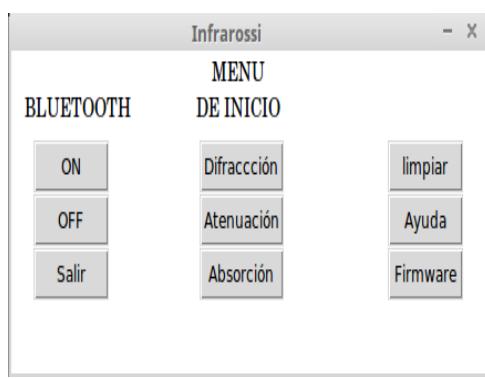


Figura 8. Interfaz grafica software FREEinfraROSSI.

realiza una estadística sobre toda la muestra de estos datos capturados.

Análisis de resultados

El programa FREEinfraROSSI después de recoger los datos realiza un análisis estadístico de los mismos, según (Ostle, 1981) “un ajuste lineal de una función exponencial de la forma $Y^* = aX^b$, siendo Y la intensidad irradiada, a la amplitud, X la distancia de avance, b el valor del exponente de atenuación, Y^* es el valor esperado de la variable física; aplicando logaritmo natural para linealizar se obtiene:

$$\ln(Y^*) = \ln(a) + b\ln(X) \dots \Rightarrow V^* = A + bU, \quad (14)$$

donde V^* es $\ln(Y^*)$, A es el $\ln(a)$ y U es igual al $\ln(X)$.

La suma de todos los errores debe ser diferentes a cero $\sum e \neq 0$.

El exponente de ajuste b se halla con la varianza de U sobre V dividida entre la varianza de U sobre U , obteniendo

$$b = \frac{S_{UV}}{S_U^2} = \frac{\frac{1}{n} \sum_{i=1}^n UV - \bar{U}\bar{V}}{\frac{1}{n} \sum_{i=1}^n U^2 - \bar{U}^2}, \quad (15)$$

ahora el valor de A sera igual al valor medio del logaritmo de la variable física \bar{V} menos el valor de la multiplicación entre el exponente b y el valor medio del logaritmo de la variable experimental:

$$A = \bar{V} - b\bar{U}, \quad (16)$$

deshaciendo el logaritmo de A se obtiene el valor de la amplitud a :

$$a = \text{antiln}(A) = \text{antiln}(\bar{V} - b\bar{U}), \quad (17)$$

de modo que el ajuste efectuado es:

$$Y^* = aX^b = [\text{antiln}(\bar{V} - b\bar{U})] * X^{\left(\frac{\frac{1}{n} \sum_{i=1}^n UV - \bar{U}\bar{V}}{\frac{1}{n} \sum_{i=1}^n U^2 - \bar{U}^2}\right)}, \quad (18)$$

la bondad del ajuste es el error cuadrático medio o ECM y es igual a:

$$ECM = \frac{\sum_{i=1}^n e_i^2}{n}, \quad (19)$$

siendo e_i cada una de las diferencias entre las variables físicas y los valores estimados para las variables físicas $e_i = Y_i - Y_i^*$; al haber transformado la variable física ya no se minimiza $\sum e^2$ sino $\sum (\ln(Y) - \ln(Y^*))^2$, de ahí que $\sum e \neq 0$.

Luego de este análisis estadístico, aparece la gráfica de estos, la cual se aprecia en la figura 9, esta ya contiene el análisis estadístico y arroja el valor del exponente que debe

CONCLUSIONES

Tabla 1

Análisis del experimento de difracción con el vehículo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**.

P	a ($\times 10^{-6}$)	b	T	E.C.M. ($\times 10^{-7}$)	E.E.
1	251	-1,922	-2	1.479	0.078
2	251	-1,922	-2	1.479	0.078
3	275	-1,874	-2	26.257	0.126
4	251	-1,922	-2	1.479	0.078
5	251	-1,922	-2	1.479	0.078
6	275	-1,874	-2	26.257	0.126

Nota. Datos obtenidos de seis pruebas para medir el exponente de atenuación, con el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**; a es la amplitud de la función, b es el exponente con el que se atenúa la función, *E.C.M.* es el error cuadrático medio, o también llamado bondad en el ajuste, *E.E.* es el error en el exponente.

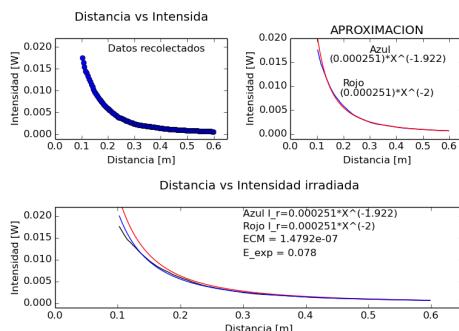


Figura 9. Gráfica generada por el software **FREEinfraROSSI**.

tener la distancia; los datos del experimento de atenuación están en puntos azules, la gráfica estimada estadísticamente esta en líneas azules, la gráfica teórica son las líneas de color rojo.

La gráfica de análisis y los datos capturados se almacenan dentro del archivo llamado Carpetas/Atenuacion con la fecha y hora del análisis de datos.

En la tabla 1, se muestra los resultados obtenidos después de seis pruebas del exponente de atenuación con el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**; en ella se observa que la amplitud de la función intensidad, tiene un valor medio de 0,000259, el valor medio del factor de atenuación de la distancia es -1,906; el valor obtenido en el análisis estadístico esta

muy próximo al valor teórico; por lo que la cantidad de energía radiada por el diodo emisor decrece con el inverso del cuadrado de la distancia con un error de 4,7 %.

Por tanto la intensidad en función de la distancia en este trabajo tiene una ecuación estimada en la ecuación 20 :

$$I(x) = 0,000259X^{-1,906}. \quad (20)$$

Conclusiones

- * La amplitud del valor medio de la función intensidad es 0,000259 [W][m²], por lo que se considera este valor, como la amplitud ideal para la función intensidad.
- * El valor medio en el factor de atenuación de la distancia es -1,906 que esta muy cercano al predicho por la teoría el cual es -2, con un error en su estimación de 4,7 %, lo cual indica que el decaimiento en la densidad de flujo electromagnético que atraviesa una superficie en la unidad de tiempo, que a su vez se refleja y por ende que llega al sistema detector del modulo motorizado **infraROSSI**, decrece con el cuadrado de la distancia.
- * El promedio del error cuadrático medio en los datos obtenidos por el experimentador indican que 11,286x10⁻⁵ % de estos no pueden ser explicados por este modelo físico-matemático, dejando solamente un 99,9998 % de fiabilidad en el ajuste estadístico, utilizado en este trabajo.
- * El modulo motorizado **infraROSSI** y su software de control ilustran de manera cuantitativa y cualitativa fenómenos ondulatorios y corpusculares de la radiación electromagnética como la atenuación con el inverso del cuadrado de la distancia, la radiansa, la irradiansa, fenómenos de transporte e inyección y su análisis estadístico, calculando de una manera aproximada el exponente que acompaña a la atenuación debido a la distancia de propagación del flujo de energía radiante producida en el diodo emisor infrarrojo.
- * El modulo motorizado **infraROSSI** y su software de control es una herramienta fácil de usar y muy precisa, capaz de ser utilizada para diversos propósitos en el aula de clase combino modelo pedagógico, tanto de profesionales como estudiantes de diferentes carreras; haciendo de este recurso pedagógico una herramienta indispensable en el aula de clase.

REFERENCIAS

- Referencias
- Albella, J. M., y Martínez-Duart, J. M. (1996). *Fundamentos de electrónica física y microelectrónica*. Addison-Wesley Iberoamericana.
- Gautreau, R., Savin, W., y Velazquez Valle, D. (2001). Física moderna.
- Miákishev, G. (1995). *Bujovsev. física 4. editorial mir moscú. moscú. 198658. frumento a. biofísica.* Mosby/Doyma Libros. Madrid.
- Ostle, B. (1981). *Estadística aplicada*. Limusa.
- Shalímonova, K. V., y Grdiam, A. (1975). *Física de los semiconductores*.

**APÉNDICE D. CALCULO DEL FACTOR DE PERDIDA DE LA DENSIDAD DE FLUJO DE RADIACIÓN EMITIDA
POR UN DIODO LED EMISOR INFRARROJO, CUANDO ESTA RADIACIÓN INTERACTÚA CON LA MATERÍA**

**Apéndice D. Calculo del factor de perdida de la densidad
de flujo de radiación emitida por un diodo led emisor
infrarrojo, cuando esta radiación interactúa con la
matería**



Calculo del factor de perdida de la densidad de flujo de radiación emitida por un diodo led emisor infrarrojo, cuando esta radiación interactúa con la materia; utilizando el modulo motorizado infraROSSI y su software de control FREEinfraROSSI

Diego Alberto Parra Garzón
diegoestudianteud1@gmail.com,
PhD. Julian Andrés Salamanca Bernal
jasalamanca@udistrital.edu.co

Universidad Distrital “Francisco José de Caldas”, Facultad de Ciencias y Educación
calle 3 No 26A-40 Bogotá-Colombia
Grupo de Física e Informática “FISINFOR”

Abstrac

This paper illustrates the calculation of the loss factor flux density of radiation emitted by a LED diode infrared emitter, when this radiation interacts with matter; using construction paper of various colors, the motor module **infraROSSI** and control software **FREEinfraROSSI**, in a GNU-Linux environment; highlighting the functionality of this instrument in illustrating the calculation factor in the loss of intensity on the measuring instrument and its proportionality with the flux of radiation emitted by a LED diode emitting infrared radiation when it interacts with matter.

Keywords: Motor module, infrared sensors, microcontroller module bluetooth, electromagnetic wave, reflection, refraction, absorption, attenuation, reflectance, transmittance.

Resumen

El presente trabajo ilustra el calculo del factor de perdida de la densidad de flujo de radiación emitida por un diodo led emisor infrarrojo, cuando esta radiación interactúa con la metería; utilizando cartulinas de varios colores, el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**, en un entorno GNU-linux; resaltando la funcionalidad de este instrumento en la ilustración del calculo en el factor de perdida de la intensidad en el instrumento de medida y su proporcionalidad con la densidad de flujo de radiación emitida por un diodo led emisor infrarrojo, cuando esta radiación interactúa con la metería.

Palabras clave: Modulo motorizado, sensores infrarrojos, microcontrolador, reflectancia, refracción, modulo bluetooth, ondas electromagnéticas, reflexión, absorción, atenuación, transmitancia.

	Índice	Conclusiones	9
INTRODUCCION	2	Referencias	9
Marco teórico	2	Índice de figuras	
Montaje experimental	5	1. Transiciones con radiación entre una banda y los estados de impureza.	2
Materiales del montaje	5	2. Montaje laboratorio absorción utilizando el modulo motorizado infraROSSI y un octavo de cartulina.	3
Montaje	6	3. Reflexión y transmisión de un haz incidente. . .	4
Análisis de resultados	6		

MARCO TEÓRICO

4.	Relación irradiancia vs voltaje.	5
5.	Montaje modulo motorizado infraROSSI para el laboratorio de absorción.	6
6.	Esquema de pines del sistema de control de hardware.	6
7.	Esquema de pines del sistema de control de avance.	6
8.	Esquema de pines del sensor de electro-recepción activa en infrarrojo.	7
9.	Interfaz grafica software FREEinfraROSSI.	7
11.	Imagen generada con la librería matplotlib de python 2.7, donde se aprecia las funciones estimadas por el programa y la función ideal de intensidad.	8
10.	Imagen generada por el programa FREEinfraROSSI.	8
12.	Imagen generada con la librería matplotlib de python 2.7, donde se aprecia una porción distingible de las funciones estimadas por el programa.	8

Índice de tablas

1.	Ánalisis del experimento de absorción con el vehículo motorizado infraROSSI y su software de control FREEinfraROSSI.	8
----	---	---

INTRODUCCION

Los fenómenos de las ondas siempre han fascinado los pensamientos de los seres humanos; es allí donde la física con ayuda de la matemática muestran su majestuosidad, al explicar de manera muy detallada estos fenómenos de transporte; desde el punto de vista clásico en la interacción del flujo de radiación electromagnética con la materia pueden darse varios fenómenos tales como la reflexión, refracción, absorción, atenuación, reflectancia, transmitancia; en estos fenómenos, incide radiación sobre una superficie de muestra y se mide la proporción en la cantidad de radiación que el material absorbe, que deja pasar y la que se refleja, estos cálculos son muy utilizados en espectrofotómetros infrarrojos y en diversos estudios que implican la interacción radiación-materia.

En este trabajo se hace un calculo del índice de reflectancia de las diferentes muestras, enviando radiación infrarroja sobre el material y midiendo el factor de perdida en la intensidad de la radiación electromagnética que incide sobre el detector infrarrojo del vehículo motorizado **infraROSSI** después que la radiación ha interactuado con la muestra y ha sido reflejada.

Marco teórico

Los fotones producidos por el diodo emisor de la fuente emisora de radiación infrarroja, según (Shalímová y Grdiam, 1975) “se dan debido a que en los semiconductores, con estructura compleja de las bandas energéticas, son posibles las transiciones indirectas de los electrones de la banda de conducción a la de valencia acompañadas de la emisión de un fotón. En este caso la recombinación del electrón libre con el hueco se desarrolla con la aparición de un fonón, lo que asegura la conservación del cuasi impulso. Lo más probable es que el fonón sea emitido. Si en el semiconductor se desarrollan procesos de recombinación entre bandas tanto directa como indirectas, en el espectro de radiación se observan dos bandas de luminiscencia.

Las transiciones de los electrones de la banda de conducción a los niveles de los pequeños donadores (o de los huecos de la banda de valencia a los niveles de los pequeños aceptores), que hacen que estos últimos se neutralicen, pueden ser con radiación. En este caso es de esperar la aparición de luminosidad en la región infrarroja remota del espectro.

Pero los cálculos muestran que en estas transiciones lo más probable es que sea emitido un fonón y no un fotón, es decir, que el proceso se desarrolle sin radiación. La recombinación con radiación se produce por lo general como viene mostrado en la figura 1.

En la banda prohibida de los semiconductores reales existe una gran cantidad de estados localizados, que están ligados a los átomos de impureza, defectos de la estructura, infracciones de la periodicidad de la estructura en la superficie, entre otros. Estos estados localizados desempeñan un papel importante en los procesos de luminiscencia. El

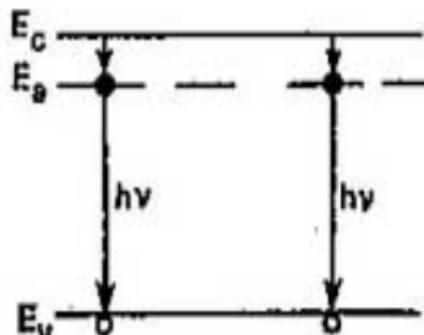


Figura 1. Transiciones con radiación entre una banda y los estados de impureza.

MARCO TEÓRICO

estudio de los espectros de luminescencia relacionados a diversas impurezas y defectos permite obtener información sobre estas infracciones de la estructura.

En los semiconductores se pueden producir pares electrón-hueco ligados por la atracción coulombiana, es decir, excitones. Si uno de estos pares se aniquila, se produce la emisión de un fotón. La energía de esta radiación es:

$$hv = E_g - E, \quad (1)$$

donde E es la energía de enlace del excitón."

Ahora se tiene un flujo de fotones de energía hv saliendo del diodo emisor infrarrojo, según (Gautreau, Savin, y Velazquez Valle, 2001) "como los fotones viajan a la velocidad de la luz deben, de acuerdo con la teoría de la relatividad, tener una masa en reposo igual a cero; de aquí que su energía sea completamente cinética. Si un fotón existe, entonces se mueve a la velocidad de la luz, c ; si deja de moverse a velocidad c , deja de existir. Para $m_0 = 0$ la relación relativista momentum-energía se convierte en $E = pc$. de esta forma, cada fotón tiene un momentum de

$$p = \frac{E}{c} = \frac{hv}{c} = \frac{h}{\lambda}, \quad (2)$$

Desde el punto de vista cuántico, un haz de energía electromagnética se compone de fotones que se desplazan a la velocidad c . La intensidad del haz será proporcional al número de fotones que cruza un área unitaria por unidad de tiempo. Entonces, si el haz es monocromático (de una frecuencia), la intensidad I se obtendrá de

$$I = (hv) \times \left(\frac{N}{A \times t} \right), \quad (3)$$

h es la constante de Plank que tiene un valor de $6,626 \times 10^{-34}(J * s)$; N es el número de fotones que pasan por segundo a través de la superficie; A es la superficie; t es el tiempo en segundos."

Esta radianza de fotones desde el diodo emisor, avanza por el espacio proyectando un ángulo sólido, por lo que la irradiancia será igual al cociente de la radianza con el ángulo sólido proyectado.

La máxima distancia que se toma para la radianza de fotones como se observa en la figura 2, es de 30 centímetros desde el diodo emisor hasta el material de muestra¹ colocado perpendicular al flujo de energía, por lo que habrá reducido su intensidad con el inverso del cuadrado de la distancia de la fuente a la muestra $\frac{1}{(0.30)^2}$.

Ahora el flujo de fotones que interactúa con la superficie

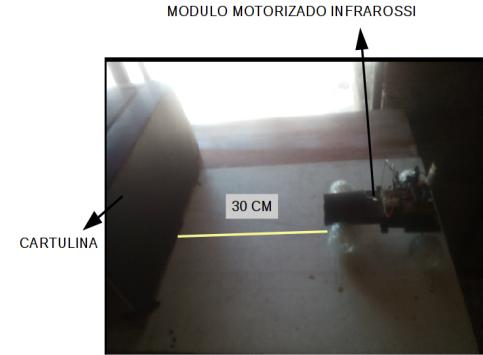


Figura 2. Montaje laboratorio absorción utilizando el modulo motorizado infraROSSI y un octavo de cartulina.

del material de muestra dan como resultado desde el punto de vista clásico varios fenómenos físicos como lo son la reflexión, refracción, absorción, atenuación, reflectancia, transmitancia. Ahora el flujo de fotones es visto como el flujo de ondas electromagnéticas, las que interactúan con la materia en donde parte de la energía es transmitida al material aumentando la energía cinética media de sus constituyentes, otra parte traspasa el material y el resto del flujo electromagnético avanza paralelamente contrario a la dirección de desplazamiento inicial, si no se considera interacción de las ondas electromagnéticas con el aire como medio disipativo y otras formas de perdida de energía.

Según (Hecht, Dal Col, Talavera, y Pérez, 2000) "Consideremos un haz de luz circular que incide en una superficie, tal como se muestra en la figura 3, de tal modo que se produzca una zona iluminada cuya área sea A . Recordemos que la potencia por unidad de área que cruza una superficie en el vacío cuya normal es paralela a \mathbf{S} , el vector de Poynting viene determinado por:

$$\mathbf{S} = c^2 \epsilon_0 \mathbf{E} \times \mathbf{B}. \quad (4)$$

A demás la densidad de flujo radiante (W/m^2) o irradiancia es entonces

$$I = < S >_t = \frac{c \epsilon_0}{2} E_0. \quad (5)$$

Este el promedio de energía por unidad de tiempo que cruza un área unidad, normal a \mathbf{S} (en medios isótropos \mathbf{S} es paralela al vector de onda \mathbf{k}). En el caso que nos ocupa (figura 3) sean I_i , I_r y I_t las densidades de flujo incidente, reflejado y transmitido, serán respectivamente, $A \cos(\theta_i)$, $A \cos(\theta_r)$ y $A \cos(\theta_t)$.

¹La muestra son octavos de cartulina de colores.

MARCO TEÓRICO

De acuerdo con esto, la potencia incidente es $I_i \cos(\theta_i)$, esta es la energía por unidad de tiempo que fluye en el rayo incidente y, por consiguiente, la potencia que llega a la superficie de A . Del mismo modo, $I_r \cos(\theta_r)$, es la potencia en el rayo reflejado, e $I_t \cos(\theta_t)$, es la potencia que se transmite a través de A . Definimos la **reflectancia** R como el cociente entre la potencia (o flujo) reflejada y la potencia incidente, es decir:

$$R \equiv \frac{I_r \cos(\theta_r)}{I_i \cos(\theta_i)} = \frac{I_r}{I_i}, \quad (6)$$

Del mismo modo la **transmitancia** T se define como el

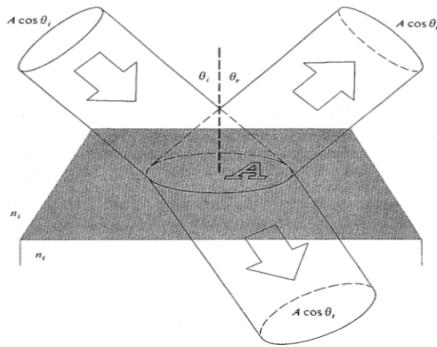


Figura 3. Reflexión y transmisión de un haz incidente.

cociente entre el flujo transmitido y el flujo incidente y viene dada por:

$$T \equiv \frac{I_t \cos(\theta_t)}{I_i \cos(\theta_i)} = \frac{I_t}{I_i}, \quad (7)$$

escribamos ahora una ecuación que represente la conservación de energía para la configuración que se muestra en la figura 3. Dicho de otro modo, la energía total que llega al área A por unidad de tiempo debe ser igual a la energía que fluye hacia fuera de ella por unidad de tiempo:

$$I_i \cos(\theta_i) = I_r \cos(\theta_r) + I_t \cos(\theta_t), \quad (8)$$

multiplicando a ambos lados por c esta expresión queda:

$$n_i E_{0i}^2 \cos(\theta_i) = n_r E_{0r}^2 \cos(\theta_r) + n_t E_{0t}^2 \cos(\theta_t)$$

Dividiendo a ambos lados por $n_i E_{0i}^2 \cos(\theta_i)$ obtenemos:

$$1 = \left(\frac{E_{0r}}{E_{0i}} \right)^2 + \left(\frac{n_t \cos(\theta_t)}{n_i \cos(\theta_i)} \right) \left(\frac{E_{0t}}{E_{0i}} \right)^2, \quad (9)$$

Pero esto es simplemente:

$$R + T = 1, \quad (10)$$

En donde n_i es el índice de refracción del medio del que incide, n_t es el índice de refracción en el medio transmitido."

Si se toma R como la reflectancia y T como el factor de perdida de energía por la muestra, podemos decir que:

$$I_{fot} = RI_{fot} + TI_{fot}$$

En donde I_{fot} es el flujo de fotones que inciden por unidad de tiempo sobre la superficie, RI_{fot} es el flujo de fotones que son reflejados por unidad de tiempo sobre la superficie y TI_{fot} es el flujo de fotones que se transmitieron sobre la superficie. De esta manera los fotones salen de la muestra con una radianza de RI_{fot} .

Esta radianza de fotones desde el punto de vista cuántico RI_{fot} , o frente de onda desde el punto de vista clásico $\left(\frac{E_{0r}}{E_{0i}} \right)^2$, tiene que avanzar ahora 30 centímetros desde la muestra hasta el detector, por lo que ahora disminuye su intensidad desde la muestra hasta el foto-detector con el inverso del cuadrado de la distancia $\frac{RI_{fot}}{(0.30)^2}$.

Al iluminar el diodo receptor infrarrojo con esta energía radiante, según (Albella y Martínez-Duart, 1996) "en el semiconductor por cada fotón absorbido se rompe un enlace y se crea un par electrón-hueco. Es importante destacar que no todos los portadores fotogenerados contribuyen a la conducción, ya que una fracción importante de ellos se recombinan antes de llegar al extremo correspondiente del semiconductor. El cálculo del incremento de corriente ΔI_e , debida al exceso de electrones generados en la banda de conducción, Δn , es

$$\Delta I_e = q\mu_e(\Delta n)ES \quad (11)$$

siendo E el campo eléctrico aplicado, μ_e la movilidad de los electrones y S la sección transversal del fotoconductor.

En condiciones de iluminación, el estado estacionario se alcanza cuando la velocidad de generación de portadores en todo el volumen del semiconductor, G , se iguala a la velocidad de recombinación, R , es decir $R = G$. para un conductor intrínseco en el cual existe un exceso de portadores, $\Delta n = \Delta p$, la velocidad de recombinación de los portadores vendrá dada por:

$$R = \frac{\Delta n}{\tau} = \frac{\Delta p}{\tau} \quad (12)$$

siendo τ el tiempo de vida media de los portadores fotogenerados. En un semiconductor de longitud L en el que suponemos que el espesor es suficiente para que toda la luz que incide sobre él, sea absorbida en su interior, se tiene ahora para la velocidad de generación de portadores en la banda de conducción:

$$G = \eta n_{fot} = \eta \frac{P_i}{SL} \quad (13)$$

siendo n_{fot} el número de fotones incidentes en el semiconductor por unidad de volumen y de tiempo, y η la eficiencia de la conversión en la generación de portadores.

MONTAJE EXPERIMENTAL

El valor n_{for} se calcula a través del cociente entre la potencia de la luz incidente, P_i , y la energía de la radiación, $h\nu$, dividido a su vez por el volumen del material.

Sabiendo que la velocidad de arrastre de los electrones por el campo eléctrico viene dada por: $v_e = \mu_e E$, las igualdades anteriores permiten escribir para la corriente de electrones fotogenerada entre los dos electrodos:

$$\Delta I_e = qv_e \eta \frac{P_i}{h\nu} \tau \quad (14)$$

si se tiene en cuenta que el cociente $t_r = L/v_e$, representa el tiempo de tránsito de los electrones entre los dos electrodos, resulta para ΔI :

$$\Delta I_e = q\eta \frac{P_i}{h\nu} \frac{\tau}{t_r} \quad (15)$$

con una expresión similar para la corriente de huecos en la banda de valencia. En la ecuación anterior, el factor $q\eta(P_i/h\nu) = I_{tot}$ tiene dimensiones de corriente y representa la velocidad de generación de carga en el semiconductor. En función de este parámetro, se define el factor de ganancia del fotoconductor a través del cociente:

$$\frac{\Delta I}{I_{tot}} = \frac{\tau}{t_r} \quad (16)$$

ahora bien un diodo operando con cierto voltaje aplicado, V , en presencia de radiación electromagnética capaz de excitar portadores a través de la banda prohibida dejará pasar una intensidad I dada por:

$$I = I_0 [e^{(qV/kT)} - 1] - I_L \quad (17)$$

donde I_0 representa la corriente típica de un diodo, I_L representa la corriente debida a los portadores generados. El valor de I_L puede calcularse de la siguiente manera:

$$I_L = qGS(L_e - L_h) \quad (18)$$

siendo G el número de portadores generados por unidad de volumen y de tiempo y S el área de la sección transversal del diodo. L_e y L_h representan las longitudes de difusión de los electrones y huecos. El dispositivo funciona entonces como detector del nivel de iluminación convirtiendo una señal óptica en señal eléctrica."

Como la corriente I en el diodo es proporcional a la irradiancia de la superficie iluminada por fotones infrarrojos y esta última se atenúa con el inverso del cuadrado de la distancia del frente de energía a la fuente; el voltaje que mide el microcontrolador es directamente proporcional a la corriente I en el diodo receptor infrarrojo y esta última es proporcional a la densidad de potencia media de iluminación infrarroja sobre este, por consiguiente el voltaje medido

en el semiconductor debe tener una relación inversamente proporcional al cuadrado de la distancia, para el caso ideal:

$$I_{tot}(x) \propto V(x) = \frac{A_0}{x^2} \quad (19)$$

para este caso en el que el material absorbe parte de esa energía, solo RI_{tot}/x^2 estarán incidiendo en la superficie del foto detector infrarrojo por lo que ahora el voltaje medido en el microcontrolador sera:

$$RI_{tot}(x) \propto RV(x) = \frac{RA_0}{x^2} \quad (20)$$

Como la irradiancia de fotones infrarrojos es proporcional al inverso del cuadrado de la distancia de separación a la fuente, y la función voltaje que mide el modulo motorizado **infraROSSI** es proporcional al inverso del cuadrado de la distancia de separación de la fuente al vehículo **infraROSSI**; se realiza el cociente de la intensidad irradiada en el diodo led que es según el fabricante de $20 \text{ mW}/\Omega$, sobre el voltaje producido en el sensor que es de 5000mV ; como se muestra en la figura 4.

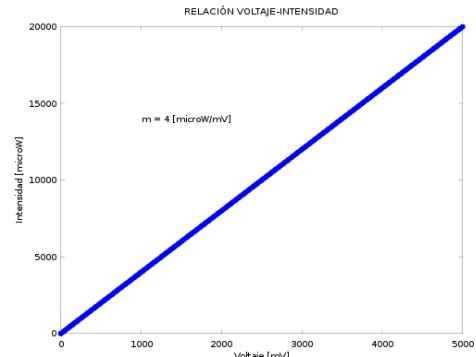


Figura 4. Relación irradiancia vs voltaje.

Una vez absorbidos los fotones infrarrojos, el microcontrolador atmega 328 del vehículo motorizado **infraROSSI**, mide la relación de voltaje en el sensor de electro-recepción activa que es proporcional a la irradiancia del emisor infrarrojo la cual equivale a $4\mu\text{W}/\text{mV}$ lo que significa que por cada $4\mu\text{W}$ de radiación incidente el sensor esta arrojando en su salida 1mV .

$$I(x) = \frac{RB_0}{x^2} \quad (21)$$

Montaje experimental

Materiales del montaje

Para la realización de este montaje se utilizaron los siguientes materiales:

ANÁLISIS DE RESULTADOS



Figura 5. Montaje modulo motorizado infraROSSI para el laboratorio de absorción.

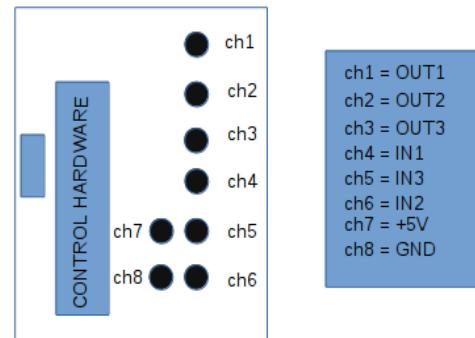


Figura 6. Esquema de pines del sistema de control de hardware.

- Ordenador con sistema operativo GNU-Linux.
- Modulo motorizado infraROSSI.
- Software de control FREEinfraROSSI.
- Un octavo de cartulina de los siguientes colores: azul, amarilla, roja, verde oscura, negra.
- Modulo bluetooth para pc.

Montaje

Colocar el sensor de electro-recepción activa junto con el control de avance y el control de hardware en el vehículo motorizado infraROSSI como se observa en la figura 5.

Los esquemas de pines de las partes utilizadas para el montaje se muestran en las figuras 6 , 7 y 8 ; se debe conectar el pin ch1 con el pin ea4, el pin ch3 con el pin ca9, el pin de ch4 con el pin ea2, el pin ch7 con el pin ca2, el pin ch8 con el pin ca1, el pin ca5 con el pin ea1, el pin ca6 con el pin ea3, por ultimo el pin ca7 y el pin ca8 son las conexiones al motor.

Situar el vehículo motorizado infraROSSI a una distancia de 30 cm de la muestra (octavo de cartulina) tal como se muestra en la figura 3, la muestra debe estar perpendicular a la parte frontal del vehículo. Abrir una terminal de GNU-Linux y escribir **infrarossi**, oprimir enter y la clave de superusuario, luego de abrir el programa debe oprimir el botón on, esperar que se empareje el bluetooth, una vez emparejado el modulo bluetooth, el programa desplegará un tercer menú como se observa en la figura 9, oprimir el botón de absorción y esperar que el programa tome los datos necesarios.

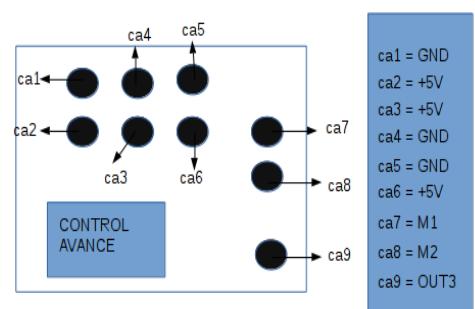


Figura 7. Esquema de pines del sistema de control de avance.

Ahora el software de control FREEinfraROSSI le envía vía bluetooth la señal de avanzar y capturar datos al vehículo motorizado infraROSSI el cual avanza 2 milímetros por cada paso, recolecta 140 datos por cada avance enviándolos vía bluetooth al ordenador en donde el software de control realiza un análisis estadístico de los mismos, cuando termina este análisis envía una señal al modulo motorizado infraROSSI vía bluetooth, indicándole que avance nuevamente y repita el proceso, esto lo realiza 117 veces; una vez terminado de recoger todos los datos realiza una estadística sobre toda la muestra de estos datos capturados.

Análisis de resultados

El programa FREEinfraROSSI después de recoger los datos realiza un análisis estadístico de los mismos con un (Ostle, 1981) “ajuste lineal de una función exponencial de la forma $Y^* = aX^b$, siendo Y la intensidad irradiada, a la

ANÁLISIS DE RESULTADOS

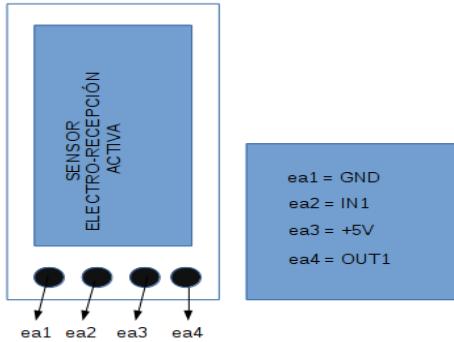


Figura 8. Esquema de pines del sensor de electro-recepción activa en infrarrojo.

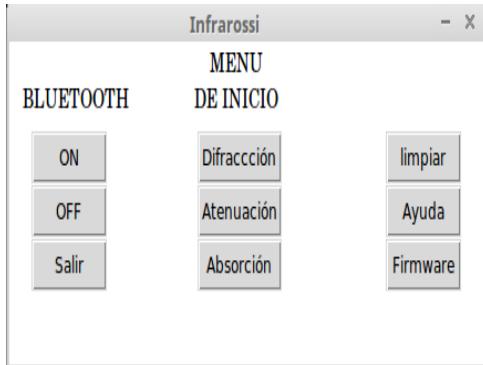


Figura 9. Interfaz grafica software FREEinfraROSSI.

amplitud, X la distancia de avance, b el valor del exponente de atenuación, Y^* es el valor esperado de la variable física; aplicando logaritmo natural para linealizar se obtiene:

$$\ln(Y^*) = \ln(a) + b\ln(X) \dots \Rightarrow V^* = A + bU \quad (22)$$

donde V^* es $\ln(Y^*)$, A es el $\ln(a)$ y U es igual al $\ln(X)$.

La suma de todos los errores debe ser diferentes a cero $\sum e \neq 0$.

El exponente de ajuste b se halla con la varianza de U sobre V dividida entre la varianza de U sobre U , obteniendo

$$b = \frac{S_{UV}}{S_U^2} = \frac{\frac{1}{n} \sum_{i=1}^n UV - \bar{U}\bar{V}}{\frac{1}{n} \sum_{i=1}^n U^2 - \bar{U}^2} \quad (23)$$

ahora el valor de A sera igual al valor medio del logaritmo de la variable física \bar{V} menos el valor de la multiplicación entre el exponente b y el valor medio del logaritmo de la variable

experimental:

$$A = \bar{V} - b\bar{U} \quad (24)$$

deshaciendo el logaritmo de A se obtiene el valor de la amplitud a:

$$a = \text{antiln}(A) = \text{antiln}(\bar{V} - b\bar{U}) \quad (25)$$

de modo que el ajuste efectuado es:

$$Y^* = aX^b = [\text{antiln}(\bar{V} - b\bar{U})] * X^{\left(\frac{1}{n} \sum_{i=1}^n UV - \bar{U}\bar{V}\right)} \quad (26)$$

la bondad del ajuste es el error cuadrático medio o ECM y es igual a:

$$ECM = \frac{\sum_{i=1}^n e_i^2}{n} \quad (27)$$

siendo e_i cada una de las diferencias entre las variables físicas y los valores estimados para las variables físicas $e_i = Y_i - Y^*$; al haber transformado la variable física ya no se minimiza $\sum e^2$ sino $\sum (\ln(Y) - \ln(Y^*))^2$, de ahí que $\sum e \neq 0$.

Este análisis se realizo previamente con un espejo, el cual deja como patrón de referencia o de intensidad ideal, el valor medio de $0,000259X^{-2}$ y en la figura 10 aparece con el nombre *ideal*; ahora el programa calcula un valor estimado para los datos de la siguiente manera: la intensidad transmitida I_T debe ser igual a la intensidad ideal I_{ideal} menos la intensidad que mide el sensor I_{real} .

$$I_t = I_{ideal} - I_{real}$$

El factor de perdida en la intensidad T debe ser igual al cociente de la intensidad I_t y la intensidad ideal I_{ideal}

$$T = \frac{I_t}{I_{ideal}}$$

por lo tanto el promedio en el factor de perdida debe ser igual a la transmittancia o mejor dicho a la energía que se transmitió a la muestra,

$$\bar{T} = \sum T_i / N$$

como la reflectancia es igual a $R = 1 - T$ (ecuación 10), el valor de la amplitud a , debe ser igual:

$$a = (1 - \bar{T})(0,000259)$$

por lo que ahora la intensidad estimada en la figura 10 con la etiqueta *Exp*, en función del inverso de la distancia, disminuye con la reflectancia R :

$$I^* = (1 - \bar{T})(0,000259)X^{-2}$$

ANÁLISIS DE RESULTADOS

Tabla 1

Análisis del experimento de absorción con el vehículo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**.

Color	a ($x 10^{-5}$)	T	R	ECM ($x 10^{-7}$)
Azul	4.5	0.825	0.175	1.3896
Amarillo	4.3	0.8322	0.1678	0.0439
Verde	4.2	0.8392	0.1608	0.2399
Negro	3.4	0.8684	0.1316	0.332
Rojo	4.7	0.8192	0.1808	0.0931

Nota. Datos de amplitud, transmitancia y reflectancia de cinco muestras de cartulina de diferentes colores, recolectados con el modulo motorizado **infraROSSI** y su software de control **FREEinfraROSSI**.

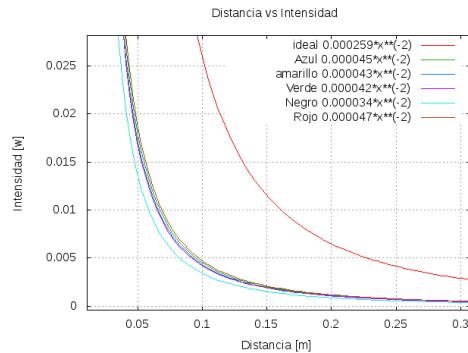


Figura 11. Imagen generada con la librería matplotlib de python 2.7, donde se aprecia las funciones estimadas por el programa y la función ideal de intensidad.

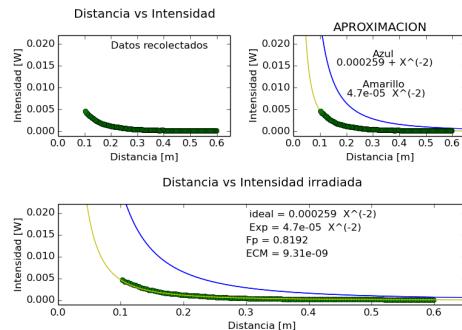


Figura 10. Imagen generada por el programa **FREEinfraROSSI**.

después de este análisis el software **FREEinfraROSSI** realiza la gráfica mostrando en ella (figura 10) la intensidad ideal $I_{ideal}(x)$, la intensidad estimada debido a la reflectancia I^* y el factor de transmitancia o factor de caída en la intensidad F_p .

En la figura 10 se observan los datos capturados en el experimento, las funciones de color azul en la gráfica es intensidad ideal I_{ideal} , la función de color amarillo es la gráfica estimada $I^* = Exp$ y los puntos de color negro son los datos recolectados.

En la tabla 1 se muestra la función intensidad a en $[W]/[m^2]$, estimado por el software **FREEinfraROSSI**, para cada una de las diferentes muestras del mismo material de cartulina pero diferentes pigmentos; la cartulina en la cual hubo más transmitancia fue la de color negro, pues el 86,84 % de intensidad de la luz incidente sobre ella fue transmitida al material, llegando solo un 13,16 % de esta al sensor; ahora la muestra que mayor reflectancia tuvo fue la de color rojo pues el 18,08 % de la intensidad lumínica llegó al sensor, indicando que solo el 81,92 % de esta fue transmitida al material de muestra.

El error cuadrático medio indica que el valor estimado para la función intensidad que se produce en el diodo cuando sobre este incide radiación infrarroja que ha interactuado con la muestra, es explicado en un 99,999958 % con este tratamiento (figura 11 y 12).

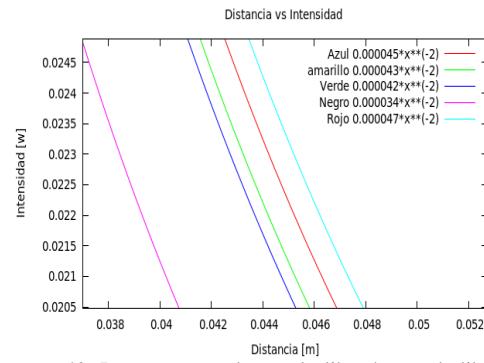


Figura 12. Imagen generada con la librería matplotlib de python 2.7, donde se aprecia una porción distinguible de las funciones estimadas por el programa.

REFERENCIAS

Conclusiones

- * La muestra en la cual hubo más transmitancia, es la cartulina de color negro, pues el 86,84 % de intensidad de la luz incidente sobre ella fue transmitida al material, llegando solo un 13,16 % de esta al sensor.
- * La muestra que mayor reflectancia tuvo fue la de color rojo pues el 18,08 % de la intensidad lumínica llegó al sensor, indicando que solo el 81,92 % de esta fue transmitida al material de muestra.
- * El modulo motorizado **infraROSSI** y su software de control ilustran de manera cuantitativa y cualitativa fenómenos ondulatorios y corpusculares de la radiación electromagnética como la atenuación con el inverso del cuadrado de la distancia, la radianza, la irradiancia, fenómenos de transporte e inyección, calculando el factor de perdida de intensidad en el sensor y de esta manera predice los valores de reflectancia y transmitancia del material estudiado.
- * El error cuadrático medio indica que solo el 0,000004197 % del experimento no se explica con este modelo **físico-matemático**, dejando

un 99,9999958 % de fiabilidad en el ajuste estadístico utilizado en este trabajo.

- * La física que se encuentra de manera implícita y explícita en este trabajo hace que el instrumento motorizado **infraROSSI** y su software de control **FREEinfraROSSI** sea un herramienta indispensable en el aula de clase, tanto de profesionales como estudiantes de ciencias afines.

Referencias

- Albella, J. M., y Martínez-Duart, J. M. (1996). *Fundamentos de electrónica física y microelectrónica*. Addison-Wesley Iberoamericana.
- Gautreau, R., Savin, W., y Velazquez Valle, D. (2001). Física moderna.
- Hecht, E., Dal Col, R., Talavera, R. W., y Pérez, J. M. G. (2000). *Óptica*. Addison Wesley.
- Ostle, B. (1981). *Estadística aplicada*. Limusa.
- Shalimova, K. V., y Grdiam, A. (1975). *Física de los semiconductores*.

Apéndice E. Códigos del programa FREEinfraROSSI

Apéndice E1. LICENSE

/home/diego/Documentos/Free-infrarossi/LICENSE	
Página 1 de 12	
vie 18 mar 2016 22:19:44 COT	

```

1          GNU GENERAL PUBLIC LICENSE
2          Version 3, 29 June 2007
3
4  Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
5  Everyone is permitted to copy and distribute verbatim copies
6  of this license document, but changing it is not allowed.
7
8          Preamble
9
10         The GNU General Public License is a free, copyleft license for
11  software and other kinds of works.
12
13         The licenses for most software and other practical works are designed
14  to take away your freedom to share and change the works. By contrast,
15  the GNU General Public License is intended to guarantee your freedom to
16  share and change all versions of a program--to make sure it remains free
17  software for all its users. We, the Free Software Foundation, use the
18  GNU General Public License for most of our software; it applies also to
19  any other work released this way by its authors. You can apply it to
20  your programs, too.
21
22         When we speak of free software, we are referring to freedom, not
23  price. Our General Public Licenses are designed to make sure that you
24  have the freedom to distribute copies of free software (and charge for
25  them if you wish), that you receive source code or can get it if you
26  want it, that you can change the software or use pieces of it in new
27  free programs, and that you know you can do these things.
28
29         To protect your rights, we need to prevent others from denying you
30  these rights or asking you to surrender the rights. Therefore, you have
31  certain responsibilities if you distribute copies of the software, or if
32  you modify it: responsibilities to respect the freedom of others.
33
34         For example, if you distribute copies of such a program, whether
35  gratis or for a fee, you must pass on to the recipients the same
36  freedoms that you received. You must make sure that they, too, receive
37  or can get the source code. And you must show them these terms so they
38  know their rights.
39
40         Developers that use the GNU GPL protect your rights with two steps:
41  (1) assert copyright on the software, and (2) offer you this License
42  giving you legal permission to copy, distribute and/or modify it.
43
44         For the developers' and authors' protection, the GPL clearly explains
45  that there is no warranty for this free software. For both users' and
46  authors' sake, the GPL requires that modified versions be marked as
47  changed, so that their problems will not be attributed erroneously to
48  authors of previous versions.
49
50         Some devices are designed to deny users access to install or run
51  modified versions of the software inside them, although the manufacturer
52  can do so. This is fundamentally incompatible with the aim of
53  protecting users' freedom to change the software. The systematic
54  pattern of such abuse occurs in the area of products for individuals to
55  use, which is precisely where it is most unacceptable. Therefore, we

```

/home/diego/Documentos/Free-infrarossi/LICENSE
Página 2 de 12

vie 18 mar 2016 22:19:44 COT

56 have designed this version of the GPL to prohibit the practice for those
57 products. If such problems arise substantially in other domains, we
58 stand ready to extend this provision to those domains in future versions
59 of the GPL, as needed to protect the freedom of users.
60

61 Finally, every program is threatened constantly by software patents.
62 States should not allow patents to restrict development and use of
63 software on general-purpose computers, but in those that do, we wish to
64 avoid the special danger that patents applied to a free program could
65 make it effectively proprietary. To prevent this, the GPL assures that
66 patents cannot be used to render the program non-free.
67

68 The precise terms and conditions for copying, distribution and
69 modification follow.
70

TERMS AND CONDITIONS

0. Definitions.

75 "This License" refers to version 3 of the GNU General Public License.
76

77 "Copyright" also means copyright-like laws that apply to other kinds of
78 works, such as semiconductor masks.
79

80 "The Program" refers to any copyrightable work licensed under this
81 License. Each licensee is addressed as "you". "Licensees" and
82 "recipients" may be individuals or organizations.
83

84 To "modify" a work means to copy from or adapt all or part of the work
85 in a fashion requiring copyright permission, other than the making of an
86 exact copy. The resulting work is called a "modified version" of the
87 earlier work or a work "based on" the earlier work.
88

89 A "covered work" means either the unmodified Program or a work based
90 on the Program.
91

92 To "propagate" a work means to do anything with it that, without
93 permission, would make you directly or secondarily liable for
94 infringement under applicable copyright law, except executing it on a
95 computer or modifying a private copy. Propagation includes copying,
96 distribution (with or without modification), making available to the
97 public, and in some countries other activities as well.
98

99 To "convey" a work means any kind of propagation that enables other
100 parties to make or receive copies. Mere interaction with a user through
101 a computer network, with no transfer of a copy, is not conveying.
102

103 An interactive user interface displays "Appropriate Legal Notices"
104 to the extent that it includes a convenient and prominently visible
105 feature that (1) displays an appropriate copyright notice, and (2)
106 tells the user that there is no warranty for the work (except to the
107 extent that warranties are provided), that licensees may convey the
108 work under this License, and how to view a copy of this License. If
109 the interface presents a list of user commands or options, such as a
110 menu, a prominent item in the list meets this criterion.

/home/diego/Documentos/Free-infrarossi/LICENSE	
Página 3 de 12	vie 18 mar 2016 22:19:44 COT
111	1. Source Code.
112	
113	
114	The "source code" for a work means the preferred form of the work
115	for making modifications to it. "Object code" means any non-source
116	form of a work.
117	
118	A "Standard Interface" means an interface that either is an official
119	standard defined by a recognized standards body, or, in the case of
120	interfaces specified for a particular programming language, one that
121	is widely used among developers working in that language.
122	
123	The "System Libraries" of an executable work include anything, other
124	than the work as a whole, that (a) is included in the normal form of
125	packaging a Major Component, but which is not part of that Major
126	Component, and (b) serves only to enable use of the work with that
127	Major Component, or to implement a Standard Interface for which an
128	implementation is available to the public in source code form. A
129	"Major Component", in this context, means a major essential component
130	(kernel, window system, and so on) of the specific operating system
131	(if any) on which the executable work runs, or a compiler used to
132	produce the work, or an object code interpreter used to run it.
133	
134	The "Corresponding Source" for a work in object code form means all
135	the source code needed to generate, install, and (for an executable
136	work) run the object code and to modify the work, including scripts to
137	control those activities. However, it does not include the work's
138	System Libraries, or general-purpose tools or generally available free
139	programs which are used unmodified in performing those activities but
140	which are not part of the work. For example, Corresponding Source
141	includes interface definition files associated with source files for
142	the work, and the source code for shared libraries and dynamically
143	linked subprograms that the work is specifically designed to require,
144	such as by intimate data communication or control flow between those
145	subprograms and other parts of the work.
146	
147	The Corresponding Source need not include anything that users
148	can regenerate automatically from other parts of the Corresponding
149	Source.
150	
151	The Corresponding Source for a work in source code form is that
152	same work.
153	
154	2. Basic Permissions.
155	
156	All rights granted under this License are granted for the term of
157	copyright on the Program, and are irrevocable provided the stated
158	conditions are met. This License explicitly affirms your unlimited
159	permission to run the unmodified Program. The output from running a
160	covered work is covered by this License only if the output, given its
161	content, constitutes a covered work. This License acknowledges your
162	rights of fair use or other equivalent, as provided by copyright law.
163	
164	You may make, run and propagate covered works that you do not
165	convey, without conditions so long as your license otherwise remains

/home/diego/Documentos/Free-infrarossi/LICENSE
Página 4 de 12

vie 18 mar 2016 22:19:44 COT

166 in force. You may convey covered works to others for the sole purpose
167 of having them make modifications exclusively for you, or provide you
168 with facilities for running those works, provided that you comply with
169 the terms of this License in conveying all material for which you do
170 not control copyright. Those thus making or running the covered works
171 for you must do so exclusively on your behalf, under your direction
172 and control, on terms that prohibit them from making any copies of
173 your copyrighted material outside their relationship with you.

174
175 Conveying under any other circumstances is permitted solely under
176 the conditions stated below. Sublicensing is not allowed; section 10
177 makes it unnecessary.

178
179 3. Protecting Users' Legal Rights From Anti-Circumvention Law.
180

181 No covered work shall be deemed part of an effective technological
182 measure under any applicable law fulfilling obligations under article
183 11 of the WIPO copyright treaty adopted on 20 December 1996, or
184 similar laws prohibiting or restricting circumvention of such
185 measures.

186
187 When you convey a covered work, you waive any legal power to forbid
188 circumvention of technological measures to the extent such circumvention
189 is effected by exercising rights under this License with respect to
190 the covered work, and you disclaim any intention to limit operation or
191 modification of the work as a means of enforcing, against the work's
192 users, your or third parties' legal rights to forbid circumvention of
193 technological measures.

194
195 4. Conveying Verbatim Copies.
196

197 You may convey verbatim copies of the Program's source code as you
198 receive it, in any medium, provided that you conspicuously and
199 appropriately publish on each copy an appropriate copyright notice;
200 keep intact all notices stating that this License and any
201 non-permissive terms added in accord with section 7 apply to the code;
202 keep intact all notices of the absence of any warranty; and give all
203 recipients a copy of this License along with the Program.

204
205 You may charge any price or no price for each copy that you convey,
206 and you may offer support or warranty protection for a fee.

207
208 5. Conveying Modified Source Versions.
209

210 You may convey a work based on the Program, or the modifications to
211 produce it from the Program, in the form of source code under the
212 terms of section 4, provided that you also meet all of these conditions:

- 213
214 a) The work must carry prominent notices stating that you modified
215 it, and giving a relevant date.
216
217 b) The work must carry prominent notices stating that it is
218 released under this License and any conditions added under section
219 7. This requirement modifies the requirement in section 4 to
220 "keep intact all notices".

/home/diego/Documentos/Free-infrarossi/LICENSE	vie 18 mar 2016 22:19:44 COT
Página 5 de 12	
221	
222	c) You must license the entire work, as a whole, under this
223	License to anyone who comes into possession of a copy. This
224	License will therefore apply, along with any applicable section 7
225	additional terms, to the whole of the work, and all its parts,
226	regardless of how they are packaged. This License gives no
227	permission to license the work in any other way, but it does not
228	invalidate such permission if you have separately received it.
229	
230	d) If the work has interactive user interfaces, each must display
231	Appropriate Legal Notices; however, if the Program has interactive
232	interfaces that do not display Appropriate Legal Notices, your
233	work need not make them do so.
234	
235	A compilation of a covered work with other separate and independent
236	works, which are not by their nature extensions of the covered work,
237	and which are not combined with it such as to form a larger program,
238	in or on a volume of a storage or distribution medium, is called an
239	"aggregate" if the compilation and its resulting copyright are not
240	used to limit the access or legal rights of the compilation's users
241	beyond what the individual works permit. Inclusion of a covered work
242	in an aggregate does not cause this License to apply to the other
243	parts of the aggregate.
244	
245	6. Conveying Non-Source Forms.
246	
247	You may convey a covered work in object code form under the terms
248	of sections 4 and 5, provided that you also convey the
249	machine-readable Corresponding Source under the terms of this License,
250	in one of these ways:
251	
252	a) Convey the object code in, or embodied in, a physical product
253	(including a physical distribution medium), accompanied by the
254	Corresponding Source fixed on a durable physical medium
255	customarily used for software interchange.
256	
257	b) Convey the object code in, or embodied in, a physical product
258	(including a physical distribution medium), accompanied by a
259	written offer, valid for at least three years and valid for as
260	long as you offer spare parts or customer support for that product
261	model, to give anyone who possesses the object code either (1) a
262	copy of the Corresponding Source for all the software in the
263	product that is covered by this License, on a durable physical
264	medium customarily used for software interchange, for a price no
265	more than your reasonable cost of physically performing this
266	conveying of source, or (2) access to copy the
267	Corresponding Source from a network server at no charge.
268	
269	c) Convey individual copies of the object code with a copy of the
270	written offer to provide the Corresponding Source. This
271	alternative is allowed only occasionally and noncommercially, and
272	only if you received the object code with such an offer, in accord
273	with subsection 6b.
274	
275	d) Convey the object code by offering access from a designated

/home/diego/Documentos/Free-infrarossi/LICENSE
Página 6 de 12

vie 18 mar 2016 22:19:44 COT

276 place (gratis or for a charge), and offer equivalent access to the
277 Corresponding Source in the same way through the same place at no
278 further charge. You need not require recipients to copy the
279 Corresponding Source along with the object code. If the place to
280 copy the object code is a network server, the Corresponding Source
281 may be on a different server (operated by you or a third party)
282 that supports equivalent copying facilities, provided you maintain
283 clear directions next to the object code saying where to find the
284 Corresponding Source. Regardless of what server hosts the
285 Corresponding Source, you remain obligated to ensure that it is
286 available for as long as needed to satisfy these requirements.
287
288 e) Convey the object code using peer-to-peer transmission, provided
289 you inform other peers where the object code and Corresponding
290 Source of the work are being offered to the general public at no
291 charge under subsection 6d.
292
293 A separable portion of the object code, whose source code is excluded
294 from the Corresponding Source as a System Library, need not be
295 included in conveying the object code work.
296
297 A "User Product" is either (1) a "consumer product", which means any
298 tangible personal property which is normally used for personal, family,
299 or household purposes, or (2) anything designed or sold for incorporation
300 into a dwelling. In determining whether a product is a consumer product,
301 doubtful cases shall be resolved in favor of coverage. For a particular
302 product received by a particular user, "normally used" refers to a
303 typical or common use of that class of product, regardless of the status
304 of the particular user or of the way in which the particular user
305 actually uses, or expects or is expected to use, the product. A product
306 is a consumer product regardless of whether the product has substantial
307 commercial, industrial or non-consumer uses, unless such uses represent
308 the only significant mode of use of the product.
309
310 "Installation Information" for a User Product means any methods,
311 procedures, authorization keys, or other information required to install
312 and execute modified versions of a covered work in that User Product from
313 a modified version of its Corresponding Source. The information must
314 suffice to ensure that the continued functioning of the modified object
315 code is in no case prevented or interfered with solely because
316 modification has been made.
317
318 If you convey an object code work under this section in, or with, or
319 specifically for use in, a User Product, and the conveying occurs as
320 part of a transaction in which the right of possession and use of the
321 User Product is transferred to the recipient in perpetuity or for a
322 fixed term (regardless of how the transaction is characterized), the
323 Corresponding Source conveyed under this section must be accompanied
324 by the Installation Information. But this requirement does not apply
325 if neither you nor any third party retains the ability to install
326 modified object code on the User Product (for example, the work has
327 been installed in ROM).
328
329 The requirement to provide Installation Information does not include a
330 requirement to continue to provide support service, warranty, or updates

/home/diego/Documentos/Free-infrarossi/LICENSE
Página 7 de 12

vie 18 mar 2016 22:19:44 COT

331 for a work that has been modified or installed by the recipient, or for
332 the User Product in which it has been modified or installed. Access to a
333 network may be denied when the modification itself materially and
334 adversely affects the operation of the network or violates the rules and
335 protocols for communication across the network.

336
337 Corresponding Source conveyed, and Installation Information provided,
338 in accord with this section must be in a format that is publicly
339 documented (and with an implementation available to the public in
340 source code form), and must require no special password or key for
341 unpacking, reading or copying.

342
343 7. Additional Terms.
344

345 "Additional permissions" are terms that supplement the terms of this
346 License by making exceptions from one or more of its conditions.
347 Additional permissions that are applicable to the entire Program shall
348 be treated as though they were included in this License, to the extent
349 that they are valid under applicable law. If additional permissions
350 apply only to part of the Program, that part may be used separately
351 under those permissions, but the entire Program remains governed by
352 this License without regard to the additional permissions.
353

354 When you convey a copy of a covered work, you may at your option
355 remove any additional permissions from that copy, or from any part of
356 it. (Additional permissions may be written to require their own
357 removal in certain cases when you modify the work.) You may place
358 additional permissions on material, added by you to a covered work,
359 for which you have or can give appropriate copyright permission.
360

361 Notwithstanding any other provision of this License, for material you
362 add to a covered work, you may (if authorized by the copyright holders of
363 that material) supplement the terms of this License with terms:
364

- 365 a) Disclaiming warranty or limiting liability differently from the
366 terms of sections 15 and 16 of this License; or
367
- 368 b) Requiring preservation of specified reasonable legal notices or
369 author attributions in that material or in the Appropriate Legal
370 Notices displayed by works containing it; or
371
- 372 c) Prohibiting misrepresentation of the origin of that material, or
373 requiring that modified versions of such material be marked in
374 reasonable ways as different from the original version; or
375
- 376 d) Limiting the use for publicity purposes of names of licensors or
377 authors of the material; or
378
- 379 e) Declining to grant rights under trademark law for use of some
380 trade names, trademarks, or service marks; or
381
- 382 f) Requiring indemnification of licensors and authors of that
383 material by anyone who conveys the material (or modified versions of
384 it) with contractual assumptions of liability to the recipient, for
385 any liability that these contractual assumptions directly impose on

/home/diego/Documentos/Free-infrarossi/LICENSE
Página 8 de 12

vie 18 mar 2016 22:19:44 COT

386 those licensors and authors.

387
388 All other non-permissive additional terms are considered "further
389 restrictions" within the meaning of section 10. If the Program as you
390 received it, or any part of it, contains a notice stating that it is
391 governed by this License along with a term that is a further
392 restriction, you may remove that term. If a license document contains
393 a further restriction but permits relicensing or conveying under this
394 License, you may add to a covered work material governed by the terms
395 of that license document, provided that the further restriction does
396 not survive such relicensing or conveying.

397
398 If you add terms to a covered work in accord with this section, you
399 must place, in the relevant source files, a statement of the
400 additional terms that apply to those files, or a notice indicating
401 where to find the applicable terms.

402
403 Additional terms, permissive or non-permissive, may be stated in the
404 form of a separately written license, or stated as exceptions;
405 the above requirements apply either way.

406
407 8. Termination.

408
409 You may not propagate or modify a covered work except as expressly
410 provided under this License. Any attempt otherwise to propagate or
411 modify it is void, and will automatically terminate your rights under
412 this License (including any patent licenses granted under the third
413 paragraph of section 11).

414
415 However, if you cease all violation of this License, then your
416 license from a particular copyright holder is reinstated (a)
417 provisionally, unless and until the copyright holder explicitly and
418 finally terminates your license, and (b) permanently, if the copyright
419 holder fails to notify you of the violation by some reasonable means
420 prior to 60 days after the cessation.

421
422 Moreover, your license from a particular copyright holder is
423 reinstated permanently if the copyright holder notifies you of the
424 violation by some reasonable means, this is the first time you have
425 received notice of violation of this License (for any work) from that
426 copyright holder, and you cure the violation prior to 30 days after
427 your receipt of the notice.

428
429 Termination of your rights under this section does not terminate the
430 licenses of parties who have received copies or rights from you under
431 this License. If your rights have been terminated and not permanently
432 reinstated, you do not qualify to receive new licenses for the same
433 material under section 10.

434
435 9. Acceptance Not Required for Having Copies.

436
437 You are not required to accept this License in order to receive or
438 run a copy of the Program. Ancillary propagation of a covered work
439 occurring solely as a consequence of using peer-to-peer transmission
440 to receive a copy likewise does not require acceptance. However,

/home/diego/Documentos/Free-infrarossi/LICENSE
Página 9 de 12

vie 18 mar 2016 22:19:44 COT

441 nothing other than this License grants you permission to propagate or
442 modify any covered work. These actions infringe copyright if you do
443 not accept this License. Therefore, by modifying or propagating a
444 covered work, you indicate your acceptance of this License to do so.
445

446 10. Automatic Licensing of Downstream Recipients.
447

448 Each time you convey a covered work, the recipient automatically
449 receives a license from the original licensors, to run, modify and
450 propagate that work, subject to this License. You are not responsible
451 for enforcing compliance by third parties with this License.
452

453 An "entity transaction" is a transaction transferring control of an
454 organization, or substantially all assets of one, or subdividing an
455 organization, or merging organizations. If propagation of a covered
456 work results from an entity transaction, each party to that
457 transaction who receives a copy of the work also receives whatever
458 licenses to the work the party's predecessor in interest had or could
459 give under the previous paragraph, plus a right to possession of the
460 Corresponding Source of the work from the predecessor in interest, if
461 the predecessor has it or can get it with reasonable efforts.
462

463 You may not impose any further restrictions on the exercise of the
464 rights granted or affirmed under this License. For example, you may
465 not impose a license fee, royalty, or other charge for exercise of
466 rights granted under this License, and you may not initiate litigation
467 (including a cross-claim or counterclaim in a lawsuit) alleging that
468 any patent claim is infringed by making, using, selling, offering for
469 sale, or importing the Program or any portion of it.
470

471 11. Patents.
472

473 A "contributor" is a copyright holder who authorizes use under this
474 License of the Program or a work on which the Program is based. The
475 work thus licensed is called the contributor's "contributor version".
476

477 A contributor's "essential patent claims" are all patent claims
478 owned or controlled by the contributor, whether already acquired or
479 hereafter acquired, that would be infringed by some manner, permitted
480 by this License, of making, using, or selling its contributor version,
481 but do not include claims that would be infringed only as a
482 consequence of further modification of the contributor version. For
483 purposes of this definition, "control" includes the right to grant
484 patent sublicenses in a manner consistent with the requirements of
485 this License.
486

487 Each contributor grants you a non-exclusive, worldwide, royalty-free
488 patent license under the contributor's essential patent claims, to
489 make, use, sell, offer for sale, import and otherwise run, modify and
490 propagate the contents of its contributor version.
491

492 In the following three paragraphs, a "patent license" is any express
493 agreement or commitment, however denominated, not to enforce a patent
494 (such as an express permission to practice a patent or covenant not to
495 sue for patent infringement). To "grant" such a patent license to a

/home/diego/Documentos/Free-infrarossi/LICENSE
Página 10 de 12

vie 18 mar 2016 22:19:44 COT

496 party means to make such an agreement or commitment not to enforce a
497 patent against the party.

498
499 If you convey a covered work, knowingly relying on a patent license,
500 and the Corresponding Source of the work is not available for anyone
501 to copy, free of charge and under the terms of this License, through a
502 publicly available network server or other readily accessible means,
503 then you must either (1) cause the Corresponding Source to be so
504 available, or (2) arrange to deprive yourself of the benefit of the
505 patent license for this particular work, or (3) arrange, in a manner
506 consistent with the requirements of this License, to extend the patent
507 license to downstream recipients. "Knowingly relying" means you have
508 actual knowledge that, but for the patent license, your conveying the
509 covered work in a country, or your recipient's use of the covered work
510 in a country, would infringe one or more identifiable patents in that
511 country that you have reason to believe are valid.

512
513 If, pursuant to or in connection with a single transaction or
514 arrangement, you convey, or propagate by procuring conveyance of, a
515 covered work, and grant a patent license to some of the parties
516 receiving the covered work authorizing them to use, propagate, modify
517 or convey a specific copy of the covered work, then the patent license
518 you grant is automatically extended to all recipients of the covered
519 work and works based on it.

520
521 A patent license is "discriminatory" if it does not include within
522 the scope of its coverage, prohibits the exercise of, or is
523 conditioned on the non-exercise of one or more of the rights that are
524 specifically granted under this License. You may not convey a covered
525 work if you are a party to an arrangement with a third party that is
526 in the business of distributing software, under which you make payment
527 to the third party based on the extent of your activity of conveying
528 the work, and under which the third party grants, to any of the
529 parties who would receive the covered work from you, a discriminatory
530 patent license (a) in connection with copies of the covered work
531 conveyed by you (or copies made from those copies), or (b) primarily
532 for and in connection with specific products or compilations that
533 contain the covered work, unless you entered into that arrangement,
534 or that patent license was granted, prior to 28 March 2007.

535
536 Nothing in this License shall be construed as excluding or limiting
537 any implied license or other defenses to infringement that may
538 otherwise be available to you under applicable patent law.

539
540 12. No Surrender of Others' Freedom.

541
542 If conditions are imposed on you (whether by court order, agreement or
543 otherwise) that contradict the conditions of this License, they do not
544 excuse you from the conditions of this License. If you cannot convey a
545 covered work so as to satisfy simultaneously your obligations under this
546 License and any other pertinent obligations, then as a consequence you may
547 not convey it at all. For example, if you agree to terms that obligate you
548 to collect a royalty for further conveying from those to whom you convey
549 the Program, the only way you could satisfy both those terms and this
550 License would be to refrain entirely from conveying the Program.

/home/diego/Documentos/Free-infrarossi/LICENSE
Página 11 de 12

vie 18 mar 2016 22:19:44 COT

551
552 13. Use with the GNU Affero General Public License.
553
554 Notwithstanding any other provision of this License, you have
555 permission to link or combine any covered work with a work licensed
556 under version 3 of the GNU Affero General Public License into a single
557 combined work, and to convey the resulting work. The terms of this
558 License will continue to apply to the part which is the covered work,
559 but the special requirements of the GNU Affero General Public License,
560 section 13, concerning interaction through a network will apply to the
561 combination as such.

562
563 14. Revised Versions of this License.
564

565 The Free Software Foundation may publish revised and/or new versions of
566 the GNU General Public License from time to time. Such new versions will
567 be similar in spirit to the present version, but may differ in detail to
568 address new problems or concerns.

569
570 Each version is given a distinguishing version number. If the
571 Program specifies that a certain numbered version of the GNU General
572 Public License "or any later version" applies to it, you have the
573 option of following the terms and conditions either of that numbered
574 version or of any later version published by the Free Software
575 Foundation. If the Program does not specify a version number of the
576 GNU General Public License, you may choose any version ever published
577 by the Free Software Foundation.

578
579 If the Program specifies that a proxy can decide which future
580 versions of the GNU General Public License can be used, that proxy's
581 public statement of acceptance of a version permanently authorizes you
582 to choose that version for the Program.

583
584 Later license versions may give you additional or different
585 permissions. However, no additional obligations are imposed on any
586 author or copyright holder as a result of your choosing to follow a
587 later version.

588
589 15. Disclaimer of Warranty.

590
591 THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY
592 APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT
593 HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY
594 OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO,
595 THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
596 PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM
597 IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF
598 ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

599
600 16. Limitation of Liability.

601
602 IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
603 WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS
604 THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
605 GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE

/home/diego/Documentos/Free-infrarossi/LICENSE
Página 12 de 12

vie 18 mar 2016 22:19:44 COT

606 USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF
607 DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD
608 PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS),
609 EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF
610 SUCH DAMAGES.

611
612 17. Interpretation of Sections 15 and 16.
613

614 If the disclaimer of warranty and limitation of liability provided
615 above cannot be given local legal effect according to their terms,
616 reviewing courts shall apply local law that most closely approximates
617 an absolute waiver of all civil liability in connection with the
618 Program, unless a warranty or assumption of liability accompanies a
619 copy of the Program in return for a fee.

620
621 END OF TERMS AND CONDITIONS
622
623
624

Apéndice E2. copyright

/home/diego/Documentos/Free-infrarossi/copyright	
Página 1 de 1	
1	Programa de control de hardware e ilustración física de las propiedades de las ondas electromagnéticas en el espectro infrarrojo.
2	Copyright (C) 2016-01-01 Universidad Distrital Francisco Jose, Diego Alberto Parra Garzón, Dr. Julian Andres Salamanca Bernal.
3	El programa free infrarroso es software libre; puedes redistribuirlo y / o modificarlo bajo los términos de la Licencia Pública General GNU publicada por la Fundación para el Software Libre; ya sea la versión 3 de la Licencia, o (a su elección) cualquier versión posterior.
4	Este programa se distribuye con la esperanza de que sea útil, pero SIN NINGUNA GARANTÍA; ni siquiera la garantía implícita de COMERCIALIZACIÓN o IDONEIDAD PARA UN PROPÓSITO PARTICULAR. Vea el Licencia Pública General GNU para más detalles.
5	Debería haber recibido una copia de la Licencia Pública General de GNU junto con este programa; si no, escriba a la Free Software Foundation, Inc., 51 Franklin Street, Quinto Piso, Boston, MA 02110-1301 EE.UU..
6	

Apéndice E3. README.md

	/home/diego/Documentos/Free-infrarossi/README.md	
Página 1 de 1		mar 05 abr 2016 12:00:54 COT
1	Programa de control de hardware e ilustración física de las propiedades de las ondas electromagnéticas en el espectro infrarrojo.	»
2	Copyright (C) 2016-01-01 Universidad Distrital Francisco Jose, Diego Alberto Parra Garzón, Dr. Julian Andres Salamanca Bernal.	»
3	El programa free infrarroso es software libre; puedes redistribuirlo y / o modificarlo bajo los términos de la Licencia Pública General GNU publicada por la Fundación para el Software Libre; ya sea la versión 3 de la Licencia, o (a su elección) cualquier versión posterior.	»
4	Este programa se distribuye con la esperanza de que sea útil, pero SIN NINGUNA GARANTÍA; ni siquiera la garantía implícita de COMERCIALIZACIÓN o IDONEIDAD PARA UN PROPÓSITO PARTICULAR. Vea el Licencia Pública General GNU para más detalles.	»
5	Debería haber recibido una copia de la Licencia Pública General de GNU junto con este programa; si no, escriba a la Free Software Foundation, Inc., 51 Franklin Street, Quinto Piso, Boston, MA 02110-1301 EE.UU..	»
6		

Apéndice E4. instalador.py

```

/home/diego/Documentos/Free-infrarossi/install/instalador.py          mar 05 abr 2016 12:02:40 COT
Página 1 de 2

1  #!/usr/bin/python
2  #!**-* coding:utf-8 *-*
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Diego Alberto Parra Garzón
13 # Dr Julian Andres Salamanca Bernal
14 # Colombia, Bogota D.C.

15
16 import os
17 import time
18 class Instalador:
19     def Presentacion(self):
20         os.system("clear")
21         print chr(27)+"[2;32m"+ "\n\n\t @@@@@@@@@@@@CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC"
22         print "\t\t Instalador de free_infrarossi"
23         print "\t @@@@@@@@@@@@CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC"
24
25         print chr(27)+"[0m"+ "\n \n\t Bienvenido al software free_infrarossi, el
26         cual le permitira tener\n\t varias horas de diversión; este software fue
27         diseñado para el control,\n\t recolección y análisis de datos del vehículo
28         motorizado infrarossi,\n\t el cual permite ilustrar el estudio de las
29         propiedades de las ondas\n\t electromagnéticas en el espectro infrarrojo,
30         tales como la difracción,\n\t atenuación y absorción. Disfrutara de una
31         interfaz amigable al usuario."
32         print "\n\n\t @@@@@@@@@@@@CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC"
33         print chr(27)+"[5;36m"+ "\t\t DESEA CONTINUAR CON LA INSTALACION: "
34         print chr(27)+"[0m"+ "\t @@@@@@@@@@@@CCCCCCCCCCCCCCCCCCCCCCCCCCCC"
35         print "\t Oprima 1 para si \n\t Oprima 2 para no"
36         Pr1 = int(raw_input("Ingrese su respuesta: "))
37         if Pr1 == 1:
38             self.Instalar()
39         elif Pr1 ==2:
40             self.exit()
41             os.system("killall python")
42         else:
43             self.exit()
44             os.system("killall python")
45         def Instalar(self):
46             os.system("clear")
47             print chr(27)+"[2;32m"+ "\n\n ¿Que desea hacer ?\n\n"
48             print chr(27)+"[0m"+ "\n\t oprima 1 para instalar free_infrarossi"
49             print "\n\t Oprima 2 para desinstalar free_infrarossi"
50             print chr(27)+"[3;36m"+ "\n\t Oprima cualquier tecla para SALIR del
51             instalador\n\n"
52             Pr2 = int(raw_input(chr(27)+"[5;32m"+ "Ingrese su petición : "+chr(27)+"[0m"))
53             if Pr2 == 1:
54                 os.system("apt-get update ")

```

```

/home/diego/Documentos/Free-infrarossi/install/instalador.py
Página 2 de 2
mar 05 abr 2016 12:02:40 COT

49     os.system("apt-get install xterm bluez* gcc g++ emacs gnuplot gnuplot-qt evince octave python-matplotlib python-numpy python-tk python-gnuplot python-serial python-visual* libgtkglextmm* arduino fritzing binutils")
50     archi = open('/etc/bash.bashrc', 'a+')
51     archi.write("\nalias infrarossi='cd ~/Documentos/Free-infrarossi/free_infrarossi && ./infrarossi'")
52     archi.close()
53     print chr(27)+"[5;32m"+"INSTALACION TERMINADA "
54     print "Disfrute su software \n reinicie su pc"
55
56 elif Pr2 == 2:
57     os.system("apt-get --purge remove emacs gnuplot gnuplot-qt evince octave python-matplotlib python-scipy python-numpy python-tk python-gnuplot python-serial python-visual* libgtkglextmm* arduino fritzing")
58     os.system("apt-get autoremove")
59     os.system("apt-get update")
60     os.system("bash unistall.sh")
61     os.system("clear")
62     print "\t\t\tDESINSTALACION EXITOSA "
63     print "Se ha creado una copia de respaldo del archivo"+chr(27)+"[3;36m+"+"\n\n\n/etc/bash.bashrc como /etc/bash.bashrc.respaldo\n" + chr(27)+ "[0m"+"\n\n\n si tiene algun problema con este archivo despues de la desinstalación;\n solamente ejecute este comando en la terminal con permisos\n de administrador y sin las comillas \n\n \t "+chr(27)+"[5;31m"+ "+ANOTELO QUE ES \n\t MUY IMPORTANTE"+chr(27)+"[0m"+"\n\n"+chr(27)+"[3;33m"+cp /etc/bash.bashrc.respaldo /etc/bash.bashrc"+chr(27)+"[0m"+ "\n\n"
64     time.sleep(60)
65     print "Desinstalacion completada ---"
66     print "reinicie su pc"
67     self.exit()
68
69 else:
70     print "cerrando el instalador"
71     self.exit()
72
73 def exit(self):
74     exit()
75     exit()
76     exit()
77
78 def __init__(self):
79     self.Presentacion()
80     self.__del__()
81
82 def __del__(self):
83     print chr(27)+"[5;33m"+"FIN DEL PROGRAMA"
84     os.system("exit")
85     os.system("rm logs.txt")
86     time.sleep(4)
87
88 if __name__ == "__main__":
89     Instalador()
90

```

Apéndice E5. INSTALADOR.py

```
/home/diego/Documentos/Free-infrarossi/install/INSTALADOR.py  
Página 1 de 1
```

```
mar 05 abr 2016 12:12:39 COT
```

```
1 #!/usr/bin/python  
2 #!**-* coding:utf-8 **-*  
3 # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo  
4 # los términos de la licencia pública general de GNU, según es publicada  
5 # por la free software fundation bien la versión 3 de la misma licencia  
6 # o de cualquier versión posterior. (según su elección ).  
7 # Si usted hace alguna modificación en esta aplicación, deberá siempre  
8 # mencionar el autor original de la misma.  
9 # Autor:  
10 # Universidad Distrital Francisco Jose  
11 # Grupo de fisica e informatica  
12 # Diego Alberto Parra Garzón  
13 # Dr Julian Andres Salamanca Bernal  
14 # Colombia, Bogota D.C.  
15  
16 import os  
17 os.system('sudo python instalador.py')  
18 os.system("infrarossi= 'cd ~/Documentos/Free-infrarossi/free_infrarossi &&  
19 ./infrarossi'")
```

Apéndice E6. unistall.py

```
/home/diego/Documentos/Free-infrarossi/install/unistall.sh  
Página 1 de 1  
mar 05 abr 2016 12:13:21 COT  
1 #!/bin/bash  
2 # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo  
3 # los términos de la licencia pública general de GNU, según es publicada  
4 # por la free software fundation bien la versión 3 de la misma licencia  
5 # o de cualquier versión posterior. (según su elección ).  
6 # Si usted hace alguna modificación en esta aplicación, deberá siempre  
7 # mencionar el autor original de la misma.  
8 # Autor:  
9 # Universidad Distrital Francisco Jose  
10 # Grupo de física e informática  
11 # Diego Alberto Parra Garzón  
12 # Dr Julian Andres Salamanca Bernal  
13 # Colombia, Bogota D.C.  
14  
15 cp /etc/bash.bashrc /etc/bash.bashrc.respaldo  
16 rm -rf ~/Documentos/Free-infrarossi  
17 cat /etc/bash.bashrc | grep -n "infrarossi" | cut -d ":" -f 1,1 > logs.txt  
18 NUM=`cat logs.txt`  
19 LET=`echo d`  
20 LINEA=`echo -e "$NUM$LET"`  
21 sed -i "$LINEA" /etc/bash.bashrc  
22
```

Apéndice E7. *infrarossi.py*

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/infrarossi.py
Página 1 de 4                                         mar 05 abr 2016 12:14:29 COT

1  #!/usr/bin/python
2  # -*- coding:utf-8 -*-
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Diego Alberto Parra Garzón
13 # Dr Julian Andres Salamanca Bernal
14 # Colombia, Bogota D.C.

15
16 import numpy as np
17 import pylab as pl
18 import serial
19 import os
20 import subprocess
21 import math
22 import time
23 import Gnuplot
24 from Tkinter import *
25 import tkMessageBox
26 import Tkinter
27 import shutil
28
29 class App:
30     def Modulo(self):
31         bicho = Tk()
32         bicho.geometry("430x180+180+60")
33         bicho.config(bg="white")
34         bicho.title("Infrarossi")
35         bicho.resizable(width=0, height=0)
36
37     def Salir():
38         tkMessageBox.showinfo("Infrarossi", message= "! Cerrando el programa i")
39         os.system("rm MAC.txt MACD.txt puerto.txt conexion.txt direccion.txt")
40         os.system("rm log.txt mac.txt macd.txt texpus.log x.log")
41         os.system("rm MAC.txt MACD.txt puerto.txt")
42         os.system("rm bin/conexion.txt bin/direccion.txt")
43         os.system("sh bin/d_Blu.sh &")
44         exit()
45
46     def Verifica():
47         tkMessageBox.showinfo("Infrarossi", message= "! Conectando con el dispositivo, por favor espere i")
48         os.system('xterm -T Infrarossi -geom 50x8+185+100 +cm -bg blue -e sh c_Blu.sh ')
49         Conecta()
50
51     def Conecta():
52         os.system('rfcomm -a > conexion.txt | cut -d ":" -f 1,1')

```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/infrarossi.py
Página 2 de 4                                         mar 05 abr 2016 12:14:29 COT

    conexion.txt > direccion.txt ')
53     puerto = open('direccion.txt', 'r')
54     self.puerto = puerto.read(7)
55     puerto.close()
56     self.puerta = "/dev/" +self.puerto
57     print self.puerta
58
59     Conectar()
60
61     def Conectar():
62         tkMessageBox.showinfo("Infrarossi", message= "! Verificando conexion i")
63         print self.puerta
64         try:
65             arduino = serial.Serial(self.puerta, 9600)
66             time.sleep(4)
67                 arduino.write("aa")
68                 Valido()
69                 Valido()
70         except:
71             tkMessageBox.showinfo("Infrarossi", message= "! No hay conexion i")
72             # os.system('xterm -T Infrarossi -geom 50x8+185+100 +cm -bg blue -e sh c_Blu.sh &')
73
74
75         def Valido():
76             # ..... Botones menu de inicio .....
77             x1 = int(150)
78             y3 = int(300)
79             lblRapidez = Label(bicho, text="\nMENU\nDE INICIO", fg = ("black"), bg =
80             = ("white"), font = ("Century Schoolbook L",10)).place(x=15+x1, y=280
81             -y3)
80             btnComenzar = Button(bicho, text= "Difracción", width=6, height=1,
81             command= Difraccion).place(x=20+x1, y=350-y3)
81             btnDetener= Button(bicho, text= "Atenuación", width=6, height=1,
81             command= Atenuacion).place(x=20+x1, y=380-y3)
82             btnLimpiar = Button(bicho, text= "Absorción", width=6, height=1, command=
82             Absorcion).place(x=20+x1, y=410-y3)
83
84
85
86         #----- Funcion limpiar pantalla
87         -----
87             def Reset():
88                 tkMessageBox.showinfo("Infrarossi", message= "! Limpiando, por favor
89                 espere i")
89                 os.system("rm MAC.txt MACD.txt puerto.txt conexion.txt direccion.txt
90                 datos/dats1.dat")
90                 os.system("sh bin/d_Blu.sh &")
91                 os.system("python infrarossi.py &")
92                 os.system("rm MAC.txt MACD.txt puerto.txt")
93                 exit()
94
95         #----- Funciones menu
96         -----

```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/infrarossi.py
Página 3 de 4                                         mar 05 abr 2016 12:14:29 COT

96     def Difraccion():
97         tkMessageBox.showinfo("Infrarossi", message= "Espere por favor,
98                               Preparando todo para empezar con el experimento.")
99         os.system("python bin/Difraccion.py &")
100        # os.system("python bin/estadis2.py &")
101
102    def Atenuacion():
103        tkMessageBox.showinfo("Infrarossi", message= "Espere por favor,
104                               Preparando todo para empezar con el experimento.")
105        os.system("python bin/Atenuacion.py &")
106        # os.system("xterm -T Infrarossi -geom 50x8+185+100 +cm -bg blue -e
107          python bin/Atenuacion.py & ")
108
109    def Absorcion():
110        tkMessageBox.showinfo("Infrarossi", message= "Espere por favor,
111                               Preparando todo para empezar con el experimento.")
112        os.system("python bin/Absorcion.py & ")
113
114    #----- Bluetooth desconectado
115    -----
116
117    def Bl_off():
118        tkMessageBox.showinfo("Infrarossi", message= "!Bluetooth desconectado!")
119        os.system("rm MAC.txt MACD.txt puerto.txt conexion.txt direccion.txt")
120        os.system("sh bin/d_Blu.sh &")
121        os.system("python infrarossi.py &")
122        os.system("rm MAC.txt MACD.txt puerto.txt")
123        exit()
124
125    # ----- Definiendo Funcion firmware
126    -----
127
128    def Firmware():
129        tkMessageBox.showinfo("Infrarossi", message= "Conecte la tarjeta
130                               microcontroladora arduino uno, con un microcontrolador listo para su
131                               uso.\n\nProcediendo con el instalador del firmware")
132        os.system("python bin/firmware/G_firmware.py & ")
133
134    #-----Definiendo función
135    Documentación-----
136    def Documentacion():
137        tkMessageBox.showinfo("Infrarossi", message= " ! Abriendo
138                               documentación, tenga pascuencia i ")
139        os.system("xdg-open 'Montaje/Articulo_montaje_infrarossi.pdf' & ")
140
141
142    #----- Botones Bluetooth
143    -----
144
145    y1 = int(300)
146    lblBlue = Label(bicho, text="BLUETOOTH ", fg = ("black"), bg = ("white"),
147                    font = ("Century Schoolbook L",10)).place(x=10, y=320-y1)

```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/infrarossi.py
Página 4 de 4
mar 05 abr 2016 12:14:29 COT

139     btnConectar= Button(bicho, text= " ON ", width=5, height=1, command= Verifica).place(x=20, y=350-y1)
140 #     btnConectar= Button(bicho, text= " ON ", width=5, height=1, command= Valido).place(x=20, y=350-y1)
141     btnDesconectar= Button(bicho, text= " OFF ", width=5, height=1, command= Bl_off).place(x=20, y=380-y1)
142     btnSalir=Button(bicho, text = "Salir", command=Salir, height=1, width =5). place(x=20, y=410-y1)
143
144
145 #----- Botones firmware -----
146 y2 = int(270)
147 btnfirmware=Button(bicho, text = "Firmware", command=Firmware, height=1, width =5).place(x=340, y=380-y2)
148 btnDocumentacion=Button(bicho, text = "Ayuda", command=Documentacion, height=1 , width =5).place(x=340, y=350-y2)
149     btnLimpiar = Button(bicho, text= "limpiar", width=5, height=1, command= Reset).place(x=340, y=320-y2)
150     bicho.mainloop()
151
152
153 def __init__(self):
154     self.Modulo()
155     self.__del__()
156
157 def __del__(self):
158     print ("PROGRAMA TERMINADO")
159
160 if __name__ == "__main__":
161     App()
162
163

```

Apéndice E8. infrarossi

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/infrarossi      mar 05 abr 2016 12:23:54 COT
Página 1 de 1

1 #!/usr/bin/python
2 # -*- coding:utf-8 -*-
3 # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4 # los términos de la licencia pública general de GNU, según es publicada
5 # por la free software fundation bien la versión 3 de la misma licencia
6 # o de cualquier versión posterior. (según su elección).
7 # Si usted hace alguna modificación en esta aplicación, deberá siempre
8 # mencionar el autor original de la misma.
9 # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Diego Alberto Parra Garzón
13 # Dr Julian Andres Salamanca Bernal
14 # Colombia, Bogota D.C.
15
16 import os
17 import time
18
19
20 os.system("sudo python ~/Documentos/Free-infrarossi/free_infrarossi/infrarossi.py")
21
22
```

Apéndice E9. c_Blue.sh

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/c_Blu.sh  
Página 1 de 1  
mar 05 abr 2016 12:24:32 COT  
1 #!/bin/bash  
2 #/etc/init.d/bluetooth restart  
3 # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo  
4 # los términos de la licencia pública general de GNU, según es publicada  
5 # por la free software fundation bien la versión 3 de la misma licencia  
6 # o de cualquier versión posterior. (según su elección ).  
7 # Si usted hace alguna modificación en esta aplicación, deberá siempre  
8 # mencionar el autor original de la misma.  
9 # Autor:  
10 # Universidad Distrital Francisco Jose  
11 # Grupo de fisica e informatica  
12 # Diego Alberto Parra Garzón  
13 # Dr Julian Andres Salamanca Bernal  
14 # Colombia, Bogota D.C.  
15 rfcomm release rfcomm0  
16 hcitool dev > log.txt | grep -e 'hci0' log.txt > mac.txt  
17 cut -d "0" -f 2,3 mac.txt > MAC.txt  
18 MAC=`cat MAC.txt`  
19 echo "La mac del dispositivo es : $MAC"  
20 hcitool scan > dispo.txt | grep -e "HC" dispo.txt > macd.txt  
21 cut -d "H" -f 1 macd.txt > MACD.txt  
22 #/etc/init.d/bluetooth status  
23 #/etc/init.d/bluetooth force-reload  
24 #/etc/init.d/bluetooth reset  
25 rm log.txt mac.txt dispo.txt macd.txt  
26 MACD=`cat MACD.txt`  
27 echo "dispositivo HC-05 encontrado MAC: $MACD"  
28 rfcomm connect -i $MACD  
29  
30  
31
```

Apéndice E10. g_p_Abs.py

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/g_p_Abs.py
Página 1 de 3                                         mar 05 abr 2016 12:25:12 COT

1  #!/usr/bin/python
2  #!**-* coding:utf-8 **-
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Dr Julian Andres Salamanca Bernal
13 # Diego Alberto Parra Garzón
14 # Colombia, Bogota D.C.

15
16 import numpy as np
17 import os
18 import serial
19 import subprocess
20 import math
21 import time
22 import Gnuplot
23 import shutil
24 import matplotlib.pyplot as pl
25 class App:
26     def Analisis(self):
27         # os.system("octave bin/estadistica1.m")
28         os.system("octave estadistica2.m")
29         self.BX = np.loadtxt("datos/BX.dat")
30         self.BY = np.loadtxt("datos/BY.dat")
31         self.error_CM1 = np.loadtxt("datos/ECM1.dat")
32         self.q1 = np.loadtxt("datos/q1.dat")
33         self.Amplitud = np.loadtxt("datos/Amplitud.dat")
34         self.Ampli = round(self.Amplitud, 6)
35         self.T = np.loadtxt("datos/Transmitancia.dat")
36         self.A = np.loadtxt("datos/Vteo.dat")
37         self.V_A = np.loadtxt("datos/V_aproximado.dat")
38         f = np.loadtxt("datos/F.dat")
39         a = np.loadtxt("datos/a.dat")
40         b = np.loadtxt("datos/b.dat")
41         self.Yest = np.loadtxt("datos/Yest.dat")
42         self.Yteo = np.loadtxt("datos/Yteo.dat")
43         self.ECM = np.loadtxt("datos/ECM.dat")
44         self.error_CM11 = round(self.error_CM1, 11)
45         self.X = f[:, 0]
46         self.V = f[:, 1]
47         self.a = round(a, 6)
48         self.b = round(b, 3)
49         print self.X, self.V, "\n\n\n", self.a, " X^(" ,self.b, ")   "
50
51     def Grafica(self):
52         pl.subplot(221)
53         pl.title('Distancia vs Intensidad \n')
54         pl.xlabel('Distancia [m]')
55         pl.ylabel('Intensidad [W]')

```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/g_p_Abs.py      mar 05 abr 2016 12:25:12 COT
Página 2 de 3

56     pl.plot(self.X, self.V, 'g:o')
57     pl.axis([0, 0.65, -0.001, 0.022])
58     pl.text(0.2, 0.019, r'Datos recolectados')
59
60     def Grafical(self):
61         pl.subplots_adjust(left=0.13)
62         pl.subplots_adjust(bottom=0.13)
63         pl.subplots_adjust(right=0.97)
64         pl.subplots_adjust(top=0.87)
65         pl.subplots_adjust(wspace=0.37)
66         pl.subplots_adjust(hspace=0.68)
67
68     def Grafica2(self):
69         pl.subplot(222)
70         pl.title('APROXIMACION')
71         pl.xlabel('Distancia [m]')
72         pl.ylabel('Intensidad [W]')
73         pl.plot(self.X, self.A, 'B', self.q1, self.BY, 'y', self.q1, self.BX, 'b')
74         pl.plot(self.X, self.V_A, 'Y', self.X, self.V, 'g:o')
75         pl.axis([0, 0.65, -0.001, 0.022])
76         pl.text(0.15, 0.015, r' ' + str(0.000259) + ' + X^(' + str(-2) + ')')
77         pl.text(0.3, 0.017, r'Azul')
78         pl.text(0.2, 0.010, r' Amarillo')
79         pl.text(0.19, 0.008, r' ' + str(self.Ampli) + ' X^(-2)')
80
81
82     def Grafica3(self):
83         pl.subplot(212)
84         pl.plot(self.X, self.V, 'g:o', self.X, self.V_A, 'Y', self.X, self.A, 'B', self.q1, self.BY, 'y', self.q1, self.BX, 'b')
85         pl.subplot(212)
86         pl.xlabel('Distancia [m]')
87         pl.ylabel('Intensidad [W]')
88         pl.title('Distancia vs Intensidad irradiada \n')
89         pl.legend(loc='upper left')
90         pl.axis([0, 0.65, -0.001, 0.022])
91         pl.text(0.3, 0.019, r' ideal = ' + str(0.000259) + ' + X^(' + str(-2) + ')')
92         pl.text(0.3, 0.016, r' Exp = ' + str(self.Ampli) + ' + X^(-2)')
93         pl.text(0.12, 3.5, r'V1')
94         pl.text(0.01, 2, r'V2*')
95         pl.text(0.3, 0.013, r'Fp = ' + str(round(self.T, 4)))
96         pl.text(0.3, 0.010, r'ECM = ' + str(round(self.error_CM1, 11)))
97         pl.savefig('datos/Absolucion.png')
98         pl.show()
99
100    def Ordenar(self):
101        os.system("python bin/o_Carpetas2.py")
102
103    def __init__(self):
104        self.Analisis()
105        self.Grafica()
106        self.Grafical()
107        self.Grafica2()
108        self.Grafica3()
109        self.Ordenar()

```

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/g_p_Abs.py  
Página 3 de 3
```

```
mar 05 abr 2016 12:25:12 COT
```

```
110         self.__del__()  
111  
112     def __del__(self):  
113         print ("PROGRAMA TERMINADO")  
114  
115  
116     Iniciar = App()  
117
```

Apéndice E11. g_p_Ate.py

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/g_p_Ate.py
Página 1 de 2                                         mar 05 abr 2016 12:30:12 COT

1  #!/usr/bin/python
2  #!**-* coding:utf-8 **-
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Dr Julian Andres Salamanca Bernal
13 # Diego Alberto Parra Garzón
14 # Colombia, Bogota D.C.
15 import numpy as np
16 import os
17 import serial
18 import subprocess
19 import math
20 import time
21 import Gnuplot
22 import shutil
23 import matplotlib.pyplot as pl
24 class App:
25     def Analisis(self):
26         # os.system("octave datos/estadistical.m")
27         os.system("octave estadistical.m")
28         f = np.loadtxt("datos/F.dat")
29         a = np.loadtxt("datos/a.dat")
30         b = np.loadtxt("datos/b.dat")
31         self.Yest = np.loadtxt("datos/Yest.dat")
32         self.Yteo = np.loadtxt("datos/Yteo.dat")
33         self.ECM = np.loadtxt("datos/ECM.dat")
34         self.ECM1 = round(self.ECM, 11)
35         self.X = f[:,0]
36         self.V = f[:,1]
37         self.a = round(a, 6)
38         self.b = round(b, 3)
39         self.error = ((-2*float(self.b))**2)**0.5
40         print self.X, self.V, "\n\n", self.a, " X^(", self.b, ")"
41
42     def Grafica(self):
43         pl.subplot(221)
44             pl.title('Distancia vs Intensidad \n')
45             pl.xlabel('Distancia [m]')
46             pl.ylabel('Intensidad [W]')
47             pl.plot(self.X, self.V, 'o --')
48             pl.axis([0, 0.65, -0.001, 0.022])
49             pl.text(0.2, 0.019, r'Datos recolectados')
50
51
52
53     def Grafical(self):
54         pl.subplots_adjust(left=0.13)
55         pl.subplots_adjust(bottom=0.13)
```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/g_p_Ate.py
Página 2 de 2                                         mar 05 abr 2016 12:30:12 COT

56     pl.subplots_adjust(right=0.97)
57     pl.subplots_adjust(top=0.87)
58     pl.subplots_adjust(wspace=0.37)
59     pl.subplots_adjust(hspace=0.68)
60
61
62     def Grafica2(self):
63         pl.subplot(222)
64         pl.title('APROXIMACION')
65         pl.xlabel('Distancia [m]')
66         pl.ylabel('Intensidad [W]')
67         pl.plot(self.X, self.V, 'B')
68         pl.plot(self.X, self.Yest, 'R')
69         pl.axis([0, 0.65, -0.001, 0.020])
70         pl.text(0.3, 0.017, r'Azul')
71         pl.text(0.15, 0.015, r'+' + "(" + str(self.a) + ")*X^(+" + str(self.b)+")')
72         pl.text(0.2, 0.010, r'Rojo')
73         pl.text(0.19, 0.008, r'+' + "(" + str(self.a) + ")*X^(-2)')
74
75
76     def Grafica3(self):
77         pl.subplot(212)
78         pl.plot(self.X, self.V, 'K')
79         pl.plot(self.X, self.Yteo, 'R')
80         pl.plot(self.X, self.Yest, 'B')
81         pl.subplot(212)
82         pl.xlabel('Distancia [m]')
83         pl.ylabel('Intensidad [W]')
84         pl.title('Distancia vs Intensidad irradiada \n')
85         pl.legend(loc='upper left')
86         pl.axis([0, 0.65, -0.001, 0.022])
87         pl.text(0.3, 0.02, r'Azul I_r=' + str(self.a) + '*X^(+' + str(self.b)+")")
88         pl.text(0.3, 0.017, r'Rojo I_r=' + str(self.a) + '*X^(-2)')
89         pl.text(0.3, 0.014, r'ECM = ' + str(self.ECM1))
90         pl.text(0.3, 0.011, r'E_exp = ' + str(self.error))
91         pl.savefig('datos/Atenuacion.png')
92         pl.show()
93
94     def Ordenar(self):
95         os.system("python bin/o_Carpetas.py")
96
97     def __init__(self):
98         self.Analisis()
99         self.Grafica()
100        self.Grafical()
101        self.Grafica2()
102        self.Grafica3()
103        self.Ordenar()
104        self.__del__()
105
106    def __del__(self):
107        print ("PROGRAMA TERMINADO")
108
109
110 Iniciar = App()

```

Apéndice E12. Estadistica1.m

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/estadistica1.m
Página 1 de 2
mar 05 abr 2016 12:30:43 COT

1 #!/usr/bin/python
2 #!**-* coding:utf-8 *-*
3 # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4 # los términos de la licencia pública general de GNU, según es publicada
5 # por la free software fundation bien la versión 3 de la misma licencia
6 # o de cualquier versión posterior. (según su elección).
7 # Si usted hace alguna modificación en esta aplicación, deberá siempre
8 # mencionar el autor original de la misma.
9 # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Dr Julian Andres Salamanca Bernal
13 # Diego Alberto Parra Garzón
14 # Colombia, Bogota D.C.
15
16 f = load('datos/dats1.dat'); #-----entrada de datos a través del archivo .dat
17 X = f(:,1)/100 #-----datos de distancia en metros
18 Y = f(:,2)/1000000 #-----datos de intensidad en vatios
19 ff = [X, Y] # -----voltaje como función X e Y
20 U = log(X); # -----logaritmo natural de la distancia
21 V = log(Y); #----- logaritmo natural del voltaje
22 U2 = U .* U; #----- el cuadrado del logaritmo de la distancia
23 UV = U .* V; # .....logaritmo del producto del voltaje y la distancia
24 z = size(f); #----- tamaño de filas y columnas del archivo .dat
25 z1 = z(:,1); #----- tamaño de columnas del archivo .dat
26 sumX = sum(X); #----- sumatoria de todos los datos de la distancia en metros
27 sumY = sum(Y); #----- sumatoria de todos los datos del voltaje en voltios
28 sumU = sum(U); #----- sumatoria de todos los datos del logaritmo de la distancia
29 sumV = sum(V); #----- sumatoria de todos los datos del logaritmo del voltaje
30 sumU2 = sum(U2); #----- sumatoria de todos los datos del cuadrado del logaritmo de la distancia
31 sumUV = sum(UV); #----- sumatoria de todos los datos del producto de el logaritmo del voltaje y la distancia
32 promX = sumX/z1; #----- el promedio de la distancia
33 promY = sumY/z1; #----- el promedio del voltaje
34 promU = sumU/z1; #----- promedio del logaritmo de la distancia
35 promV = sumV/z1; #----- promedio del logaritmo del voltaje
36 promU2 = sumU2/z1; #----- promedio del logaritmo la distancia
37 promUV = sumUV/z1; #----- promedio del cociente del logaritmo de la distancia y el voltaje.
38 #-----Primer calculo-----
39 Suv = promUV - promU*promV;
40 Su2 = promU2 - promU*promU;
41 #b = (Suv / Su2) - 0.5
42 b = (Suv / Su2)
43 A = (promV - b*promU)
44 #a = exp(A)/5 + 0.02
45 a = exp(A)
46 Yest = a* (X .^ b);

```

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/estadistica1.m
Página 2 de 2
mar 05 abr 2016 12:30:43 COT
```

```
47 Yteo= a * (X .^(-2));
48 error = Y .- Yest;
49 ECM = sum(error .^(2)) /z1
50 save -ascii 'datos/a.dat' a;
51 save -ascii 'datos/F.dat' ff;
52 save -ascii 'datos/b.dat' b;
53 save -ascii 'datos/ECM.dat' ECM;
54 save -ascii 'datos/Yest.dat' Yest;
55 save -ascii 'datos/Yteo.dat' Yteo;
56 #plot (X, Y, 'o')
57 #pause
58
59
```

Apéndice E13. Estadistica2.m

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/estadistica2.m
Página 1 de 2
mar 05 abr 2016 12:31:00 COT

1 #!/usr/bin/python
2 #!**-* coding:utf-8 **-
3 # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4 # los términos de la licencia pública general de GNU, según es publicada
5 # por la free software fundation bien la versión 3 de la misma licencia
6 # o de cualquier versión posterior. (según su elección).
7 # Si usted hace alguna modificación en esta aplicación, deberá siempre
8 # mencionar el autor original de la misma.
9 # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Dr Julian Andres Salamanca Bernal
13 # Diego Alberto Parra Garzón
14 # Colombia, Bogota D.C.

15
16
17 f = load('datos/dats1.dat'); #-----entrada de datos a través del archivo .dat
18 X = f(:,1)/100 #-----datos de distancia en metros
19 Y = f(:,2)/1000000 #-----datos de voltaje en voltios
20 ff = [X, Y]; # -----voltaje como función X e Y
21 U = log(X); # -----logaritmo natural de la distancia
22 V = log(Y); #----- logaritmo natural del voltaje
23 U2 = U .* U; #----- el cuadrado del logaritmo de la distancia
24 UV = U .* V; # .....logaritmo del producto del voltaje y la distancia
25 z = size(f); #----- tamaño de filas y columnas del archivo .dat
26 z1 = z(:,1); #----- tamaño de columnas del archivo .dat
27 sumX = sum(X); #----- sumatoria de todos los datos de la distancia en metros
28 sumY = sum(Y); #----- sumatoria de todos los datos del voltaje en voltios
29 sumU = sum(U); #----- sumatoria de todos los datos del logaritmo de la distancia
30 sumV = sum(V); #----- sumatoria de todos los datos del logaritmo del voltaje
31 sumU2 = sum(U2); #----- sumatoria de todos los datos del cuadrado del logaritmo de la distancia
32 sumUV = sum(UV); #----- sumatoria de todos los datos del producto de el logaritmo del voltaje y la distancia
33 promX = sumX/z1; #----- el promedio de la distancia
34 promY = sumY/z1; #----- el promedio del voltaje
35 promU = sumU/z1; #----- promedio del logaritmo de la distancia
36 promV = sumV/z1; #----- promedio del logaritmo del voltaje
37 promU2 = sumU2/z1; #----- promedio del logaritmo la distancia
38 promUV = sumUV/z1; #----- promedio del cociente del logaritmo de la distancia y el voltaje.
39 #-----Primer calculo-----
40 Suv = promUV - promU*promV;
41 Su2 = promU2 - promU*promU;
42 #b = (Suv / Su2) - 0.5
43 b = (Suv / Su2)
44 A = (promV - b*promU)
45 #a = exp(A)/5 + 0.02
46 a = exp(A)

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/estadistica2.m
Página 2 de 2

mar 05 abr 2016 12:31:00 COT

```

47 Yest = a* (X .^ b);
48 Yteo= a * (X .^(-2));
49 error = Y .- Yest;
50 ECM = sum(error .^(2)) /z1
51 save -ascii 'datos/a.dat' a;
52 save -ascii 'datos/F.dat' ff;
53 save -ascii 'datos/b.dat' b;
54 save -ascii 'datos/ECM.dat' ECM;
55 save -ascii 'datos/Yest.dat' Yest;
56 #-----segundo calculo-----
57 Vteo = 0.000259 * (X .^(-2))
58 q1 = [0.01:0.001:5];
59 Pprueba = (Y .* Y)/125 #----- Potencia en el diodo
60 Voltaje_entrante = 100*(Y ./ Vteo)
61 Pteori = (Vteo .* Vteo)/125
62 DV = Vteo .- Y;
63 T = DV ./ Vteo;
64 sT = sum(T);
65 Tp = sT / z1;
66 Trasmitancia = Tp
67 Reflectancia = 1 - Tp
68 EEp= (Reflectancia)*(0.000259);
69 densidad_voltaje = EEp
70 Voltaje_aproximado = (EEp)*(X .^(-2));
71 BX = (0.000259)*(q1 .^(-2));
72 BY = EEp * (q1 .^(-2));
73 V_estimado = EEp*(X .^(-2));
74 error1 = Y .- V_estimado;
75 ECM1 = sum(error1 .^(2)/z1)
76 save -ascii 'datos/Yteo.dat' Yteo;
77 save -ascii 'datos/Vteo.dat' Vteo;
78 save -ascii 'datos/Amplitud.dat' EEp;
79 save -ascii 'datos/Transmitancia.dat' Tp;
80 save -ascii 'datos/V_aproximado.dat' Voltaje_aproximado;
81 save -ascii 'datos/q1.dat' q1;
82 save -ascii 'datos/BX.dat' BX;
83 save -ascii 'datos/BY.dat' BY;
84 save -ascii 'datos/ECM1.dat' ECM1;
85
86 #plot (q1, BX, '-' , q1, BY, 'o-' , X, Voltaje_aproximado, "-")
87 #pause
88
89

```

Apéndice E14. Atenuacion.py

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/Atenuacion.py
Página 1 de 2                                         mar 05 abr 2016 13:13:41 COT

1  #!/usr/bin/python
2  #!**-* coding:utf-8 *-*
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Diego Alberto Parra Garzón
13 # Dr Julian Andres Salamanca Bernal
14 # Colombia, Bogota D.C.
15 import serial
16 import os
17 import subprocess
18 import math
19 import time
20 import Gnuplot
21 from Tkinter import *
22 import tkMessageBox
23 import Tkinter
24 import shutil
25
26 class Gramo():
27     def Atenua(self):
28         bicho = Tk()
29         bicho.geometry("280x170+200+90")
30         bicho.config(bg="white")
31         bicho.title("Infrarossi")
32         bicho.resizable(width=0, height=0)
33
34     def Verifica():
35         print "ola"
36
37     def Salir():
38         tkMessageBox.showinfo("Infrarossi", message= "Saliendo . . . ")
39         arduino = serial.Serial("/dev/rfcomm0", 9600)
40         arduino.write('aa')
41         exit()
42         exit()
43
44     def Grafica():
45         os.system("python g_p_Ate.py &")
46
47     def Comenzar1():
48         tkMessageBox.showinfo("Infrarossi", message= "Se procede a capturar
49         datos, para detener el proceso cierre la ventana de captura de datos
50         'de color azul'")
51         os.system("xterm -T Infrarossi -geom 50x8+185+100 +cm -bg blue -e python
52         bin/c_p_Ate.py &")
53         # os.system("python bin/c_p_Ate.py")
54
55     # -----CONFIGURACION DE VENTANA

```

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/Atenuacion.py
Página 2 de 2
mar 05 abr 2016 13:13:41 COT

53     X=8
54     Y=10
55     lblTitulo = Label(bicho, text="ATENUACION", fg = ("blue"), bg = ("white"),
56                         font = ("Century Schoolbook L",23)).place(x=30, y=20)
57     btnConectar1 = Button(bicho, text= " INICIAR ", width=5, height=1, command =
58                           = Comenzar1).place(x=20+X, y=100+Y)
59     btnSalir = Button(bicho, text= " SALIR ", width=5, height=1, command=
60                          Salir).place(x=170+X, y=100+Y)
61     btnGrafica = Button(bicho, text= " GRAFICA ", width=5, height=1, command=
62                           Grafica).place(x=95+X, y=100+Y)
63     Verifica()
64     bicho.mainloop()
65
66
67     def __init__(self):
68         self.Atenua()
69         self.__del__()
70
71
72     modulo  = Gramo()
73
```

Apéndice E15. Absorcion.py

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/Absorcion.py
Página 1 de 2
                                         mar 05 abr 2016 13:14:18 COT

1  #!/usr/bin/python
2  #!**-* coding:utf-8 *-*
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Dr Julian Andres Salamanca Bernal
13 # Diego Alberto Parra Garzón
14 # Colombia, Bogota D.C.
15 import serial
16 import os
17 import subprocess
18 import math
19 import time
20 import Gnuplot
21 from Tkinter import *
22 import tkMessageBox
23 import Tkinter
24 import shutil
25
26 class Gramo():
27     def Absor(self):
28         bicho = Tk()
29         bicho.geometry("280x170+200+90")
30         bicho.config(bg="white")
31         bicho.title("Infrarossi")
32         bicho.resizable(width=0, height=0)
33
34     def Verifica():
35         print "ola"
36
37     def Salir():
38         tkMessageBox.showinfo("Infrarossi", message= "Saliendo . . . ")
39         arduino = serial.Serial("/dev/rfcomm0", 9600)
40         arduino.write('aa')
41         exit()
42         exit()
43
44     def Comenzar1():
45         tkMessageBox.showinfo("Infrarossi", message= "Se procede a capturar      ↵
46             datos, para detener el proceso cierre la ventana de captura de datos      ↵
47             'de color azul'")
48         os.system("xterm -T Infrarossi -geom 50x8+185+100 +cm -bg blue -e python      ↵
49             bin/c_p_Abs.py &")
50         # os.system("python bin/c_p_Ate.py")
51
52     def Grafica():
53         os.system("python g_p_Abs.py &")
54
55     # -----CONFIGURACION DE VENTANA      ↵

```

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/Absorcion.py
Página 2 de 2                                         mar 05 abr 2016 13:14:18 COT
```

```
53         X=10
54         Y=10
55         lblTitulo = Label(bicho, text="ABSORCION", fg = ("blue"), bg = ("white"), font =
56             = ("Century Schoolbook L",23)).place(x=30, y=20)
57         btnConectar1 = Button(bicho, text= " INICIAR ", width=5, height=1, command=
58             Comenzar1).place(x=20+X, y=100+Y)
59         btnSalir = Button(bicho, text= " SALIR ", width=5, height=1, command=
60             Salir).place(x=170+X, y=100+Y)
61         btnGrafica = Button(bicho, text= " GRAFICA ", width=5, height=1, command=
62             Grafica).place(x=95+X, y=100+Y)
63     Verifica()
64     bicho.mainloop()
65
66
67     def __init__(self):
68         self.Absor()
69         self.__del__()
70
71
72     modulo = Gramo()
73
```

Apéndice E16. Difraccion.py

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/Difraccion.py
Página 1 de 2                                         mar 05 abr 2016 13:19:05 COT

1  #!/usr/bin/python
2  #!**-* coding:utf-8 *-*
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Diego Alberto Parra Garzón
13 # Dr Julian Andres Salamanca Bernal
14 # Colombia, Bogota D.C.
15 import serial
16 import os
17 import subprocess
18 import math
19 import time
20 import Gnuplot
21 from Tkinter import *
22 import tkMessageBox
23 import Tkinter
24 import shutil
25
26 class App():
27     def Difrac(self):
28         bicho = Tk()
29         bicho.geometry("280x170+200+90")
30         bicho.config(bg="white")
31         bicho.title("Infrarossi")
32         bicho.resizable(width=0, height=0)
33
34     def Salir():
35         os.system('rfcomm -a > conexion.txt | cut -d ":" -f 1,1 > direccion.txt')
36         puerto = open('direccion.txt', 'r')
37         self.puerto = puerto.read(7)
38         puerto.close()
39         self.puerta = "/dev/" + self.puerto
40         print self.puerta
41         tkMessageBox.showinfo("Infrarossi", message= "Saliendo . . . ")
42         arduino = serial.Serial(self.puerta, 9600)
43         arduino.write('aa')
44         exit()
45         exit()
46
47     def Comenzar1():
48         tkMessageBox.showinfo("Infrarossi", message= "Se procede a capturar
49         datos, para detener el proceso cierre la ventana captura de datos
50         'color azul'")
51         os.system("xterm -T Infrarossi -geom 50x8+185+100 +cm -bg blue -e python
52         bin/c_p_Dif.py &")
53
54     # -----CONFIGURACION DE VENTANA

```

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/Difraccion.py
Página 2 de 2
                                         mar 05 abr 2016 13:19:05 COT

52         X=10
53         Y=10
54         lblTitulo = Label(bicho, text="DIFRACCION", fg = ("blue"), bg = ("white"),
55                               font = ("Century Schoolbook L",23)).place(x=30, y=20)
56         btnConectar1 = Button(bicho, text= " INICIAR ", width=5, height=1, command =
57                               = Comenzar1).place(x=20+X, y=100+Y)
58         btnSalir = Button(bicho, text= " SALIR ", width=5, height=1, command=
59                               Salir).place(x=160+X, y=100+Y)
60         bicho.mainloop()
61
62
63
64     def __init__(self):
65         self.Difrac()
66         self.__del__()
67
68
69     modulo = App()
70
```

Apéndice E17. c_p_Abs.py

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/c_p_Abs.py
Página 1 de 2                                         mar 05 abr 2016 13:19:21 COT

1  #!/usr/bin/python
2  #!**-* coding:utf-8 **-
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Diego Alberto Parra Garzón
13 # Dr Julian Andres Salamanca Bernal
14 # Colombia, Bogota D.C.

15
16 import numpy as np
17 import os
18 import serial
19 import subprocess
20 import math
21 import time
22 import Gnuplot
23 import shutil
24 import matplotlib.pyplot as pl
25 class App:
26     def Verifica(self):
27         os.system('rfcomm -a > conexion.txt | cut -d ":" -f 1,1 > direccion.txt')
28         puerto = open('direccion.txt', 'r')
29         self.puerto = puerto.read()
30         puerto.close()
31         self.puerta = "/dev/" + self.puerto
32         print self.puerta
33         try:
34             arduino = serial.Serial(self.puerta, 9600)
35             arduino.write("aa")
36         except:
37             os.system("exit")
38
39     def Salir(self):
40         exit()
41
42     def Comenzar(self):
43         # Distancia de separación 28 cm
44         # Distancia de recolección 25 cm
45         # Paso en centímetros pausada 1 = 0,34
46         # Paso en centímetros pausada 2 = 0,30
47         # Paso en centímetros pausada 3 = 0,27
48         # Paso en centímetros pausada 4 = 0,245
49         # Paso en centímetros pausada 5 = 0,213
50         for n in range (0, 117):
51             os.system('rm datos/dat.dat')
52             #time.sleep(2)
53             arduino= serial.Serial(self.puerta, 9600)
54             print "aca va la lectura"

```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/c_p_Abs.py
Página 2 de 2                                         mar 05 abr 2016 13:19:21 COT

55     arduino.write("aa")
56     #time.sleep(1)
57     arduino.write('4')
58     arduino.close()
59     arduino=serial.Serial(self.puerta, 9600)
60     time.sleep(2)
61     arduino.write('zz')
62     for i in range(0, 140):
63         arduino=serial.Serial(self.puerta, 9600)
64         archi = open('datos/dat.dat', 'a+')
65         time.sleep(0.00005)
66         x = arduino.readline()
67         z = 0.21367*2*(140 - n)
68         xo = str(z)
69         yo = str(x)
70         print "paso numero", n
71         print "(cm) \t (microw)"
72         print('{0} {1}'.format(xo, yo))
73         archi.write (xo)
74         archi.write (" ")
75         archi.write (yo)
76         archi.close()
77     else:
78         os.system("octave bin/prom1.m")
79         archi = open('datos/prom.dat', 'a+')
80         print("aca va la pausa")
81         Lectura = archi.read()
82         archi.close()
83         archil = open('datos/dats1.dat', 'a+')
84         archil.write(Lectura)
85         archil.close()
86         arduino.write('aa')
87         arduino.close()
88
89     def Grafw(self):
90         os.system("python g_p_Abs.py")
91
92     def Ordenar(self):
93         os.system("python bin/o_Carpetas2.py")
94
95
96
97     def __init__(self):
98         self.Verifica()
99         self.Comenzar()
100        self.Grafw()
101        self.Ordenar()
102        self.__del__()
103
104    def __del__(self):
105        print ("PROGRAMA TERMINADO")
106
107
108    Iniciar = App()
109

```

Apéndice E18. c_p_Dif.py

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/c_p_Dif.py
Página 1 de 2                                         mar 05 abr 2016 13:19:55 COT

1  #!/usr/bin/python
2  #!**-* coding:utf-8 **-
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Diego Alberto Parra Garzón
13 # Dr Julian Andres Salamanca Bernal
14 # Colombia, Bogota D.C.
15 import numpy as np
16 import os
17 import serial
18 import subprocess
19 import math
20 import time
21 import Gnuplot
22 import shutil
23 import matplotlib.pyplot as pl
24 class App:
25     def Verifica(self):
26         os.system('rfcomm -a > conexion.txt | cut -d ":" -f 1,1 > direccion.txt')
27         puerto = open('direccion.txt', 'r')
28         self.puerto = puerto.read(7)
29         puerto.close()
30         self.puerta = "/dev/" + self.puerto
31         print self.puerta
32         try:
33             arduino = serial.Serial(self.puerta, 9600)
34             arduino.write("aa")
35         except:
36             os.system("exit")
37
38     def Salir(self):
39         exit()
40
41     def Comenzar(self):
42         # Distancia de separación 28 cm
43         # Distancia de recolección 25 cm
44         #Paso en centímetros pausada 1 = 0,34
45         #Paso en centímetros pausada 2 = 0,30
46         #Paso en centímetros pausada 3 = 0,27
47         #Paso en centímetros pausada 4 = 0,245
48         #Paso en centímetros pausada 5 = 0,213
49         for n in range (0, 94):
50             os.system('rm datos/dat.dat')
51             #time.sleep(2)
52             arduino= serial.Serial(self.puerta, 9600)
53             print "aca va la lectura"
54             arduino.write("aa")

```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/c_p_Dif.py
Página 2 de 2                                         mar 05 abr 2016 13:19:55 COT

55     #time.sleep(1)
56     arduino.write('4')
57     arduino.close()
58     arduino=serial.Serial(self.puerta, 9600)
59     time.sleep(2)
60     arduino.write('hh')
61     for i in range(0, 140):
62         arduino=serial.Serial(self.puerta, 9600)
63         archi = open('datos/dat.dat', 'a+')
64         time.sleep(0.00005)
65         x = arduino.readline()
66         z = 0.21367*n
67         xo = str(z)
68         yo = str(x)
69         print "paso numero", n
70         print "(cm) \t (microw)"
71         print('{0} {1}'.format(xo, yo))
72         archi.write (xo)
73         archi.write (" ")
74         archi.write (yo)
75         archi.close()
76     else:
77         os.system("octave bin/prom1.m")
78         archi = open('datos/prom.dat', 'a+')
79         print("aca va la pausa")
80         Lectura = archi.read()
81         archi.close()
82         archil = open('datos/dats1.dat', 'a+')
83         archil.write(Lectura)
84         archil.close()
85         arduino.write('aa')
86         arduino.close()
87
88     def Analisis(self):
89     # os.system("python bin/Estadistica2.py")
90     os.system("python bin/estadis2.py")
91     time.sleep(3)
92
93
94     def Ordenar(self):
95         os.system("python bin/o_Carpetas1.py")
96
97
98
99     def __init__(self):
100        self.Verifica()
101        self.Comenzar()
102        self.Analisis()
103        self.__del__()
104
105    def __del__(self):
106        print ("PROGRAMA TERMINADO")
107
108
109 Iniciar = App()

```

Apéndice E19. *c_p_Ate.py*

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/c_p_Ate.py
Página 1 de 2                                         mar 05 abr 2016 13:19:33 COT

1  #!/usr/bin/python
2  #!**-* coding:utf-8 **-
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Diego Alberto Parra Garzón
13 # Dr Julian Andres Salamanca Bernal
14 # Colombia, Bogota D.C.
15 import numpy as np
16 import os
17 import serial
18 import subprocess
19 import math
20 import time
21 import Gnuplot
22 import shutil
23 import matplotlib.pyplot as pl
24 class App:
25     def Verifica(self):
26         os.system('rfcomm -a > conexion.txt | cut -d ":" -f 1,1 > direccion.txt')
27         puerto = open('direccion.txt', 'r')
28         self.puerto = puerto.read(7)
29         puerto.close()
30         self.puerta = "/dev/" + self.puerto
31         print self.puerta
32         try:
33             arduino = serial.Serial(self.puerta, 9600)
34             arduino.write("aa")
35         except:
36             os.system("exit")
37
38     def Salir(self):
39         exit()
40
41     def Salir(self):
42         exit()
43
44     def Comenzar(self):
45         # Distancia de separación 28 cm
46         # Distancia de recolección 25 cm
47         #Paso en centímetros pausada 1 = 0,34
48         #Paso en centímetros pausada 2 = 0,30
49         #Paso en centímetros pausada 3 = 0,27
50         #Paso en centímetros pausada 4 = 0,245
51         #Paso en centímetros pausada 5 = 0,213
52         for n in range (0, 117):
53             os.system('rm datos/dat.dat')
54             #time.sleep(2)

```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/c_p_Ate.py
Página 2 de 2                                         mar 05 abr 2016 13:19:33 COT

55     arduino= serial.Serial(self.puerta, 9600)
56     print "aca va la lectura"
57     arduino.write("aa")
58     #time.sleep(1)
59     arduino.write('4')
60     arduino.close()
61     arduino=serial.Serial(self.puerta, 9600)
62     time.sleep(2)
63     arduino.write('zz')
64     for i in range(0, 140):
65         arduino=serial.Serial(self.puerta, 9600)
66         archi = open('datos/dat.dat', 'a+')
67         time.sleep(0.00005)
68         x = arduino.readline()
69         z = 0.21367*2*(140 - n)
70         xo = str(z)
71         yo = str(x)
72         print "paso numero", n
73         print "(cm) \t (microw)"
74         print('{0} {1}'.format(xo, yo))
75         archi.write (xo)
76         archi.write (" ")
77         archi.write (yo)
78         archi.close()
79     else:
80         os.system("octave bin/prom1.m")
81         archi = open('datos/prom.dat', 'a+')
82         print("aca va la pausa")
83         Lectura = archi.read()
84         archi.close()
85         archil = open('datos/dats1.dat', 'a+')
86         archil.write(Lectura)
87         archil.close()
88         arduino.write('aa')
89         arduino.close()
90
91     def Grafw(self):
92         os.system("python g_p_Ate.py")
93
94     def Ordenar(self):
95         os.system("python bin/o_Carpetas.py")
96
97     def __init__(self):
98         self.Verifica()
99         self.Comenzar()
100    self.Grafw()
101    self.Ordenar()
102    self.__del__()
103
104   def __del__(self):
105       print ("PROGRAMA TERMINADO")
106
107
108   Iniciar = App()
109

```

Apéndice E20. *Estadis2.py*

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/estadis2.py
Página 1 de 3                                         mar 05 abr 2016 13:23:06 COT

1  #!/usr/bin/python
2  #!**-* coding:utf-8 **-
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Dr Julian Andres Salamanca Bernal
13 # Diego Alberto Parra Garzón
14 # Colombia, Bogota D.C.
15 from matplotlib.widgets import RectangleSelector
16 from numpy import *
17 import numpy as np
18 import matplotlib.pyplot as pl
19 import os
20 import subprocess
21 import math
22 import time
23 import shutil
24 import Gnuplot
25 from matplotlib.widgets import Cursor
26 from pylab import *
27 class Estadistica:
28     def Cargar(self):
29         self.f= np.loadtxt('datos/dats1.dat')
30         self.x , self.y = np.loadtxt('datos/dats1.dat', unpack=True, usecols=[0,1])
31         # self.x , self.y = np.loadtxt('dats1.dat', unpack=True, usecols=[0,1])
32         self.y = self.y*10
33         self.n = np.size(self.x)
34         self.k = np.ceil(1 + np.log2(self.n)) # comando ceil redondea el numero al mayor entero
35         self.c = np.sort(self.y) # comando sort guarda los datos de y
36         # self.lon = self.c[110]/8
37         print self.n
38         # print "El tamaño de la muestra es: ", self.n , "\nEl tamaño del intervalo es ", self.k, self.lon
39
40     def Grafica(self):
41         def onselect(eclick, erelease):
42             print(' startposition : (%f, %f)' % (eclick.xdata, eclick.ydata))
43             print(' endposition   : (%f, %f)' % (erelease.xdata, erelease.ydata))
44             print(' used button   : ', eclick.button)
45             self.xinicial = round(eclick.xdata,2)
46             self.yinicial= round(eclick.ydata,11)
47             self.xfinal = round(erelease.xdata,2)
48             self.yfinal = round(erelease.ydata,11)
49             print self.xinicial, self.yinicial , self.xfinal, self.yfinal
50
51         def toggle_selector(event):
52             print(' Key pressed.')
53             if event.key in ['Q', 'q'] and toggle_selector.RS.active:

```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/estadis2.py
Página 2 de 3
                                         mar 05 abr 2016 13:23:06 COT

54     print(' RectangleSelector deactivated.')
55     toggle_selector.RS.set_active(False)
56     if event.key in ['A', 'a'] and not toggle_selector.RS.active:
57         print(' RectangleSelector activated.')
58         toggle_selector.RS.set_active(True)
59
60     pl.xlabel('Distancia [cm]')
61     pl.ylabel('Intensidad [micro W]')
62     pl.title('LONGITUD DE ONDA DIODO INFRARROJO \n')
63     fig = figure
64     ax = subplot(111)
65     #     pl.legend(loc='upper left')
66     ax.plot(self.x, self.y, 'o--')
67     cursor = Cursor(ax, useblit=True, color='red', linewidth=2)
68     toggle_selector.RS = RectangleSelector(ax, onselect, drawtype='line')
69     connect('key_press_event', toggle_selector)
70     pl.subplots_adjust(right=0.97)
71     pl.subplots_adjust(left=0.18)
72     pl.subplots_adjust(bottom=0.13)
73     pl.subplots_adjust(top=0.87)
74     pl.subplots_adjust(wspace=0.32)
75     pl.subplots_adjust(hspace=0.71)
76     pl.show()
77
78     def Estadistica(self):
79         self.distancia = self.xfinal - self.xinicial
80         d = 4.81*10**(-6)
81         x = self.distancia*10**(-2)
82         y = 45*10**(-2)
83         tetharad = math.atan(x/y)
84         tethagra = math.degrees(tetharad)
85         sintetha= math.sin(tetharad)
86         self.lamda = round((2*d*sintetha)*10**(9),0)
87         self.error = round(pow(pow(850-self.lamda,2),0.5), 2)
88         self.error1 = round(pow(pow(100-self.lamda*100/850, 2), 0.5), 3)
89         print tethagra, "\n ", sintetha
90         print "la longitud de onda aproximada para el diodo infrarrojo es", self.
91         lamda , "nanometros, con un error de ", self.error1 , " %"
92
93     def Grafical(self):
94         self.x1 = self.x/100
95         self.y1 = self.y/1000000
96         pl.subplot(221)
97         pl.plot(self.x1,self.y1, 'o--')
98         pl.title('Datos Capturados \n')
99         pl.xlabel('Distancia [m]')
100        pl.ylabel('Intensidad [W]')
101        pl.ylim(0, 0.001)
102        pl.plot([self.xinicial/100, self.xinicial/100], [0, 0.0005], '--')
103        pl.plot([self.xfinal/100, self.xfinal/100], [0, 0.0005], '--')
104    # pl.ylim(0, 0.02)
105
106    def Grafica2(self):
107        pl.subplot(222)
108        pl.plot(self.x1, self.y1, 'o--')

```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/estadis2.py
Página 3 de 3
                                         mar 05 abr 2016 13:23:06 COT

108     pl.title('Patrones de Interferencia \n')
109     pl.xlabel('Distancia [m]')
110     pl.ylabel('Intensidad [W]')
111     pl.text(0.001, 0.0008, r' x1 = ' + str(self.xinicial/100))
112     pl.text(0.001, 0.0006, r' x2 = ' + str(self.xfinal/100))
113     pl.text(0.001, 0.0004, r' x2-x1 = ' + str(self.distancia/100))
114     pl.ylim(0, 0.001)
115         pl.plot([self.xinicial/100, self.xinicial/100], [0, 0.0005], '--')
116         pl.plot([self.xfinal/100, self.xfinal/100], [0, 0.0005], '--')
117         pl.plot(self.x1, self.y1, 'o--')
118 #    pl.ylim(0, 0.022)
119
120     def Grafica3(self):
121         pl.subplot(212)
122             pl.plot(self.x1, self.y1, 'o--')
123         pl.ylim(0, 0.001)
124             pl.plot([self.xinicial/100, self.xinicial/100], [0, 0.0005], '--')
125             pl.plot([self.xfinal/100, self.xfinal/100], [0, 0.0005], '--')
126 #            pl.plot([self.xinicial/100, self.xfinal/100], [0, 0.0000000001], '-')
127         pl.text(0.001, 0.0008, r' m:1 Lamda IRE = ' + str(self.lamda) + ' [nm] +/- ' + str(self.error) + ' [nm]')
128         pl.text(0.001, 0.0006, r' Distancia entre asintotas ' + str(self.distancia/100) + ' [m]')
129         pl.text(0.001, 0.0004, r' Error porcentual ' + str(self.error1) + ' %')
130 #    pl.ylim(0, 0.022)
131         pl.xlabel('Distancia [m]')
132         pl.ylabel('Intensidad [W]')
133         pl.title('LONGITUD DE ONDA DIODO INFRARROJO \n')
134
135     def Plotear(self):
136         pl.subplots_adjust(right=0.97)
137             pl.subplots_adjust(left=0.18)
138             pl.subplots_adjust(bottom=0.13)
139             pl.subplots_adjust(top=0.87)
140             pl.subplots_adjust(wspace=0.62)
141             pl.subplots_adjust(hspace=0.71)
142             pl.savefig('datos/Graficas.png')
143             pl.show()
144
145     def ordena(self):
146         os.system("python bin/o_Carpetas1.py")
147
148     def __init__(self):
149         self.Cargar()
150         self.Grafica()
151         self.Estadistica()
152         self.Grafica1()
153         self.Grafica2()
154         self.Grafica3()
155         self.Plotear()
156         self.ordera()
157
158 esto = Estadistica()
159

```

Apéndice E21. o_Carpetas.py

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/o_Carpetas.py
Página 1 de 1
                                         mar 05 abr 2016 13:23:30 COT
```

```

1  #!/usr/bin/python
2  # -*- coding: utf-8 -*-
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Diego Alberto Parra Garzón
13 # Dr Julian Andres Salamanca Bernal
14 # Colombia, Bogota D.C.
15
16 import os
17 import time
18 import shutil
19 import Gnuplot
20
21 class o_Carpetas:
22
23     def Carpeta(self):
24         self.Carpeta = str(time.asctime())
25         os.mkdir(self.Carpeta)
26
27     def c_Carpeta(self):
28         archi = open("datos/name.dat","w")
29         archi.write(self.Carpeta)
30         archi.close()
31         time.sleep(3)
32         shutil.move("datos/Yteo.dat", self.Carpeta)
33         shutil.move("datos/Yest.dat", self.Carpeta)
34         shutil.move("datos/F.dat", self.Carpeta)
35         shutil.move("datos/a.dat", self.Carpeta)
36         shutil.move("datos/b.dat", self.Carpeta)
37         shutil.move("datos/ECM.dat", self.Carpeta)
38         shutil.move("datos/dats1.dat", self.Carpeta)
39         shutil.move("datos/Atenuacion.png", self.Carpeta)
40         os.system("sh bin/m_Carpeta.sh")
41         os.system("rm datos/dat.dat prom.dat")
42
43     def __init__(self):
44         self.Carpeta()
45         self.c_Carpeta()
46         self.__del__()
47
48     def __del__(self):
49         print ("PROGRAMA TERMINADO")
50
51
52 if __name__ == "__main__":
53     o_Carpetas()
```

Apéndice E22. o_Carpetas1.py

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/o_Carpetas1.py
Página 1 de 1
```

```
mar 05 abr 2016 13:23:59 COT
```

```
1 #!/usr/bin/python
2 # -*- coding: utf-8 -*-
3 # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4 # los términos de la licencia pública general de GNU, según es publicada
5 # por la free software fundation bien la versión 3 de la misma licencia
6 # o de cualquier versión posterior. (según su elección).
7 # Si usted hace alguna modificación en esta aplicación, deberá siempre
8 # mencionar el autor original de la misma.
9 # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Diego Alberto Parra Garzón
13 # Dr Julian Andres Salamanca Bernal
14 # Colombia, Bogota D.C.
15
16 import os
17 import time
18 import shutil
19 import Gnuplot
20
21 class o_Carpetas:
22
23     def Carpeta(self):
24         self.Carpeta = str(time.asctime())
25         os.mkdir(self.Carpeta)
26
27     def c_Carpeta(self):
28         archi = open("datos/name.dat","w")
29         archi.write(self.Carpeta)
30         archi.close()
31         time.sleep(3)
32         shutil.move("datos/dats1.dat", self.Carpeta)
33         shutil.move("datos/Graficas.png", self.Carpeta)
34
35     def Llama(self):
36         os.system("sh bin/m_Carpeta1.sh")
37
38     def __init__(self):
39         self.Carpeta()
40         self.c_Carpeta()
41         self.Llama()
42         self.__del__()
43
44     def __del__(self):
45         print ("PROGRAMA TERMINADO")
46
47
48 if __name__ == "__main__":
49     o_Carpetas()
```

Apéndice E23. o_Carpetas2.py

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/o_Carpetas2.py
Página 1 de 2
mar 05 abr 2016 13:43:43 COT

1 #!/usr/bin/python
2 # -*- coding: utf-8 -*-
3 # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4 # los términos de la licencia pública general de GNU, según es publicada
5 # por la free software fundation bien la versión 3 de la misma licencia
6 # o de cualquier versión posterior. (según su elección).
7 # Si usted hace alguna modificación en esta aplicación, deberá siempre
8 # mencionar el autor original de la misma.
9 # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Diego Alberto Parra Garzón
13 # Dr Julian Andres Salamanca Bernal
14 # Colombia, Bogota D.C.
15
16 import os
17 import time
18 import shutil
19 import Gnuplot
20
21 class o_Carpetas:
22
23     def Carpeta(self):
24         self.Carpeta = str(time.asctime())
25         os.mkdir(self.Carpeta)
26
27     def c_Carpeta(self):
28         archi = open("datos/name.dat","w")
29         archi.write(self.Carpeta)
30         archi.close()
31         time.sleep(3)
32         shutil.move("datos/Yteo.dat", self.Carpeta)
33         shutil.move("datos/Yest.dat", self.Carpeta)
34         shutil.move("datos/F.dat", self.Carpeta)
35         shutil.move("datos/Amplitud.dat", self.Carpeta)
36         shutil.move("datos/BY.dat", self.Carpeta)
37         shutil.move("datos/ECM1.dat", self.Carpeta)
38         shutil.move("datos/q1.dat", self.Carpeta)
39         shutil.move("datos/Transmitancia.dat", self.Carpeta)
40         shutil.move("datos/V_aproximado.dat", self.Carpeta)
41         shutil.move("datos/Vteo.dat", self.Carpeta)
42         shutil.move("datos/a.dat", self.Carpeta)
43         shutil.move("datos/b.dat", self.Carpeta)
44         shutil.move("datos/ECM.dat", self.Carpeta)
45         shutil.move("datos/dats1.dat", self.Carpeta)
46         shutil.move("datos/Absorcion.png", self.Carpeta)
47         os.system("sh bin/m_Carpeta2.sh")
48         os.system("rm datos/dat.dat prom.dat")
49
50     def __init__(self):
51         self.Carpeta()
52         self.c_Carpeta()
53         self.__del__()
54
55     def __del__(self):
```

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/o_Carpetas2.py  
Página 2 de 2
```

```
56     print ("PROGRAMA TERMINADO")  
57  
58  
59 if __name__ == "__main__":  
60     o_Carpetas()  
61
```

```
mar 05 abr 2016 13:43:43 COT
```

Apéndice E24. m_Carpeta.sh

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/m_Carpeta.sh
Página 1 de 1
```

```
mar 05 abr 2016 13:42:16 COT
```

```
1 #!/bin/bash
2 # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
3 # los términos de la licencia pública general de GNU, según es publicada
4 # por la free software fundation bien la versión 3 de la misma licencia
5 # o de cualquier versión posterior. (según su elección).
6 # Si usted hace alguna modificación en esta aplicación, deberá siempre
7 # mencionar el autor original de la misma.
8 # Autor:
9 # Universidad Distrital Francisco Jose
10 # Grupo de física e informática
11 # Diego Alberto Parra Garzón
12 # Dr Julian Andres Salamanca Bernal
13 # Colombia, Bogota D.C.
14
15 ARCHIVO=`cat datos/name.dat`
16 mv -f "$ARCHIVO" Carpetas/Atenuacion
17
18
```

Apéndice E25. m_Carpetas1.sh

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/m_Carpeta1.sh  
Página 1 de 1
```

```
mar 05 abr 2016 13:42:30 COT
```

```
1 #!/bin/bash  
2 # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo  
3 # los términos de la licencia pública general de GNU, según es publicada  
4 # por la free software fundation bien la versión 3 de la misma licencia  
5 # o de cualquier versión posterior. (según su elección ).  
6 # Si usted hace alguna modificación en esta aplicación, deberá siempre  
7 # mencionar el autor original de la misma.  
8 # Autor:  
9 # Universidad Distrital Francisco Jose  
10 # Grupo de física e informática  
11 # Diego Alberto Parra Garzón  
12 # Dr Julian Andres Salamanca Bernal  
13 # Colombia, Bogota D.C.  
14 ARCHIVO=`cat datos/name.dat`  
15 mv -f "$ARCHIVO" Carpetas/Difraccion  
16  
17
```

Apéndice E26. m_Carpetas2.sh

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/m_Carpeta2.sh
Página 1 de 1
```

```
mar 05 abr 2016 13:42:42 COT
```

```
1 #!/bin/bash
2 # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
3 # los términos de la licencia pública general de GNU, según es publicada
4 # por la free software fundation bien la versión 3 de la misma licencia
5 # o de cualquier versión posterior. (según su elección).
6 # Si usted hace alguna modificación en esta aplicación, deberá siempre
7 # mencionar el autor original de la misma.
8 # Autor:
9 # Universidad Distrital Francisco Jose
10 # Grupo de fisica e informatica
11 # Diego Alberto Parra Garzón
12 # Dr Julian Andres Salamanca Bernal
13 # Colombia, Bogota D.C.
14 ARCHIVO=`cat datos/name.dat`
15 mv -f "$ARCHIVO" Carpetas/Absorción
16
17
```

Apéndice E27. *est_fuente.py*

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/Carpetas/FUENTE/est_fuente.py
Página 1 de 2
  mar 05 abr 2016 13:45:44 COT

1 #!/bin/python
2 # -*- coding:utf-8 -*-
3 # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4 # los términos de la licencia pública general de GNU, según es publicada
5 # por la free software fundation bien la versión 3 de la misma licencia
6 # o de cualquier versión posterior. (según su elección).
7 # Si usted hace alguna modificación en esta aplicación, deberá siempre
8 # mencionar el autor original de la misma.
9 # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Dr Julian Andres Salamanca Bernal
13 # Diego Alberto Parra Garzón
14 # Colombia, Bogota D.C.

15
16
17 from matplotlib.widgets import RectangleSelector
18 from numpy import *
19 import numpy as np
20 import matplotlib.pyplot as pl
21 import os
22 import subprocess
23 import math
24 import time
25 import shutil
26 import Gnuplot
27 from matplotlib.widgets import Cursor
28 from pylab import *
29 class estadis:
30     def Llamar(self):
31         os.system("octave est_fuente.m")
32
33     def Estadis(self):
34         self.x , self.y = np.loadtxt('de.dat', unpack=True, usecols=[0,1])
35         self.xp , self.yp = np.loadtxt('dse.dat', unpack=True, usecols=[0,1])
36         print self.x, self.y
37
38     def Graf0(self):
39         pl.subplot(221)
40         pl.xlabel('ANGULO [grados]')
41         pl.ylabel('Intensidad [micro W]')
42         pl.title('DISPERSION SIN SISTEMA OPTICO\n')
43         pl.grid()
44         pl.plot(self.xp, self.yp, 'o--')
45
46     def Graf1(self):
47         pl.subplot(222)
48         pl.xlabel('ANGULO [grados]')
49         pl.ylabel('Intensidad [micro W]')
50         pl.title('DISPERSION CON SISTEMA OPTICO\n')
51         pl.grid()
52         pl.plot(self.x, self.y, 'R')
53
54     def Graf2(self):
55         pl.subplot(212)
```

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/Carpetas/FUENTE/est_fuente.py
Página 2 de 2
                                         mar 05 abr 2016 13:45:44 COT

56     pl.xlabel('ANGULO [grados]')
57     pl.ylabel('Intensidad [micro W]')
58     pl.title('DISPERSION SIN Y CON SISTEMA OPTICO\n')
59     pl.grid()
60     pl.plot(self.x, self.y, 'R')
61     pl.plot(self.xp, self.yp, 'o--')
62
63     def Plotear(self):
64         pl.subplots_adjust(left=0.11)
65         pl.subplots_adjust(bottom=0.13)
66         pl.subplots_adjust(right=0.90)
67         pl.subplots_adjust(top=0.87)
68         pl.subplots_adjust(wspace=0.56)
69         pl.subplots_adjust(hspace=0.71)
70         pl.savefig('Graficas.png')
71         pl.show()
72
73
74     def __init__(self):
75         self.Llamar()
76         self.Estadis()
77         self.Graf0()
78         self.Graf1()
79         self.Graf2()
80         self.Plotear()
81     esto = estadis()
82
```

Apéndice E28. G_firmware.py

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/G_firmware.py
Página 1 de 2
                                         mar 05 abr 2016 14:08:15 COT

1  #!/usr/bin/python
2  #!**-* coding:utf-8 **-
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Diego Alberto Parra Garzón
13 # Dr Julian Andres Salamanca Bernal
14 # Colombia, Bogota D.C.
15 import numpy as np
16 import pylab as pl
17 import serial
18 import os
19 import subprocess
20 import math
21 import time
22 import Gnuplot
23 from Tkinter import *
24 import tkMessageBox
25 import Tkinter
26 import shutil
27
28 class G_firmware:
29     def Difraccion(self):
30         def Salir():
31             tkMessageBox.showinfo("free_infrarossi", message= " ! Cerrando el programa ")
32             exit()
33
34         def Continuar():
35             tkMessageBox.showinfo("free_infrarossi", message= "Cargando el firmware en la tarjeta microcontroladora. \n\n si no carga el firmware: \n\n* Revise su conexión con la tarjeta programadora arduino uno. \n\n* Revise el microcontrolador que este funcionando bien ")
36             os.system("python bin/firmware/firmware_free_infrarossi.py &")
37
38         bicho = Tk()
39         bicho.geometry("280x180+507+60")
40         bicho.config(bg="white")
41         bicho.title("free_infrarossi")
42         bicho.resizable(width=0, height=0)
43         yn= 400
44         xn = 350
45         # imgBoton2=PhotoImage(file="Imagenes/cap8.gif")
46         # btnLogo= Label(bicho, image=imgBoton2, height=150, width=180).place(x=400+xn, y=215+yn)
47         # lblFisinfor = Label(bicho, text=" GRUPO DE FISICA E INFORMATICA ", fg = ("black"), bg = ("white"), font = ("Century Schoolbook L",10)).place(x=360+xn, y=371+yn)
48         # lblInfo = Label(bicho, text="Dr. Julian Andres Salamanca\n Diego Alberto "

```

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/G_firmware.py
Página 2 de 2
                                         mar 05 abr 2016 14:08:15 COT
    Parra Garzón", fg = ("black"), bg = ("white"), font = ("Century Schoolbook
L",10)).place(x=400+xn, y=390+yn)
49     lblFisinfor = Label(bicho, text="DEBE TENER LA TARJETA
\nMICROCONTROLADORA ARDUINO\n CONECTADA PARA CONTINUAR", fg = ("red"), bg
= ("white"), font = ("Century Schoolbook L",11)).place(x=350-xn, y=440-yn)
50     btnSalir=Button(bicho, text = "Salir", command=Salir, height=1, width =5).
place(x=20, y=540-yn)
51     btnContinuar=Button(bicho, text = "Continuar", command=Continuar, height=1,
width =8).place(x=168, y=540-yn)
52
53     bicho.mainloop()
54
55
56
57     def __init__(self):
58         self.Difraccion()
59         self.__del__()
60
61     def __del__(self):
62         print ("PROGRAMA TERMINADO")
63
64 if __name__ == "__main__":
65     G_firmware()
66
```

Apéndice E29. *firmware_free_infrarossi.py*

/home/diego/Documentos/Free-infrarossi/free_in...arossi/bin/firmware/firmware_free_infrarossi.py
Página 1 de 2 mar 05 abr 2016 14:08:37 COT

```

1  #!/usr/bin/python
2  #!*- coding:utf-8 *-
3  # Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
4  # los términos de la licencia pública general de GNU, según es publicada
5  # por la free software fundation bien la versión 3 de la misma licencia
6  # o de cualquier versión posterior. (según su elección).
7  # Si usted hace alguna modificación en esta aplicación, deberá siempre
8  # mencionar el autor original de la misma.
9  # Autor:
10 # Universidad Distrital Francisco Jose
11 # Grupo de fisica e informatica
12 # Diego Alberto Parra Garzón
13 # Dr Julian Andres Salamanca Bernal
14 # Colombia, Bogota D.C.
15 import serial
16 import os
17 import time
18
19 class Firmware:
20     def b_Puerto(self):
21         os.system("ls /dev/ttyACM* > puerto.txt")
22         archi=open("puerto.txt","r")
23         puerto = archi.readline()
24         self.puerto = str(puerto)
25         archi.close()
26         print self.puerto
27
28     def c_arch_Carga(self):
29         archi=open("bin/firmware/a_carga.sh", "a+")
30         archi.write("#!/bin/bash")
31         archi.write("\n")
32         archi.write("CARGA=`avrdude -F -V -c arduino -p ATMEGA328P -b 115200 -U
33             flash:w:bin/firmware/infrarossi.hex -P `")
34         archi.write(self.puerto)
35         archi.write(``)
36         archi.write("\n")
37         archi.write('$CARGA')
38         # archi.write("\n")
39         # archi.write("echo firmware instalado correctamente")
40         archi.write("\n")
41         archi.write("sleep 10")
42         archi.close()
43         time.sleep(1)
44         os.system("xterm -T infrarossi -geom 82x27+0+55 +cm -bg blue -e 'sh
45             bin/firmware/a_carga.sh' &")
46         time.sleep(8)
47
48     def limpiar(self):
49         os.system("rm bin/firmware/a_carga.sh bin/firmware/puerto.txt")
50
51     def __init__(self):
52         self.b_Puerto()
53         self.c_arch_Carga()
54         self.limpiar()

```

```
/home/diego/Documentos/Free-infrarossi/free_in...arossi/bin/firmware/firmware_free_infrarossi.py
Página 2 de 2
mar 05 abr 2016 14:08:37 COT
```

```
54
55 if __name__ == "__main__":
56     Firmware()
57
```

Apéndice E30. *infrarossi.cpp*

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.cpp
Página 1 de 9
                                         mar 05 abr 2016 14:09:00 COT

1 // Este script es software libre. Puede redistribuirlo y/o modificarlo bajo
2 // los términos de la licencia pública general de GNU, según es publicada
3 // por la free software fundation bien la versión 3 de la misma licencia
4 //o de cualquier versión posterior. (según su elección).
5 // Si usted hace alguna modificación en esta aplicación, deberá siempre
6 // mencionar el autor original de la misma.
7 // Autor:
8 // Universidad Distrital Francisco Jose
9 // Grupo de fisica e informatica
10 // Diego Alberto Parra Garzón
11 // Dr Julian Andres Salamanca Bernal
12 // Colombia, Bogota D.C.
13 #include "Arduino.h"
14 void setup();
15 void Velo0();
16 void Velo1();
17 void Velo2();
18 void Velo3();
19 void Velo4();
20 void Velo5();
21 void Velo6();
22 void Vepa1();
23 void Vepa2();
24 void Vepa3();
25 void Vepa4();
26 void Vepa5();
27 void Inten1();
28 void Inten2();
29 void inicio();
30 void Menu();
31 void loop();
32 #line 1
33 int motor = 3;
34 const int sensor1 = A0;
35 const int sensor2 = A5;
36 const int ledRojo = 12;
37 const int ledAzul = 9;
38 const int ledVerde = 10;
39 const int ledEmisor = 6;
40 //const int resetea = 4;
41 //variables
42 long miliVolts1;
43 long intensidad1;
44 int brillo1;
45
46
47 void setup()
48 {
49     Serial.begin(9600);
50     pinMode(ledAzul, OUTPUT);
51     pinMode(ledVerde, OUTPUT);
52     pinMode(ledRojo, OUTPUT);
53     pinMode(ledEmisor, OUTPUT);
54 //    pinMode(resetea, OUTPUT);
55 }

```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.cpp
Página 2 de 9
                                         mar 05 abr 2016 14:09:00 COT

56
57     void Velo0()
58     {
59         //int velocidad = map(velocidad,'0', '5', 0, 255);
60         //velocidad = 0;
61         //analogWrite(motor, velocidad);
62         analogWrite(motor, 0);
63         delay(20);
64     }
65
66     void Velo1()
67     {
68         //int velocidad = map(velocidad,'0', '255', 0, 255);
69         //velocidad = '150';
70         //analogWrite(motor, velocidad);
71         analogWrite(motor, 135);
72         //delay(20);
73     }
74
75     void Velo2()
76     {
77         //int velocidad = map(velocidad,'0', '255', 0, 255);
78         //velocidad = '160';
79         //analogWrite(motor, velocidad);
80         analogWrite(motor, 140);
81         //delay(20);
82     }
83
84     void Velo3()
85     {
86         //int velocidad = map(velocidad,'0', '255', 0, 255);
87         //velocidad = '170';
88         //analogWrite(motor, velocidad);
89         analogWrite(motor, 145);
90         //delay(20);
91     }
92
93     void Velo4()
94     {
95         //int velocidad = map(velocidad,'0', '255', 0, 255);
96         //velocidad = '180';
97         //analogWrite(motor, velocidad);
98         analogWrite(motor, 150);
99         //delay(20);
100    }
101
102    void Velo5()
103    {
104        //int velocidad = map(velocidad,'0', '255', 0, 255);
105        //velocidad = '190';
106        //analogWrite(motor, velocidad);
107        analogWrite(motor, 160);
108        //delay(20);
109    }
110

```

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.cpp
Página 3 de 9
                                         mar 05 abr 2016 14:09:00 COT

111 void Velo6()
112 {
113     //int velocidad = map(velocidad,'0', '255', 0, 255);
114     //velocidad = '200';
115     //analogWrite(motor, velocidad);
116     analogWrite(motor, 190);
117     //    delay(20);
118 }
119
120 void Vepa1()
121 {
122     analogWrite(motor, 150);
123     delay(55);
124     Velo0();
125     delay(1000);
126     // digitalWrite(resetea, HIGH);
127 }
128
129
130 void Vepa2()
131 {
132     analogWrite(motor, 150);
133     delay(50);
134     Velo0();
135     delay(1000);
136 }
137
138 void Vepa3()
139 {
140     analogWrite(motor, 150);
141     delay(45);
142     Velo0();
143     delay(1000);
144 }
145
146 void Vepa4()
147 {
148     analogWrite(motor, 150);
149     delay(40);
150     Velo0();
151     delay(1000);
152 }
153
154 void Vepa5()
155 {
156     analogWrite(motor, 150);
157     delay(37);
158     Velo0();
159     delay(1000);
160 }
161
162 void Inten1()
163 {
164     miliVolts1 = ((analogRead(sensor1) *5000L) /1023)*4; //obteniendo el valor sensor
165     intensidad1 =miliVolts1;
```

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.cpp
Página 4 de 9                                         mar 05 abr 2016 14:09:00 COT

166     brillo1 = map(intensidad1, 0, 5000, 0, 255); //funcion map (mapeo) convierte
167     la variable y le da un rango y un dominio
168     brillo1 = constrain(brillo1, 0, 255); //funcion constrain o contenido en el
169     intervalo de analogWrite (0, 255)
170     digitalWrite(ledAzul, 255 - brillo1); //Salida del led si esta el obstaculo
171     esta lejos
172     digitalWrite(ledVerde, brillo1); //Salida del led si el obstaculo esta cerca
173     Serial.print(" "); //salida al Serialporth
174     Serial.print(intensidad1);
175     Serial.println(" ");
176     delay (3);
177 }
178
179 void Inten2()
180 {
181     miliVolts1 = ((analogRead(sensor2) *5000L) /(1023))*4; //obteniendo el valor
182     sensor
183     intensidad1 =miliVolts1;
184     brillo1 = map(intensidad1, 0, 5000, 0, 255); //funcion map (mapeo) convierte
185     la variable y le da un rango y un dominio
186     brillo1 = constrain(brillo1, 0, 255); //funcion constrain o contenido en el
187     intervalo de analogWrite (0, 255)
188     digitalWrite(ledEmisor, 255);
189     digitalWrite(ledAzul, 255-brillo1); //Salida del led si esta el obstaculo esta
190     lejos
191     digitalWrite(ledVerde, brillo1); //Salida del led si el obstaculo esta cerca
192     Serial.print(" "); //salida al Serialport
193     Serial.print(intensidad1);
194     Serial.println(" ");
195     delay (3);
196 }
197
198 //void resetear()
199 //{
200     //digitalWrite(resetea, HIGH);
201     //delay(150);
202     //digitalWrite(resetea, LOW);
203     //delay(150);
204 //}
205
206 void inicio()
207 {
208     digitalWrite(ledRojo, HIGH); // turn the LED on (HIGH is the voltage level)
209     digitalWrite(ledAzul, 255);
210     digitalWrite(ledVerde, 255);
211     delay(100); // wait for a second
212     digitalWrite(ledRojo, LOW); // turn the LED off by making the voltage LOW
213     digitalWrite(ledAzul, 255);
214     digitalWrite(ledVerde, 255);
215     delay(100);
216 }
217
218 void Menu()
219 {
220 }
```

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.cpp
Página 5 de 9
                                         mar 05 abr 2016 14:09:00 COT

214     char opcion = Serial.read();
215     switch (opcion )
216     {
217         case 'a':
218
219             while(opcion=='a')
220             {
221                 inicio();
222                 Velo0();
223                 Menu();
224             }
225             break;
226
227         case 'b':
228
229             while(opcion=='b')
230             {
231                 inicio();
232                 Velo1();
233                 Menu();
234             }
235             break;
236
237         case 'c':
238             while(opcion=='b')
239             {
240                 inicio();
241                 Velo2();
242                 Menu();
243             }
244             break;
245
246         case 'd':
247             while(opcion=='d')
248             {
249                 inicio();
250                 Velo3();
251                 Menu();
252             }
253             break;
254
255         case 'e':
256             while(opcion=='e')
257             {
258                 inicio();
259                 Velo4();
260                 Menu();
261             }
262             break;
263
264         case 'f':
265             while(opcion=='f')
266             {
267
```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.cpp
Página 6 de 9
                                         mar 05 abr 2016 14:09:00 COT

269         inicio();
270         Velo5();
271         Menu();
272     }
273     break;
274
275     case 'g':
276         while(opcion=='g')
277     {
278         inicio();
279         Velo6();
280         Menu();
281     }
282     break;
283
284     case 'h':
285         while(opcion=='h')
286     {
287             Velo0();
288             Inten1();
289             Menu();
290         }
291         break;
292
293     case 'i':
294         while(opcion=='i')
295     {
296             Velo1();
297             Inten1();
298             Menu();
299         }
299         break;
300
301     case 'j':
302         while(opcion=='j')
303     {
304             Velo2();
305             Inten1();
306             Menu();
307         }
308         break;
309
310     case 'k':
311
312         while(opcion=='k')
313     {
314             Velo3();
315             Inten1();
316             Menu();
317         }
318         break;
319
320     case 'l':
321         while(opcion=='l')
322     {
323

```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.cpp
Página 7 de 9
mar 05 abr 2016 14:09:00 COT

324         Velo4();
325         Inten1();
326         Menu();
327     }
328     break;
329
330     case 'm':
331         while(opcion=='m')
332     {
333         Velo5();
334         Inten1();
335         Menu();
336     }
337     break;
338
339     case 'n':
340         while(opcion=='n')
341     {
342         Velo6();
343         Inten1();
344         Menu();
345     }
346     break;
347
348     case 'z':
349         while(opcion=='z')
350     {
351         Velo0();
352         Inten2();
353         Menu();
354     }
355     break;
356
357     case 'o':
358         while(opcion=='o')
359     {
360             Velo1();
361             Inten2();
362             Menu();
363         }
364         break;
365
366     case 'p':
367         while(opcion=='p')
368     {
369             Velo2();
370             Inten2();
371             Menu();
372         }
373         break;
374
375     case 'q':
376         while(opcion=='q')
377     {
378             Velo3();

```

```

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.cpp
Página 8 de 9
                                         mar 05 abr 2016 14:09:00 COT

379         Inten2();
380         Menu();
381     }
382     break;
383
384     case 'r':
385         while(opcion=='r')
386     {
387         Velo4();
388         Inten2();
389         Menu();
390     }
391     break;
392
393     case 's':
394         while(opcion=='s')
395     {
396         Velo5();
397         Inten2();
398         Menu();
399     }
400     break;
401
402     case 't':
403         while(opcion=='t')
404     {
405         Velo6();
406         Inten2();
407         Menu();
408     }
409     break;
410 //case 'u':
411 //    {
412 //        resetear();
413 //        Menu();
414 //    }
415 //    break;
416
417     case '1':
418     {
419         Vepa1();
420         Menu();
421     }
422     break;
423
424     case '2':
425     {
426         Vepa2();
427         Menu();
428     }
429     break;
430     case '3':
431     {
432         Vepa3();
433         Menu();

```

```
/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.cpp
Página 9 de 9
mar 05 abr 2016 14:09:00 COT
```

```
434 }
435     break;
436     case '4':
437     {
438         Vepa4();
439         Menu();
440     }
441     break;
442     case '5':
443     {
444         Vepa5();
445         Menu();
446     }
447     break;
448 }
449 }
450 }
451
452 void loop()
453 {
454     inicio();
455     Menu();
456 }
457 }
458 }
459 }
460 }
```

Apéndice E31. *infrarossi.hex*

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.hex	
Página 1 de 6	
	mar 05 abr 2016 14:09:16 COT
1	:100000000C94A7000C94CF000C94CF000C94CF005C
2	:100010000C94CF000C94CF000C94CF000C94CF0024
3	:100020000C94CF000C94CF000C94CF000C94CF0014
4	:100030000C94CF000C94CF000C94CF000C94CF0004
5	:100040000C940F040C94CF000C940E070C945107E1
6	:100050000C94CF000C94CF000C94CF000C94CF00E4
7	:100060000C94CF000C94CF006303660369036C0308
8	:100070006F03720372037203720372037203DB
9	:100080007203720372037203720372037203C8
10	:10009000720372037203720372037203B8
11	:1000A000720372037203720372037203A8
12	:1000B000720372037203720372037203720398
13	:1000C000720372037203D702DE027203E50247
14	:1000D000EC02F302FA02010308030F0316031D03E7
15	:1000E00024032B033903400347034E0355035C03EA
16	:1000F00072037203720372033203000000077B
17	:10010000002010000030406000000000000000DF
18	:10011000010204081020408001020408102001029E
19	:1001200004081020040404040404040202026B
20	:10013000020203030303030300000000250028005C
21	:100140002B0000000000240027002A000D081124C5
22	:100150001FBECFEFD8E0DEBFCD8F11E0A0E0B1E021
23	:10016000E8E4F2E102C005900D92AA31B107D9F797
24	:1001700011E0AAE1B1E001C01D92A73DB107E1F78E
25	:1001800011E0CEE4D1E004C02297FE010E940409F0
26	:10019000CC34D107C9F70E9459050C9422090C945C
27	:1001A000000040E855E260E070E08DE291E00E94DE
28	:1001B0008C0761E089E00E94A00361E08AE00E9470
29	:1001C000A00361E08CE00E94A00361E086E00C9453
30	:1001D000A00360E070E0809104010E94FA0464E1F1
31	:1001E00070E080E090E00C947C0467E870E080911F
32	:1001F00004010C94FA046CE870E0809104010C9402
33	:10020000FA0461E970E0809104010C94FA0466E953
34	:1002100070E0809104010C94FA0460EA70E080912F
35	:1002200004010C94FA046EEB70E0809104010C94CC
36	:10023000FA0466E970E0809104010E94FA0467E321
37	:1002400070E080E090E00E947C040E94E90068EE8B
38	:1002500073E080E090E00C947C0466E970E08091AB
39	:1002600004010E94FA0462E370E080E090E00E94E2
40	:100270007C040E94E90068EE73E080E090E00C945A
41	:100280007C0466E970E0809104010E94FA046DE24A
42	:1002900070E080E090E00E947C040E94E90068EE3B
43	:1002A00073E080E090E00C947C0466E970E080915B
44	:1002B00004010E94FA0468E270E080E090E00E948D
45	:1002C0007C040E94E90068EE73E080E090E00C940A
46	:1002D0007C0466E970E0809104010E94FA0465E202
47	:1002E00070E080E090E00E947C040E94E90068EEE8
48	:1002F00073E080E090E00C947C04AF92BF92CF92C8
49	:10030000DF92EF92FF920F931F938EE00E94DD0425
50	:1003100028E833E1DC010E94AA082FEF33E040E037
51	:1003200050E00E94E308CA01B90122E0660F771F7E
52	:10033000881F991F2A95D1F760932001709321019E
53	:10034000809322019093230160931C0170931D01FF
54	:1003500080931E0190931F011F921F921F922FEFF7
55	:100360002F93A12CB12C650118E8E12E13E1F12E99

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.hex	
Página 2 de 6	
56	:100370000E010E020E030E0A9010E9463080F9047
57	:100380000F900F900F906F3F710519F014F06FEF01
58	:1003900070E077FF02C060E070E070931B01609333
59	:1003A0001A018FEF90E0861B970BBC0189E00E9439
60	:1003B000FA0460911A0170911B018AE00E94FA040C
61	:1003C00067E071E08DE291E00E94A40540911C017C
62	:1003D00050911D0160911E0170911F012AE030E0D3
63	:1003E0008DE291E00E942C0667E071E08DE291E0E1
64	:1003F0000E94C10563E070E080E090E01F910F91E2
65	:10040000FF90EF90DF90CF90BF90AF900C947C0462
66	:10041000AF92BF92CF92DF92EF92FF920F931F9312
67	:1004200083E10E94DD0428E833E1DC010E94AA0890
68	:100430002FF33E040E050E00E94E308CA01B90129
69	:1004400042E0660F771F881F991F4A95D1F7609386
70	:1004500020017093210180932201909323016093E6
71	:100460001C0170931D0180931E0190931F011F9228
72	:100470001F921F922FFEF2F93A12CB12C650118E82A
73	:10048000E12E13E1F12E00E10E020E030E0A901C0
74	:100490000E9463080F900F900F900F906F3F7105AF
75	:1004A00019F014F06FEF70E077FF02C060E070E0C9
76	:1004B00070931B0160931A016FEF70E086E00E9459
77	:1004C000FA046FEF70E080911A0190911B01681B94
78	:1004D000790B89E00E94FA0460911A0170911B0166
79	:1004E0008AE00E94FA0467E071E08DE291E00E94E8
80	:1004F000A40540911C0150911D0160911E01709155
81	:100500001F012AE030E08DE291E00E942C0667E0B6
82	:1005100071E08DE291E00E94C10563E070E080E04F
83	:1005200090E01F910F91FF90EF90DF90CF90BF90E0
84	:10053000AF900C947C0461E08CE00E94D9036FEFD3
85	:1005400070E089E00E94FA046FEF70E08AE00E9498
86	:10055000FA0464E670E080E090E00E947C0460E0D1
87	:1005600070E08CE00E94FA046FEF70E089E00E9476
88	:10057000FA046FEF70E08AE00E94FA0464E670E02B
89	:1005800080E090E00C947C048DE291E00E94A30650
90	:10059000992787FD9095A92FB92FFC01F197EA348F
91	:1005A000F10508F09FC0EC5CFF4F0C9402090E941B
92	:1005B0009B020E94E9000E94C402F9CF0E949B02A4
93	:1005C0000E94F5000E94C402F9CF0E949B020E9483
94	:1005D00001010E94C402F9CF0E949B020E94070100
95	:1005E0000E94C402F9CF0E949B020E940D010E944A
96	:1005F000C402F9CF0E949B020E9413010E94C40210
97	:10060000F9CF0E94E9000E947D010E94C402F9CF47
98	:100610000E94F5000E947D010E94C402F9CF0E9451
99	:10062000FB000E947D010E94C402F9CF0E940101DB
100	:100630000E947D010E94C402F9CF0E9407010E941E
101	:100640007D010E94C402F9CF0E940D010E947D012C
102	:100650000E94C402F9CF0E9413010E947D010E94F2
103	:10066000C402F9CF0E94E9000E9408020E94C4025D
104	:10067000F9CF0E94F5000E9408020E94C402F9CF3F
105	:100680000E94FB000E9408020E94C402F9CF0E944F
106	:1006900001010E9408020E94C402F9CF0E940701D2
107	:1006A0000E9408020E94C402F9CF0E940D010E941C
108	:1006B00008020E94C402F9CF0E9413010E9408029E
109	:1006C0000E94C402F9CF0E9419015ECF0E942D0141
110	:1006D0005BCF0E94410158CF0E94550155CF0E9427

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.hex	
Página 3 de 6	
mar 05 abr 2016 14:09:16 COT	
111	:1006E000690152CF08950E949B020C94C40283308A
112	:1006F00081F028F4813099F08230A1F0089586309D
113	:10070000A9F08730B9F08430D1F4809180008F7DDA
114	:1007100003C0809180008F7780938000089584B516
115	:100720008F7702C084B58F7D84BD08958091B0001D
116	:100730008F7703C08091B0008F7D8093B0000895C3
117	:10074000CF93DF9390E0FC01E05FFE4F2491FC012A
118	:10075000EC5DFE4F8491882349F190E0880F991F4A
119	:10076000FC01EE5BFE4FA591B491885C9E4FFC01AD
120	:10077000C591D4919FB7611108C0F8948C912095D0
121	:1007800082238C93888182230AC0623051F4F894CA
122	:100790008C91322F309583238C938881822B888390
123	:1007A00004C0F8948C91822B8C939FBFD91CF91E2
124	:1007B00008950F931F93CF93DF931F92CDB7DEB7AA
125	:1007C000282F30E0F901E450FF4F8491F901E05FF8
126	:1007D000FE4F1491F901EC5DFE4F04910023C9F026
127	:1007E000882321F069830E9477036981E02FF0E07C
128	:1007F000EE0FFF1FE85CFE4FA591B4919FB7F894F0
129	:100800008C91611103C01095812301C0812B8C93C1
130	:100810009FBF0F90DF91CF911F910F9108951F926D
131	:100820000F920FB60F9211242F933F938F939F93A4
132	:10083000AF93BF938091250190912601A09127014C
133	:10084000B09128013091240123E0230F2D3720F4AB
134	:100850000196A11DB11D05C026E8230F0296A11D1A
135	:10086000B11D209324018093250190932601A0932C
136	:100870002701B09328018091290190912A01A0912C
137	:100880002B01B0912C010196A11DB11D809329016E
138	:1008900090932A01A0932B01B0932C01BF91AF91AB
139	:1008A0009F918F913F912F910F900FBE0F901F90AE
140	:1008B00018953FB7F8948091290190912A01A09151
141	:1008C0002B01B0912C0126B5A89B05C02F3F19F034
142	:1008D0000196A11DB11D3FBF6627782F892F9A2F42
143	:1008E000620F711D811D911D42E0660F771F881FE9
144	:1008F000991F4A95D1F70895CF92DF92EF92FF9218
145	:10090000CF93DF936B017C010E945904EB01C1146A
146	:10091000D104E104F10479F00E9459046C1B7D0BB1
147	:10092000683E7340A0F381E0C81AD108E108F108DD
148	:10093000C851DC4FECCFDF91CF91FF90EF90DF906B
149	:10094000CF900895789484B5826084BD84B5816029
150	:1009500084BD85B5826085BD85B5816085BDEEE6C7
151	:10096000F0E0808181608083E1E8F0E010828081A6
152	:1009700082608083808181608083E0E8F0E0808114
153	:1009800081608083E1EBF0E0808184608083E0EB34
154	:10099000F0E0808181608083EAЕ7F0E0808184601C
155	:1009A0008083808182608083808181608083808178
156	:1009B000806880831092C10008958E3008F08E50B8
157	:1009C00087702091060190E4299F90011124822BC9
158	:1009D00080937C0080917A00806480937A0080917B
159	:1009E0007A0086FDFFCCF809178002091790090E01C
160	:1009F000922B08951F93CF93DF93182FEB0161E0A3
161	:100A00000E94A003209711F460E004C0CF3FD105FD
162	:100A100039F461E0812FDF91CF911F910C94D903BC
163	:100A2000E12FF0E0E450FF4FE491E330F9F048F4B7
164	:100A3000E130B9F0E230A1F584B5806284BDC8BD73
165	:100A400034C0E630E9F0E73019F1E43049F580913F

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.hex	
Página 4 de 6	
166	:100A50008000806280938000D0938B00C0938A00D6
167	:100A600024C084B5806884BDC7BD1FC0809180004C
168	:100A7000806880938000D0938900C093880015C05F
169	:100A80008091B00080688093B000C093B3000DC027
170	:100A90008091B00080628093B000C093B40005C024
171	:100AA000C038D1050CF0B5CFAFCFD91CF911F91FA
172	:100AB00008950E94A2040E94D100C8E4D7E00E94D9
173	:100AC00073032097E1F30E944807F9CFCF92DF929A
174	:100AD000EF92FF920F931F93CF93DF936C017A01F4
175	:100AE000EB01E60EF71E00E010E0CE15DF0561F029
176	:100AF0006991D601ED91FC910190F081E02DC60144
177	:100B00000995080F191FF1CFC801DF91CF911F91EF
178	:100B10000F91FF90EF90DF90CF90089561157105D0
179	:100B200081F0DB010D900020E9F7AD014150510942
180	:100B3000461B570BDC01ED91FC910280F381E02D07
181	:100B4000099480E090E008950C948E05DC01ED910D
182	:100B5000FC910190F081E02D09940F931F93CF93A6
183	:100B6000DF93EC016DE00E94A6058C016AE0CE01E6
184	:100B70000E94A605800F911FDF91CF911F910F91C9
185	:100B800008950F931F93CF93DF93EC010E948E057E
186	:100B90008C01CE010E94AD05800F911FDF91CF9196
187	:100BA0001F910F9108958F929F92AF92BF92CF9213
188	:100BB000DF92EF92FF920F931F93CF93DF93CDB706
189	:100BC000DEB7A1970FB6F894DEBF0FBECDBF7C0194
190	:100BD000C42EE52FCB01D22E19A221E02D1510F045
191	:100BE0002AE0D22E8E010F5D1F4F8D2C912CA12C4F
192	:100BF000B12C6C2D7E2FA50194010E94C1088C2D73
193	:100C0000D29E80191124015011098A3014F4805D9C
194	:100C100001C0895CF801808321153105410551052A
195	:100C200021F0C22EE32FCA01E4CFB801C7010E9410
196	:100C30008E05A1960FB6F894DEBF0FBECDBF9133
197	:100C4000CF911F910F91FF90EF90DF90CF90BF90C9
198	:100C5000AF909F908F900895CF92DF92EF92FF9286
199	:100C60000F931F93CF93DF93EC016A017B01211552
200	:100C7000310579F4E881F9810190F081E02D642F4C
201	:100C8000DF91CF911F910F91FF90EF90DF90CF9068
202	:100C900009942A303105E9F477FF1AC06DE20E9409
203	:100CA000A6058C0144275527BA014C195D096E0928
204	:100CB0007F092AE0CE010E94D305800F911FDF91AA
205	:100CC000CF911F910F91FF90EF90DF90CF900895FB
206	:100CD0002AE0B701A601CE01DF91CF911F910F91BC
207	:100CE000FF90EF90DF90CF900C94D305FC0124850A
208	:100CF0003585F901E05CFF4F80819181F901EE5B60
209	:100D0000FF4F20813181821B930B8F7399270895A8
210	:100D1000FC0184859585FC01E05CFF4F4081518199
211	:100D2000FC01EE5BFF4F208131814217530741F0F8
212	:100D30000190F081E02DE80FF91F808190E0089587
213	:100D40008FEF9FEF0895FC0184859585FC01E05CA1
214	:100D5000FF4F40815181FC01EE5BFF4F20813181CB
215	:100D60004217530779F0A081B181A80FB91F8C9168
216	:100D7000208131812F5F3F4F2F73332731832083B1
217	:100D80009E008958FEF9FEF0895FC0181A18823E3
218	:100D900029F0A489B5898C9186FFFBCF11A2089513
219	:100DA000CF93DF93FC0126853785D901A05CBF4F27
220	:100DB0008D919C91119701968F739927E901CE5BD4

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.hex	
Página 5 de 6	
mar 05 abr 2016 14:09:16 COT	
221	:100DC000DF4F4881598184179507D9F30D90BC9165
222	:100DD000A02DA20FB31F6C93A685B785A05CBF4F53
223	:100DE00011969C938E93A689B7892C9181E090E00F
224	:100DF000078C02C0880F991F0A94E2F7282B2C93C6
225	:100E000081E081A30488F589E02D8081806480835E
226	:100E100081E090E0DF91CF91089508951F920F92A5
227	:100E20000FB60F9211242F933F934F938F939F935D
228	:100E3000EF93FF938091C00082FD1CC04091C600DB
229	:100E40008091D3019091D40101968F7399272091BD
230	:100E5000D5013091D6018217930771F0E091D3014B
231	:100E6000F091D401ED56FE4F40839093D4018093CE
232	:100E7000D30102C08091C600FF91EF919F918F91A5
233	:100E80004F913F912F910F900FBE0F901F9018958B
234	:100E90008DE291E00E947606892B11F00C940D07EB
235	:100EA00008951F920F920FB60F9211242F933F9324
236	:100EB0008F939F93EF93FF9320918F013091900137
237	:100EC00080919101909192012817390731F4809116
238	:100ED000C1008F7D8093C10014C0E0919101F09119
239	:100EE0009201E15BFE4F20818091910190919201EE
240	:100EF00001968F739927909392018093910120938B
241	:100F0000C600FF91EF919F918F913F912F910F908C
242	:100F10000FBE0F901F901895CF92DF92EF92FF9225
243	:100F2000CF93DF93EC016A017B01411581EE5807F5
244	:100F300061057105F9F0EC89FD8981F090E008A078
245	:100F400002C0880F991F0A94E2F7808360E079E07D
246	:100F50008DE390E0A70196010E94C108215031095C
247	:100F6000410951095695479537952795211580E1F7
248	:100F7000380798F0EC89FD89108260E874E88EE10A
249	:100F800090E0A70196010E94C10821503109410952
250	:100F900051095695479537952795E889F989308302
251	:100FA000EA89FB89208319A2EE89FF89408181E0CB
252	:100FB00090E09C010C8C02C0220F331F0A94E2F7D0
253	:100FC000422B4083EE89FF8940819C010D8C02C039
254	:100FD000220F331F0A94E2F7422B4083EE89FF89E8
255	:100FE00040819C010E8C02C0220F331F0A94E2F74D
256	:100FF000422B4083EE89FF8920810F8C02C0880F2D
257	:10100000991F0A94E2F7809582238083DF91CF9124
258	:10101000FF90EF90DF90CF900895109230011092E2
259	:101020002F0188EE93E0A0E0B0E08093310190932F
260	:101030003201A0933301B09334018DE091E090939D
261	:101040002E0180932D0183E991E090933A018093E2
262	:1010500039018FE491E090933C0180933B0185EC52
263	:1010600090E090933E0180933D0184EC90E090935A
264	:10107000400180933F0180EC90E090934201809387
265	:10108000410181EC90E0909344018093430182EC14
266	:1010900090E0909346018093450186EC90E0909318
267	:1010A00048018093470184E08093490183E0809365
268	:1010B0004A0187E080934B0185E080934C0181E0F9
269	:1010C00080934D0108954F925F926F927F92AF92FD
270	:1010D000BF92CF92DF92EF92FF920F931F93CF9325
271	:1010E000DF93CDB7DEB729013A01621B730B840B86
272	:1010F000950B29893A894B895C892A193B094C0946
273	:101100005D09298B3A8B4B8B5C8B9B01AC01698908
274	:101110007A898B899C890E94B108E418F508060930
275	:101120001709A80197010E94E3082A0D3B1D4C1DD9

/home/diego/Documentos/Free-infrarossi/free_infrarossi/bin/firmware/infrarossi.hex
Página 6 de 6 mar 05 abr 2016 14:09:16 COT

276	:101130005D1DCA01B901DF91CF911F910F91FF9001
277	:10114000EF90DF90CF90BF90AF907F906F905F90C7
278	:101150004F9008950E940809B7FF0895821B930BD2
279	:101160000895DB018F939F930E941709BF91AF9160
280	:10117000A29F800D911DA39F900DB29F900D1124F1
281	:101180000895A1E21A2EAA1BBB1BFD010DC0AA1FC8
282	:10119000BB1FEE1FFF1FA217B307E407F50720F0E0
283	:1011A000A21BB30BE40BF50B661F771F881F991F5B
284	:1011B0001A9469F760957095809590959B01AC01A4
285	:1011C000BD01CF010895052E97FB1EF400940E94E7
286	:1011D000FA0857FD07D00E94C10807FC03D04EF45F
287	:1011E0000C94FA0850954095309521953F4F4F4FFC
288	:1011F0005F4F089590958095709561957F4F8F4FC3
289	:101200009F4F0895EE0FFF1F0590F491E02D099474
290	:10121000A29FB001B39FC001A39F700D811D112437
291	:10122000911DB29F700D811D1124911D08950E9482
292	:101230000809A59F900DB49F900DA49F800D911D4E
293	:0812400011240895F894FFCF7A
294	:1012480001000000030001200000000000D0066635
295	:0A125800057606A3068806C5060009
296	:00000001FF
297	

APÉNDICE F. COTIZACIÓN COSTOS DE MATERIALES, PROYECTO INFRAROSSI

**Apéndice F. COTIZACIÓN COSTOS DE
MATERIALES, PROYECTO INFRAROSSI**

Tabla 6

**COTIZACIÓN COSTO DE MATERIALES, PROYECTO
INFRAROSSI**

Materiales	COTIZACIÓN COSTOS DE MATERIALES, PROYECTO INFRAROSSI						
	Cantidad	V/U pesos Col	V/T pesos Col	V/U Dolares	V/T Dolares		
Tip 122	1	2000	2000	0,606	0,606		
Transistor LM7805cv	1	3000	3000	0,909	0,909		
Diodo led infrarrojo emisor 950 nm, 5mm	1	800	800	0,242	0,242		
Diodo led infrarrojo emisor 850 nm, 3mm	1	800	800	0,242	0,242		
Diodo led infrarrojo receptor 5mm	1	600	600	0,182	0,182		
Resistencia 500 ohms, medio W	3	100	300	0,030	0,091		
Resistencia de 300 ohms, medio W	2	100	200	0,030	0,061		
Resistencia de 5 Komhs, medio W	2	100	200	0,030	0,061		
Resistencia 1 Komhs, medio W	2	100	200	0,030	0,061		
Transmisión de engranaje con motor CC	1	9000	9000	2,727	2,727		
Base de 28 pines para integrados	1	1000	1000	0,303	0,303		
Modulo bluetooth HC-05	1	30000	30000	9,091	9,091		
Microcontrolador atmega 328P-PU	1	8000	8000	2,424	2,424		
Condensador electrolítico 10 microF	2	200	400	0,061	0,121		
Condensador ceramico 12 picoF	3	100	300	0,030	0,091		
Baquela virgen 25x30cm ²	1	5000	5000	1,515	1,515		
Interruptor tipo switch	1	1000	1000	0,303	0,303		
Switch tipo cuadrado	1	600	600	0,182	0,182		
Cremallera eléctronica tipo hembra 24 pines	1	2000	2000	0,606	0,606		
Disipador térmico	2	1000	2000	0,303	0,606		
Cristal oscilador 12 Mhz	1	2000	2000	0,606	0,606		
Diodo regulador 1N4001	1	100	100	0,030	0,030		
Llanta diametro 5cm	4	5000	20000	1,515	6,061		
Láser 650 nm	1	5000	5000	1,515	1,515		
Conectores 9V batería cuadrada	2	1000	2000	0,303	0,606		
Batería cuadrada 9V recargable	2	20000	40000	6,061	12,121		
Cables de conexión tipo macho-hebra docena	2	8000	16000	2,424	4,848		
Lamina Madeflex, anchor 3mm, m ²	1	9000	9000	2,727	2,727		
Oxido ferroso media libra	1	10000	10000	3,030	3,030		
Cautín	1	25000	25000	7,576	7,576		
Soldadura 60-40 al 3 % metro	15	600	9000	0,182	2,727		
USB Bluetooth	1	20000	20000	6,061	6,061		
Pistola de silicona	1	15000	15000	4,545	4,545		
Silicona barra	4	800	3200	0,242	0,970		
Tarjeta arduino uno	1	50000	50000	15,152	15,152		
SOFTWARE FREEinfraROSSI	1	0	0	0,000	0,000		
		VALOR TOTAL	293700 PESOS	89 DOLARES			
ADICIONAL Ordenador con Ubuntu 14.04LTS							
ADICIONAL Rejilla de difracción 100 lineas/mm							
ADICIONAL Octavos cartulina diferentes colores							
ADICIONAL Espejo							

Nota. Todos los valores en los precios de la moneda local pueden variar por cambios en el valor del dólar; El costo total de los materiales para el diseño e implementación del modulo motorizado infraROSSI, la fuente FUENinfraROSSI, es de 89 dolares. El precio mantiene su estatus de económico debido a que no se incluye el costo del software por su naturaleza freesoftware y opensource.