 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01- 27

SISTEMA DE COMUNICACIÓN POR LUZ VISIBLE UTILIZANDO MODULACIÓN POR COLOR SOBRE DISPOSITIVOS MÓVILES

Diana María Morales Morales

Tecnología en Sistemas de Información

Francisco Eugenio López Giraldo

INSTITUTO TECNOLÓGICO METROPOLITANO

2017

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

RESUMEN

Los sistemas de comunicación por luz visible permiten transferir información mediante la modulación de la intensidad de la luz visible del espectro. Como fuentes de luz se usan los dispositivos de iluminación que, al día de hoy son de uso común, como, por ejemplo, los diodos emisores de luz LED. Los LED tienen una gran ventaja como elemento de emisión de señales ya que son eficientes en el uso de energía, conmutan a gran velocidad y pueden seguir cumpliendo con su principal uso que es la iluminación de interiores. Dentro de las formas en que se modula la luz emitida por un LED, está la modulación por color. Esta es una técnica empleada para codificar datos mediante el cambio del color generado por diodos emisores de luz RGB, que tienen la virtud de emitir más información por unidad de tiempo, usando constelaciones de mayor orden y pueden mantener constante la intensidad de la luz (por lo menos virtualmente para el ojo humano), logrando mantener un tono de iluminación blanco y agradable a la vista.

En este proyecto de investigación se propone un sistema de comunicación por luz visible (VLC) empleando la modulación por color y dispositivos móviles con sistema operativo Android con el fin de controlar el color generado por la pantalla de una Tablet. Se realizó un experimento basado en un transmisor (Tablet) y un receptor (Sensor RGB y sistema de desarrollo Arduino uno). Se logró obtener los rangos de valores de intensidad entregados por el sensor RGB para determinar los intervalos en los que cada símbolo que conforma la constelación 4-CSK, está definido.

Palabras Claves: VLC, CSK, ANDROID, COMUNICACIONES OPTICAS

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01- 22

RECONOCIMIENTOS

Agradezco a Dios por iluminarme cada día y por darme la fortaleza para afrontar las dificultades que se presentan. A mi familia por el apoyo incondicional.

A los docentes Francisco López, Andrés Betancur y Roger Martínez. Por compartir su conocimiento conmigo y guiarme en este mundo de la investigación.

A los estudiantes Juan David Navarro Restrepo y Juan David Rojas Usuga, los cuales fueron un gran apoyo para dar cumplimiento a este proyecto.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

ACRÓNIMOS

CSK	Color Shift Keying
ID	Indentification
IDE	Integrated Development Environment
LEDs	Light Emitting Diodes
RF	Radio Frequency
RGB	Red, Green, Blue
SDK	Software Development Kit
VLC	Visible Light Communication

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

TABLA DE CONTENIDO

Contenido

RESUMEN	2
RECONOCIMIENTOS	3
ACRÓNIMOS	4
INTRODUCCIÓN	6
1. MARCO TEÓRICO.....	8
2. METODOLOGÍA.....	10
3. RESULTADOS Y DISCUSIÓN.....	14
4. CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO.....	17
REFERENCIAS	18
APÉNDICE	19

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

INTRODUCCIÓN

En la actualidad los sistemas de comunicación inalámbricas más empleados tienen como fundamento señales radiadas por ondas electromagnéticas (Barnes, Frank and Greenebaum, 2016). Lo anterior está generando serios inconvenientes en: saturación del espectro electromagnético debido al aumento exponencial de usuarios en la telefonía celular (Barnes, Frank and Greenebaum, 2016); problemas de interferencia con equipos de medición de alta sensibilidad a señales **RF** empleado en lugares como hospitales, aviones, entre otros. Adicionalmente estudios científicos demuestran que las ondas de radio afectan negativamente la salud de los humanos a ciertos niveles de potencia (Fang et al., 2017).

Por lo anterior, en la actualidad se está trabajando un sistema de comunicación que emplea la luz visible del espectro electromagnético para transferir información a través del espacio libre. Dicho sistema recibe el nombre: Comunicación por luz visible **VLC** (Visible Light Communication). Adicionalmente el estándar IEEE 802.15.7 propuso en el año 2011 una nueva técnica de modulación por color denominada **CSK** (Color Shift Keying), la cual permite codificar datos mediante la variación del color generado por diodos generadores de luz rojo, verde y azul.

En este trabajo de investigación se propone un sistema de comunicación visible basado en la modulación por color y dispositivos móviles, orientado a controlar el color generado por la pantalla de una Tablet. Inicialmente se desarrolló un prototipo basado en un transmisor (Tablet) y un receptor (sensor **RGB** y sistema de desarrollo Arduino uno).

Inicialmente se diseñó una aplicación para sistema operativo Android empleando el entorno de desarrollo (**IDE**) Android Studio, el cual permitió controlar el color generado por la pantalla de una Tablet donde cada color corresponde a un símbolo de la constelación 4-CSK, con lo cual se logró emitir información hacia el receptor para posteriormente decodificarla.

por otra parte, se desarrolló el receptor creando un montaje conformado por un sensor **RGB** y un sistema de desarrollo Arduino uno, en este se pudo decodificar la información

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

emitida por la pantalla asignándole a cada color un dato binario, el cual fue interpretado en dicho sistema por la señal digital entregada por dicho sensor.

Este trabajo se estructura como sigue: En el capítulo 1 se muestra el Marco teórico en el cual se trabajó los conceptos más importantes de los sistemas involucrados en este proyecto; en el capítulo siguiente se aborda el desarrollo metodológico del trabajo propuesto; en el capítulo 3 se muestran los resultados y discusiones del experimento desarrollado; finalmente se muestran las conclusiones, recomendaciones y trabajos futuros.

OBJETIVOS

OBJETIVO GENERAL

Diseñar e implementar un sistema de comunicación por luz visible mediante la programación de dispositivos móviles con sistema operativo ANDROID y la modulación por color CSK, que permita transmitir datos binarios a través de la pantalla de un dispositivo móvil (Tablet o Smartphone).

OBJETIVOS ESPECÍFICOS

- Comprender la forma en que el sistema operativo ANDROID interviene en la operación de la pantalla de un celular o Tablet mediante el estudio de tutoriales y conceptos básicos de programación.
- Construir una Interfaz electrónica mediante el uso de un sensor RGB y microcontrolador Arduino UNO, que permita la recepción de la luz proveniente de la modulación por color de la pantalla de la Tablet para establecer las medidas adecuadas de intensidad para cada símbolo 4-CSK.
- Establecer un algoritmo de decodificación de la modulación por color mediante el IDE de Arduino para identificar los datos transmitidos por la modulación 4-CSK.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

1. MARCO TEÓRICO

La comunicación por luz visible (Visible Light Communication – **VLC**) es una tecnología de comunicación de corto alcance, que está en constante investigación. Como su nombre lo indica, esta forma de comunicación ocurre en el espectro visible (400nm-700nm). Una de las ventajas de los sistemas **VLC** es que al emplear el espectro visible para la transmisión de datos, no se tienen restricciones como es el caso de las comunicaciones **RF**, que por muy buenas que sean para transmitir información a grandes distancias, por su forma de propagación, no pueden ser utilizadas en todos los lugares existentes debido a interferencias con equipos y posibles afectaciones en el tejido humano según algunos estudios (Barnes, Frank and Greenebaum, 2016) .

En cuanto al modo de transmisión, los sistemas **VLC** han generado más expectativa con la evolución de los diodos emisores de luz (Light Emitting Diodes – **LEDs**), los cuales son reconocidos por su eficiencia, bajo consumo de energía, estabilidad, forma compacta y vida útil (Jin et al., 2016). Adicionalmente, la tecnología empleada en la construcción del receptor está orientada a fotodiodos de alta velocidad, donde algunas aplicaciones empleando filtros ópticos en el rojo, verde y azul **RGB** (Red, Green, Blue) que permiten trabajar con múltiples fuentes ópticas, lo cual es ventajoso para el sistema **VLC**. Dicho sensor se conoce comúnmente con el nombre de sensor **RGB**.

El sensor **RGB** es un dispositivo óptico que emplea filtros en el **RGB** a diferentes longitudes y fotodiodos para transformar las señales lumínicas incidentes en fotocorrientes y de esta manera permite interpretar los datos transmitidos. Actualmente, dicho sensor es utilizado en una amplia variedad de aplicaciones, desde el control de sistemas de iluminación mediante la discriminación del color; el diseño de mecanismos para la medición de posición angular de alta resolución, medición de concentración de algas dentro de un fotobiorreactor, hasta su utilización en sistemas de comunicaciones por luz visible. Adicionalmente, dado a las características físicas de funcionamiento de dicho sensor, se está explorando su uso en comunicaciones ópticas en medios no guiados, mediante técnicas de codificación por color, la cual es conocida como **CSK** (color shift keying), propuesta en el

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

estándar de la IEEE 802.15.7 (Standard & Society, 2011) para sistemas de comunicación por luz visible que utiliza luminarias basadas en **LEDs RGB**. **CSK** utiliza la mezcla de colores para codificar dígitos binarios, lo cual permite maximizar la capacidad del canal y optimizar el espectro óptico generado por el LED RGB (Monteiro & Hranilovic, 2014). Debido a que el canal VLC basado en **CSK**, es modelado como modulación de intensidad y detección directa, cada componente de potencia óptica **RGB** P_r, P_g, P_b transmitida y la fotocorriente inducida en el fotodiodo r_r, r_g, r_b se relaciona entre si mediante la matriz de ganancias H.

$$H = \begin{bmatrix} h_{rr} & h_{rg} & h_{rb} \\ h_{gr} & h_{gg} & h_{gb} \\ h_{br} & h_{bg} & h_{bb} \end{bmatrix} \quad (1)$$

Donde $h_{i,j}$ representa la fotocorriente generada por el sensor i, debido a la potencia óptica aportada por el LED j. Seguidamente, para estimar el vector de potencia óptica transmitido, se emplea la matriz inversa de ganancias multiplicada por el vector de fotocorriente estimado con cada fotodiodo.

$$\begin{bmatrix} \bar{R} \\ \bar{G} \\ \bar{B} \end{bmatrix} = \begin{bmatrix} h_{rr} & h_{rg} & h_{rb} \\ h_{gr} & h_{gg} & h_{gb} \\ h_{br} & h_{bg} & h_{bb} \end{bmatrix}^{-1} \begin{bmatrix} h_r \\ h_g \\ h_b \end{bmatrix} \quad (2)$$

Finalmente, cada símbolo de la constelación CSK es representado por las tres componentes ópticas estimadas, las cuales se representan por \bar{S}_i .

$$\bar{S}_i = [\bar{R}, \bar{G}, \bar{B}] \quad (3)$$

Mediante la técnica de codificación CSK mostrada anteriormente, es posible pensar en aplicaciones VLC que involucren dispositivos móviles mediante el control del color generado

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

por las pantallas de dichos dispositivos. Es por ello que en adelante se dará una introducción al sistema operativo Android, el cual tiene como función administrar los recursos de los celulares y entre ellos la pantalla.

Android es un sistema operativo basado en Linux creado inicialmente para dispositivos móviles, y posteriormente hacia otros dispositivos como: tv, autos, Tablet. En la actualidad, la gran mayoría de dispositivos móviles funcionan con el sistema operativo ANDROID, lo cual se convierten en una herramienta interesante para ser explorada en el ámbito de la investigación, y específicamente se está empleando en los sistemas de comunicación por luz visible (Fang et al., 2017).

Se puede crear en los siguientes entornos de desarrollo (IDE). En primer lugar, se tiene el IDE ANDROID STUDIO creado por Google y al que hace referencia toda la documentación oficial existente. En este IDE, se desarrolla en Java utilizando referencias al Android SDK (Software Development Kit), lo cual facilita acceder a elementos de la interfaz de usuario para visualizarlas en una vista de diseño. Entre otros IDE tenemos ECLIPSE, XAMARIN, AIDE, PYTHON, B4A, UNITY, entre otros.

Por otra parte, la plataforma de hardware libre Arduino UNO está basado en un microcontrolador ATmega 328 y un entorno de desarrollo de código abierto, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

2. METODOLOGÍA

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

El sistema VLC-ANDROID fue realizado en el laboratorio de Visión Artificial y Fotónica del ITM. El sistema propuesto está conformado por dos partes; una transmisora y una receptora. La primera parte consta de una Tablet Samsung Galaxy modelo sm-p605m con sistema operativo ANDROID y la segunda parte está compuesta por un sensor RGB ADJD-S311 del fabricante AVAGO TECHNOLOGIES, una tarjeta de desarrollo Arduino UNO y una computadora para la lectura de los datos, tal como se muestra en el diagrama de bloques de la *Figura 1*.

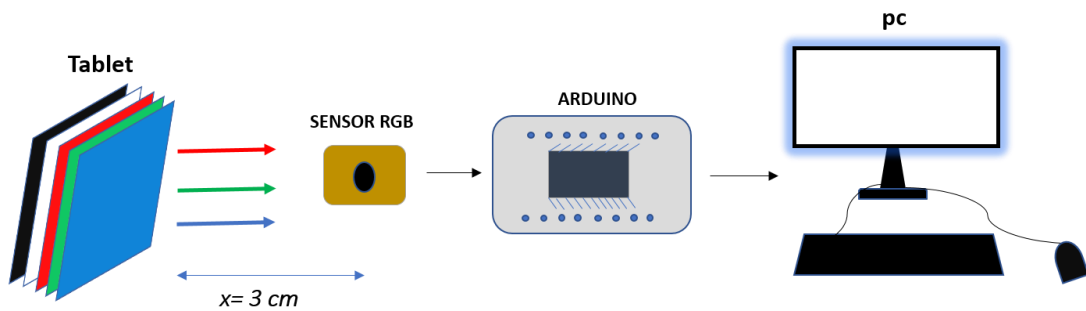


Figura 1. Sistema experimental propuesto

En primer lugar, se desarrolló una aplicación en el entorno de desarrollo (IDE) ANDROID STUDIO, la cual consistió en el control de los colores generados por la pantalla para codificar información binaria, dicha técnica se conoce como CSK (Color Shift Keying) (Monteiro & Hranilovic, 2014). CSK está basada en la codificación de datos binarios mediante el mapeo de puntos cromáticos en el plano de cromaticidad CIE-1931.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

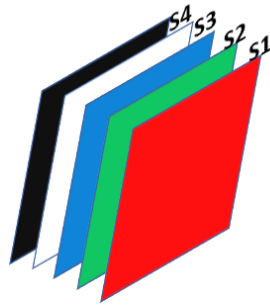


Figura 2. Esquema de envío de datos para la constelación 4-CSK.

En la *Figura 2* se ilustra el esquema utilizado para la constelación 4-CSK referente al envío de datos, donde cada color recibe el nombre de símbolos. En este experimento, se enviaron ocho símbolos (azul, rojo, verde, blanco, verde, azul, rojo, blanco). La aplicación VLC-ANDROID tiene una duración de 6,6 segundos, durante el cual se realiza dicha transmisión donde cada símbolo tuvo una duración de 0,2 segundos. Adicionalmente se dejaron 5 segundos como tiempo suficiente para que el usuario pueda ubicar la Tablet en el montaje antes del inicio del envío de símbolos.

Luego de hacer la aplicación en Android, se procedió a ubicar la pantalla de la Tablet a 3cm del sensor RGB, para garantizar que la Tablet siempre estuviese a la misma distancia, se diseñó una caja como se muestra en la *Figura 3*, la cual contiene el sensor RGB junto con la tarjeta de desarrollo Arduino UNO. Dicha caja del montaje cuenta con un orificio que permita recibir la señal lumínica generada por la Table, permitiendo que el sensor no reciba fuentes de iluminación externas las cuales son consideradas como ruido.

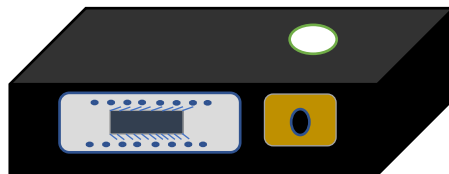


Figura 3. Montaje experimental para el receptor VLC

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Luego de tener el montaje experimental antes mencionado, se procedió a realizar el siguiente experimento. En la Tablet se configuro la emisión de un color en particular y mediante el sensor RGB se determino la respuesta generada por dichos canales. El Procedimiento anterior se realizó para la transmisión del color rojo, verde y azul, donde se escogieron 10 valores de cada color y se promediaron para luego construir la matriz de ganancias del canal óptico. Adicionalmente, se utilizó el software MATLAB para obtener la matriz inversa de ganancias, con la cual fue posible estimar las potencias ópticas emitidas y de esta forma determinar los datos binarios transmitidos. Finalmente, el sensor RGB entrega los datos digitalizados al Arduino UNO por medio de un bus de comunicación I2C. La resolución empleada para representar cada salida del sensor RGB es de 10 bits (0 - 1024).

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

3. RESULTADOS Y DISCUSIÓN

El prototipo de comunicación por luz visible basado en la modulación por color y en el sistema operativo Android es mostrado en la *Figura 4*. Como se puede observar el sistema consta de un transmisor conformado por una Tablet con sistema operativo Android y el receptor consta de un sensor RGB y un sistema de desarrollo Arduino UNO. Mediante el sistema anterior, los datos generados mediante combinaciones de colores de la pantalla de la Tablet viajan a través del espacio libre hacia el receptor. El receptor convierte las señales lumínicas en corrientes eléctricas las cuales son procesadas por el Arduino y finalmente son llevadas a un computador donde posteriormente serán analizadas.



Figura 4. Prototipo del sistema VLC propuesto

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Por otra parte, se procedió a realizar las medidas de ganancias del canal óptico. En la matriz mostrada en la *Ecuación 4* se muestra el resultado del aporte que realiza cada color incidente en el sensor RGB, que conforman la matriz de ganancias ópticas H.

$$H = \begin{bmatrix} 131.7 & 93.9 & 24.8 \\ 32.1 & 325.6 & 93.5 \\ 20.8 & 93.9 & 228 \end{bmatrix} \quad (4)$$

De manera particular la inversa de la matriz H mostrada en la *Ecuación 5* es realmente la que importa para la estimación de las potencias ópticas transmitidas que contienen los datos o información.

$$H^{-1} = \begin{bmatrix} 0.0082 & -0.0024 & 0.0001 \\ -0.0007 & 0.0037 & -0.0014 \\ -0.0005 & -0.0013 & 0.0050 \end{bmatrix} \quad (5)$$

Con respecto al diseño de la constelación para modulación CSK, en la *Tabla 1* se muestra de manera resumida los símbolos utilizados para la constelación 4-CSK y código binario asociado a cada símbolo.

Binario	CSK	Símbolo
0 0	R	S_1
0 1	G	S_2
1 0	B	S_3
1 1	R+G+B=W	S_4

Tabla 1. Constelación 4-CSK

Se realizaron pruebas de transmisión y recepción de los símbolos S_i para la constelación 4-CSK diseñada, donde $i = 1,2,3,4$. En este caso los datos de cada símbolo recibido fueron multiplicados por la matriz inversa (H^{-1}) del canal, donde los resultados asociados a cada símbolo estimado \bar{S}_i fueron mostrados en la ecuación 3.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Luego de realizar el experimento de transmisión se lograron los siguientes resultados para cada símbolo, donde se puede apreciar los rangos para cada valor RGB entregado por el sensor.

$$\bar{S}_1 = 1.4 \leq \bar{R} \leq 2.5, 0 \leq \bar{G} \leq 0.15, 0 \leq \bar{B} \leq 0.1$$

$$\bar{S}_2 = 0 \leq \bar{R} \leq 0.1, 0.7 \leq \bar{G} \leq 1.3, 0 \leq \bar{B} \leq 0.1$$

$$\bar{S}_3 = 0 \leq \bar{R} \leq 0.3, 0 \leq \bar{G} \leq 0.1, 1 \leq \bar{B} \leq 1.3$$

$$\bar{S}_4 = 0.5 \leq \bar{R} \leq 3.1, 0.9 \leq \bar{G} \leq 1.5, 0 \leq \bar{B} \leq 1.3$$

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22


4. CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO

En este trabajo de investigación se logro el diseño de un prototipo para un sistema de comunicación por luz visible empleando la modulación por color y dispositivos móviles con sistema operativo Android. Se logro identificar los rangos de los valores digitales entregados por el sensor RGB para decodificar cada uno de los símbolos que conforman la constelación 4-CSK de la modulación por color.

Se recomienda a los usuarios interesados en realizar el experimento propuesto en este trabajo, considerar el mantener en un 50% el nivel de brillo de la pantalla, ya que esta puede afectar negativamente la interpretación del color en el sistema de recepción.

A modo de trabajo futuro, se propone un estudio sobre el diseño de un protocolo de enlace para el envío de tramas de datos representada por símbolos de la modulación por color, que facilite la sincronización del transmisor y del receptor con el fin de obtener un mecanismo de comunicación que permita la correcta interpretación de los datos transmitidos.

La principal aplicación de este sistema de comunicación propuesto radica en escenarios de baja velocidad, tal como: Sistema de control de acceso, sistema VLC ID, Robótica, entre otros.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

REFERENCIAS

- Barnes, Frank and Greenebaum, B. (2016). Some Effect of weak magnetic fields on biological systems. *Int.J.Neurosci.*, 67(1–4), 125–126.
- Fang, J., Yang, Z., Long, S., Wu, Z., Zhao, X., Liang, F., ... Chen, Z. (2017). High-speed indoor navigation system based on visible light and mobile phone. *IEEE Photonics Journal*, 9(2).
<https://doi.org/10.1109/JPHOT.2017.2687947>
- Jin, Y., Fu, Y., Hu, Y., Chen, L., Wu, H., Ju, G., ... Wang, T. (2016). A high color purity deep red emitting phosphor SrGe4O9: Mn4+ for warm white LEDs. *Powder Technology*, 292, 74–79.
<https://doi.org/10.1016/j.powtec.2016.01.017>
- Monteiro, E., & Hranilovic, S. (2014). Design and Implementation of Color-Shift Keying for Visible Light Communications. In *Journal of Lightwave Technology* (Vol. 32, pp. 2053–2060).
<https://doi.org/10.1109/JLT.2014.2314358>
- Standard, I., & Society, I. C. (2011). *IEEE Standard for Local and metropolitan area networks — Part 15 . 7 : Short-Range Wireless Optical Communication Using Visible Light IEEE Computer Society S ponsored by the.*

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

APÉNDICE

APÉNDICE A

A continuación se presentan las librerías, las variable y el código creado en el IDE Android Studio.

```

//INICIO DE ACTIVITY

package com.vlcsk.rgb;

import android.graphics.Color;
import android.os.Bundle;
import android.support.design.widget.FloatingActionButton;
import android.support.design.widget.Snackbar;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.view.View;
import android.view.Menu;
import android.view.MenuItem;

import android.widget.LinearLayout; //Librería que permite controlar el la pantalla
import android.os.CountDownTimer; // librería empleada para crear
funciones de tiempo de manera descendente, con esta librería se crea la
base de tiempo para el cambio de color RGB

public class RGB extends AppCompatActivity {

    // Definición de variables globales
    private LinearLayout fondo;
    private boolean fr=true;

    int Tfinal = 5600; //variable para identificar el tiempo que dura la aplicación
    int Tintervalo = 200; // variable para identificar el tiempo en ms del intervalo
de encendido y apagado del cambio de color
    int numeroPulsos =0; // cantidad de veces que encendera la pantalla
    int contador=0;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_rgb);

```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

fondo= (LinearLayout)findViewById(R.id.fondo);

Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);

FloatingActionButton fab = (FloatingActionButton)
findViewById(R.id.fab);
fab.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Snackbar.make(view, "Replace with your own action",
Snackbar.LENGTH_LONG)
            .setAction("Action", null).show();
    }
});
}
public void eventoClick(View view)//metodo que al dar click en el boton 4csk permitira el inicio del programa
{
    switch (view.getId())
    {
        case R.id.cuatroCSK:
            transmitecodigo();
            break;
    }
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
//Inflate the menu; this adds items to the action bar if it is present.
getMenuInflater().inflate(R.menu.menu_rgb, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_settings) {
        return true;
    }

    return super.onOptionsItemSelected(item);
}

```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

/**Programación de la función para pilotear la pantalla y transmitir los datos ****

// en esta función se crea un timer descendente, mediante el cual se crea la
base de tiempo para intercambiar entre los colores RGB

```

```

public void transmitecodigo()
{
    // Crear un contador descendente. (duracion ms, intervalo ms)
    new CountdownTimer(Tfinal,Tintervalo)
    {
        // cada Tintervalo en ms se ejecuta el siguiente código
        @Override
        public void onTick(long millisUntilFinished) {
            contador++;
            if (contador >= 25) {
                switch (numeroPulsos) {
                    case 0:
                        fondo.setBackgroundColor(Color.BLUE);
                        numeroPulsos++;
                        break;
                    case 1:
                        fondo.setBackgroundColor(Color.RED);
                        numeroPulsos++;
                        break;
                    case 2:
                        fondo.setBackgroundColor(Color.GREEN);
                        numeroPulsos++;
                        break;
                    case 3:
                        fondo.setBackgroundColor(Color.WHITE);
                        numeroPulsos++;
                        break;
                    case 4:
                        fondo.setBackgroundColor(Color.GREEN);
                        numeroPulsos++;
                        break;
                    case 5:
                        fondo.setBackgroundColor(Color.BLUE);
                        numeroPulsos++;
                        break;
                    case 6:
                        fondo.setBackgroundColor(Color.RED);
                        numeroPulsos++;
                        break;
                    case 7:
                        fondo.setBackgroundColor(Color.WHITE);
                        numeroPulsos++;
                        break;
                }
            }
        }
    }
}

```

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

    }
  }
}
// al finalizar el tiempo programado, se ejecuta el siguiente código
@Override
public void onFinish() {
    fondo.setBackgroundColor(Color.BLACK);
    //coloca el layout de color negro que es con el color que termina el ciclo
    fr = true; //esta linea de codigo solo se ejecuta cuando
                se terminan los 6,6 seg.
    //Ya que activa el if con el que va iniciar el nuevo ciclo.
    numeroPulsos=0;
    contador=0;
}
}.start();
}
} // FIN DEL ACTIVITY

```

APÉNDICE B.

A continuación se presentan las librerías, las variable y el código creado en el IDE Arduino.

```

// Incluir la librería para comunicación I2C
#include <Wire.h>

// Dirección del sensor de color RGB
#define ADJD_S311_ADDRESS 0x74

// Definición de pines
#define RED 0
#define GREEN 1
#define BLUE 2
#define CLEAR 3

// Definición de la lista de registros del sensor RGB
#define CTRL 0x00
#define CONFIG 0x01
#define CAP_RED 0x06
#define CAP_GREEN 0x07

```

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

#define CAP_BLUE 0x08
#define CAP_CLEAR 0x09
#define INT_RED_LO 0xA
#define INT_RED_HI 0xB
#define INT_GREEN_LO 0xC
#define INT_GREEN_HI 0xD
#define INT_BLUE_LO 0xE
#define INT_BLUE_HI 0xF
#define INT_CLEAR_LO 0x10
#define INT_CLEAR_HI 0x11
#define DATA_RED_LO 0x40
#define DATA_RED_HI 0x41
#define DATA_GREEN_LO 0x42
#define DATA_GREEN_HI 0x43
#define DATA_BLUE_LO 0x44
#define DATA_BLUE_HI 0x45
#define DATA_CLEAR_LO 0x46
#define DATA_CLEAR_HI 0x47
#define OFFSET_RED 0x48
#define OFFSET_GREEN 0x49
#define OFFSET_BLUE 0x4A
#define OFFSET_CLEAR 0x4B

// Definición de pines SDA y SCL
int sdaPin = A4;
int sclPin = A5;

// Definición de variables
unsigned char colorCap[4] = {9, 9, 2, 5};
unsigned int colorInt[4] = {2048, 2048, 2048, 2048};
unsigned int colorData[4];
signed char colorOffset[4];

void setup()
{
// configuración del puerto serial

```

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```
Serial.begin(9600);
```

```
// configuracion e inicio del módulo I2C
```

```
Wire.begin();
```

```
delay(1);
```

```
initADJD_S311();
```

```
// calibración del color
```

```
calibrateColor();
```

```
// calibración del capacitor interno del sensor RGB
```

```
calibrateCapacitors();
```

```
}
```

```
void loop()
```

```
{
```

```
// ALGORITMO PROPUESTO PARA LA LECTURA DE SIMBOLOS
```

```
while(1)
```

```
{
```

```
  // Leer datos RGB del sensor
```

```
  getRGBC();
```

```
// MATRIZ INVERSA se determinó en MATLAB
```

```
// 0,0082 -0,0024 0,0001
```

```
// -0,0007 0,0037 -0,0014
```

```
// -0,0005 -0,0013 0,0050
```

```
// Estimación del vector de potencias ópticas transmitido. Se emplea la ecuación  $RGB = \text{inv}(H) * (hi)$ 
```

```
con  $i=1,2,3$ 
```

```
// R            hr
```

```
// G = inv(H) hg
```

```
// B            hb
```

```
float R= (0.0082)*colorData[0]+colorData[1]*(-0.0024)+colorData[2]*(0.0001);
```

```
float G=(-0.0007)*colorData[0]+colorData[1]*(0.0037)+colorData[2]*(-0.0014);
```

```
float B=(-0.0005)*colorData[0]+colorData[1]*(-0.0013)+colorData[2]*(0.0050);
```

```
// Los datos son normalizados entre 0 – 1 y Si los datos son negativos, se les carga cero.
```

```
if (R<0)
```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

R=0;
if (G<0)
G=0;
if (B<0)
B=0;

// Intervalos de valores RGB del sensor para decodificar los símbolos
if((R<=2.5&&R>=1.4)&&(G<=0.15&&G>=0)&&(B<=0.1&&B>=0))
Serial.println("S1 ");

if((R<=0.1&&R>=0)&&(G<=1.3&&G>=0.7)&&(B<=0.1&&B>=0))
Serial.println("S2 ");

if((R<=0.3&&R>=0)&&(G<=0.1&&G>=0)&&(B<=1.3&&B>=1))
Serial.println("S3 ");

if((R<=3.1&&R>=0.5)&&(G<=1.5&&G>=0.9)&&(B<=1.3&&B>=0))
Serial.println("S4 ");
// tiempo para iniciar una nueva lectura de símbolos
delay(50);
}
} // Fin del algoritmo propuesto

// Las funciones que se definen en adelante, son para configurar, leer y escribir el sensor RGB
void printADJD_S311Values()
{
for (int i=0; i<3; i++)
{
Serial.print(colorData[i]);
Serial.print("\t ");
}
Serial.println();

void initADJD_S311()
{

```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

writeRegister(colorCap[RED] & 0xF, CAP_RED);
writeRegister(colorCap[GREEN] & 0xF, CAP_GREEN);
writeRegister(colorCap[BLUE] & 0xF, CAP_BLUE);
writeRegister(colorCap[CLEAR] & 0xF, CAP_CLEAR);

writeRegister((unsigned char)colorInt[RED], INT_RED_LO);
writeRegister((unsigned char)((colorInt[RED] & 0x1FFF) >> 8), INT_RED_HI);
writeRegister((unsigned char)colorInt[BLUE], INT_BLUE_LO);
writeRegister((unsigned char)((colorInt[BLUE] & 0x1FFF) >> 8), INT_BLUE_HI);
writeRegister((unsigned char)colorInt[GREEN], INT_GREEN_LO);
writeRegister((unsigned char)((colorInt[GREEN] & 0x1FFF) >> 8), INT_GREEN_HI);
writeRegister((unsigned char)colorInt[CLEAR], INT_CLEAR_LO);
writeRegister((unsigned char)((colorInt[CLEAR] & 0x1FFF) >> 8), INT_CLEAR_HI);
}
int calibrateClear()
{
int gainFound = 0;
int upperBox=4096;
int lowerBox = 0;
int half;

while (!gainFound)
{
half = ((upperBox-lowerBox)/2)+lowerBox;
//no further halving possible
if (half==lowerBox)
gainFound=1;
else
{
writeInt(INT_CLEAR_LO, half);
performMeasurement();
int halfValue = readRegisterInt(DATA_CLEAR_LO);

if (halfValue>1000)
upperBox=half;
else if (halfValue<1000)

```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

    lowerBox=half;
else
    gainFound=1;
}
}
return half;
}
int calibrateColor()
{
int gainFound = 0;
int upperBox=4096;
int lowerBox = 0;
int half;

while (!gainFound)
{
half = ((upperBox-lowerBox)/2)+lowerBox;
//no further halving possible
if (half==lowerBox)
{
gainFound=1;
}
else {
writeInt(INT_RED_LO, half);
writeInt(INT_GREEN_LO, half);
writeInt(INT_BLUE_LO, half);

performMeasurement();
int halfValue = 0;

halfValue=max(halfValue, readRegisterInt(DATA_RED_LO));
halfValue=max(halfValue, readRegisterInt(DATA_GREEN_LO));
halfValue=max(halfValue, readRegisterInt(DATA_BLUE_LO));

if (halfValue>1000) {
upperBox=half;

```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

}
else if (halfValue<1000) {
    lowerBox=half;
}
else {
    gainFound=1;
}
}
}
return half;
}
void calibrateCapacitors()
{
    int calibrationRed = 0;
    int calibrationBlue = 0;
    int calibrationGreen = 0;
    int calibrated = 0;

    //need to store detect better calibration
    int oldDiff = 5000;

    while (!calibrated)
    {
        // sensor gain setting (Avago app note 5330)
        // CAPs are 4bit (higher value will result in lower output)
        writeRegister(calibrationRed, CAP_RED);
        writeRegister(calibrationGreen, CAP_GREEN);
        writeRegister(calibrationBlue, CAP_BLUE);

        // int colorGain = _calibrateColorGain();
        int colorGain = readRegisterInt(INT_RED_LO);
        writeInt(INT_RED_LO, colorGain);
        writeInt(INT_GREEN_LO, colorGain);
        writeInt(INT_BLUE_LO, colorGain);

        int maxRead = 0;

```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

int minRead = 4096;
int red = 0;
int green = 0;
int blue = 0;

for (int i=0; i<4 ;i ++)
{
    performMeasurement();
    red += readRegisterInt(DATA_RED_LO);
    green += readRegisterInt(DATA_GREEN_LO);
    blue += readRegisterInt(DATA_BLUE_LO);
}
red /= 4;
green /= 4;
blue /= 4;

maxRead = max(maxRead, red);
maxRead = max(maxRead, green);
maxRead = max(maxRead, blue);

minRead = min(minRead, red);
minRead = min(minRead, green);
minRead = min(minRead, blue);

int diff = maxRead - minRead;

if (oldDiff != diff)
{
    if ((maxRead==red) && (calibrationRed<15))
        calibrationRed++;
    else if ((maxRead == green) && (calibrationGreen<15))
        calibrationGreen++;
    else if ((maxRead == blue) && (calibrationBlue<15))
        calibrationBlue++;
}
else

```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

calibrated = 1;

oldDiff=diff;

int rCal = calibrationRed;
int gCal = calibrationGreen;
int bCal = calibrationBlue;
}
}
void writeInt(int address, int gain)
{ if (gain < 4096)
{
byte msb = gain >> 8;
byte lsb = gain;
writeRegister(lsb, address);
writeRegister(msb, address+1);
}
}
void performMeasurement()
{
writeRegister(0x01, 0x00); // start sensing
while(readRegister(0x00) != 0)
; // waiting for a result
}
void getRGBC()
{
performMeasurement();

colorData[RED] = readRegisterInt(DATA_RED_LO);
colorData[GREEN] = readRegisterInt(DATA_GREEN_LO);
colorData[BLUE] = readRegisterInt(DATA_BLUE_LO);
//colorData[CLEAR] = readRegisterInt(DATA_CLEAR_LO);
}
void getOffset()
{
writeRegister(0x02, 0x00); // start sensing

```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

```

while(readRegister(0x00) != 0)
    ; // waiting for a result
//writeRegister(0x01, 0x01); // set trim
//delay(100);
for (int i=0; i<4; i++)
    colorOffset[i] = (signed char) readRegister(OFFSET_RED+i);
}
/* I2C functions... */
// Write a byte of data to a specific ADJD-S311 address
void writeRegister(unsigned char data, unsigned char address)
{
    Wire.beginTransmission(ADJD_S311_ADDRESS);
    Wire.write(address);
    Wire.write(data);
    Wire.endTransmission();
}
// read a byte of data from ADJD-S311 address
unsigned char readRegister(unsigned char address)
{
    unsigned char data;

    Wire.beginTransmission(ADJD_S311_ADDRESS);
    Wire.write(address);
    Wire.endTransmission();

    Wire.requestFrom(ADJD_S311_ADDRESS, 1);
    while (!Wire.available())
        ; // wait till we can get data

    return Wire.read();
}
// Write two bytes of data to ADJD-S311 address and address+1
int readRegisterInt(unsigned char address)
{
    return readRegister(address) + (readRegister(address+1)<<8);
}

```

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-27

FIRMA ESTUDIANTES Diana María Morales Morales

FIRMA ASESOR 

FECHA ENTREGA: 16/11/2017

FIRMA COMITÉ TRABAJO DE GRADO DE LA FACULTAD _____

RECHAZADO___ ACEPTADO___ ACEPTADO CON MODIFICACIONES___

ACTA NO. _____

FECHA ENTREGA: _____

FIRMA CONSEJO DE FACULTAD _____


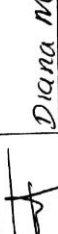
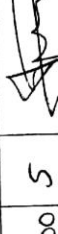
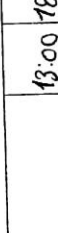



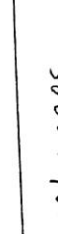
ACTA NO. _____

FECHA ENTREGA: _____

ITM Institución Universitaria		MODALIDAD TRABAJO DE GRADO PRODUCTO OBTENIDO EN TALLERES O LABORATORIOS DEL ITM			Código	FDE 146
Registro de actividades y cumplimiento de horas / Talleres o Laboratorios de DOCENCIA					Versión	02
					Fecha	2014-07-13
Documento de Identidad:		1040261366				
Nombre completo del estudiante:		DIANA MARÍA MORALES MORALES				
Programa académico ITM:		TECNOLOGIA SISTEMAS DE INFORMACION				
Nombre completo del Docente Asesor:		FRANCISCO EUGENIO LOPEZ GIRALDO				
Fecha de iniciación del producto (aaaa/mm/dd):		Fecha de terminación del producto (2017/11/07):				
Nombre Taller o Laboratorio:		LABORATORIO DE VISION ARTIFICIAL Y FOTONICA DEL ITM				
Ubicación:		PARQUE I				
Campus:		FRATERNIDAD				
Fecha A M D	Actividad desempeñada por el estudiante	Hora Ingreso	Hora salida	Total horas	Firma Laboratorista	Firma Estudiante
14-08-31	Lectura de articulos VLC	13:00	14:00	4	<i>[Firma]</i>	Diana Morales
14-09-04	Lectura de la literatura	8:00	11:00	3	<i>[Firma]</i>	Diana Morales
14-09-08	Lectura de la literatura	13:00	14:00	4	<i>[Firma]</i>	Diana Morales
14-09-11	Lectura de articulos	8:00	12:00	4	<i>[Firma]</i>	Diana Morales
14-09-12	Reunion Semillero	14:00	16:00	2	<i>[Firma]</i>	Diana Morales
14-09-15	Programacion codigo Android	13:00	14:00	4	<i>[Firma]</i>	Diana Morales
14-09-18	Programacion codigo Android	8:00	12:00	4	<i>[Firma]</i>	Diana Morales
14-09-19	Reunion Semillero	14:00	16:00	2	<i>[Firma]</i>	Diana Morales
14-09-22	Elabora con montaje experimental	14:00	19:00	5	<i>[Firma]</i>	Diana Morales
14-09-25	Mediciones	4:00	13:00	6	<i>[Firma]</i>	Diana Morales

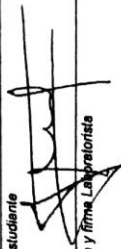
INFORME FINAL DE TRABAJO DE GRADO

Código	FDE 089
Versión	03
Fecha	2015-01-22

14-09-19	mediciones	13:00	18:00	5		Diana Morales
14-10-02	Análisis de los resultados	8:00	12:00	4		Diana Morales
14-10-03	Reunion Semillero	14:00	16:00	2		Diana Morales
14-10-06	Elaboracion del informe	13:00	14:00	4		Diana Morales
14-10-09	Elaboracion del informe	8:00	12:00	4		Diana Morales
14-10-13	Análisis de los resultados en nu+lad	13:00	18:00	5		Diana Morales
14-10-17	Reunion Semillero	14:00	16:00	2		Diana Morales
14-10-19	Elaboracion del informe de los resultados	13:00	16:00	3		Diana Morales
14-10-20	Elaboracion del informe de los resultados	13:00	16:00	3		Diana Morales
14-10-23	Elaboracion del informe de los resultados	8:00	11:00	3		Diana Morales
14-10-24	Elaboracion del informe de los resultados	13:00	18:00	5		Diana Morales
14-10-30	Elaboracion del informe de los resultados	8:00	12:00	4		Diana Morales
14-11-03	Elaboracion del informe de los resultados	13:00	15:00	2		Diana Morales
14-11-04	Elaboracion del informe de los resultados	14:00	16:00	2		Diana Morales
TOTAL HORAS				86		

Diana Morales

Firma Estudiante



Nombre y firma Laboralista

Nombre y firma Profesional Universitario - Centro de Laboratorios