

SEP

SECRETARÍA DE EDUCACIÓN PÚBLICA

SEP
Institutos Tecnológicos

# VI Verano de Formación de Jóvenes Investigadores

## Memoria de Proyectos

2011

Instituto Tecnológico de Colima
del 27 de junio al 27 de julio del 2011

# VI Verano de Formación de Jóvenes Investigadores 2011

Primera edición: 2011
Reservados todos los derechos
© Instituto Tecnológico de Colima
Av. Tecnológico No. 1
Villa de Álvarez, Col. 28976

Impreso y hecho en México

**ISBN: 978-607-95775-0-6**

# COMITÉ ORGANIZADOR

**ING. SATURNINO CASTRO REYES**
Director del Instituto Tecnológico de Colima

**M.C. OSCAR LUIS BARAJAS PASTOR**
Subdirector Académico

**M.C. GILBERTO RENÉ MARTÍNEZ**
Subdirector de Planeación y Vinculación

**M.A. SERGIO RAFAEL UREÑA FUENTES**
Subdirector de Servicios Administrativos

**M.G.T.I. RAMONA EVELIA CHÁVEZ VALDEZ**
Jefa del Departamento de Sistemas y Computación

**DR. JESÚS ALBERTO VERDUZCO RAMÍREZ**
Coordinador General del VI VFJI

**ACADEMIA DE SISTEMAS Y COMPUTACIÓN**
Apoyo Logístico

**LIC. MIGUEL ANGEL VILLASEÑOR URIBE**
Jefe del Departamento de Comunicación

**COMITÉ ORGANIZADOR INVITADO**

**M.C. RICARDO FUENTES COVARRUBIAS**
**M.C. GERARDO FUENTES COVARRUBIAS**

**INSTRUCTORES DE CURSOS**

**M.C. CARLOS CEDILLO NAKAY**
**M.C. Mč NICA VIOLETA SIERRA PEč N** - Redacción de Documentos Científicos

**M.C. RICARDO FUENTES COVARRUBIAS** - Metodología de la Investigación

**M.C. GERARDO FUENTES COVARRUBIAS** - Administración de Proyectos de Investigación

**ANGÉLICA DELGADILLO JIMENEZ**
Secretaria

**C. ALBERTO ISAAC MARTÍNEZ NUÑEZ**

**C. ANGÉLICA TALIA VILLEGAS MORENO**
Sitio Web

**C. VICTORIA CONTRERAS TORRES**
Memoria

# COMITÉ ACADÉMICO

**ARQ. ENRIQUE URIEL RIOS TRUJILLO**
Departamento de Ciencias de la Tierra

**M.C. MARÍA AURORA ANDRADE URSÚA**
Departamento de Ingeniería Química y Bioquímica

**M.C. MARCO AURELIO VÁZQUEZ OLVERA**
Departamento de Ingeniería Industrial

**DRA. ANTONIA MAGAÑA ESCOTO**
Departamento de Ciencias Económico Administrativas

**M.G.T.I. RAMONA EVELIA CHÁVEZ VALDEZ**
Departamento de Sistemas y Computación

**M.C JUAN MANUEL TORRES OCHOA**
Departamento de Ciencias Básicas

**L.I. DÉBORA FABIOLA NAVA HUITRÓN**
Departamento de Servicios Escolares

# MENSAJE DE LOS ORGANIZADORES

El Instituto Tecnológico de Colima de manera similar a todas las instituciones del país se confronta a un proceso que exige el mejoramiento continuo de sus procesos, de su planta docente y su comunidad estudiantil. En particular, el quehacer académico requiere cada vez más de profesores y estudiantes con alto grado de compromiso en la realización de sus tareas cotidianas, que desemboque en una sólida preparación académica y una profunda vocación científica.

Como una respuesta natural a estos retos, la institución ha entrado en procesos de acreditación y certificación de sus procesos académicos y administrativos. En el aspecto de la investigación de manera similar se han implementado un conjunto de acciones tendientes al fomento de esta actividad en la institución. Una de las primeras respuestas es el fomento del trabajo académico colegiado mediante la integración de los profesores en grupos especializados de trabajo, así como el inicio de la conformación de los cuerpos académicos.

Como fruto de este trabajo colegiado, desde el verano de 2006, el departamento de Sistemas y Computación promueve el programa denominado **Verano de Formación de Jóvenes Investigadores (VFJI)**. Este programa contempla una estancia de cinco semanas que totalizan 200 horas de trabajo. Durante esta estancia, los estudiantes trabajan en colaboración con profesores de la planta docente desarrollando proyectos de investigación.

Al participar durante cinco semanas en el lugar de trabajo de un experto y en el marco de proyectos de actualidad, los jóvenes encontrarán una experiencia invaluable que les ayudará a definir su vocación científica, ampliando sus conocimientos y sus opciones para futuras etapas en su formación profesional. Para enriquecer la estancia de verano, a los estudiantes se les capacita en cursos de: Metodología de la Investigación, Manejo Estadístico de Datos, Redacción de Documentos Científicos y Administración de Proyectos de Investigación. Para concluir, los participantes en el VFJI celebran una reunión en la cual presentan los resultados de su investigación y festejan la conclusión de su estancia con una convivencia.

Este documento integra los resultados de las actividades de investigación realizadas en la sexta edición de este programa de verano, mismas que se describen en las secciones siguientes.

Comité Organizador

Villa de Álvarez, Colima Mayo de 2012

# ÍNDICE

**APLICACIÓN PARA CALCULAR LA TRAYECTORIA DE LA LUZ SOLAR** 1
Moreno Pe¶a Ricardo, Puente García Simón Pedro, Rebolledo Peregrino Luis Antonio, Verduzco Ramírez Jesús Alberto.

**DESARROLLO DE ALGORITMOS DE LENGUAJE NATURAL EN C# PARA EL CONTROL DE POSICIONAMIENTO DEL BRAZO DE ROBOT MITSUBISHI RV-M1** 5
Camargo López Jesús Amor, Cerón Valle Grissel Angélica, Fuentes Covarrubias Ricardo, Fuentes Covarrubias Andrés Gerardo.

**DESARROLLO DE ALGORITMOS DE CLASIFICACIÓN DE OBJETOS DEL SIMULADOR DE ESLABÓN CINEMÁTICO DE PRENSA POR MEDIO DEL BRAZO DE ROBOT INTELLITEK ESCORBOT** 11
Salas Sergio, Capistrán Tomás, Fuentes Covarrubias Andrés Gerardo, Fuentes Covarrubias Ricardo.

**DESARROLLO DE ALGORITMOS DE VISIÓN ARTIFICIAL EMBEBIDA EN C# PARA LA CLASIFICACIÓN DE OBJETOS POR MÉTODOS MORFOLÓGICOS PARA EL BRAZO DE ROBOT MITSUBISHI RV-M1** 21
Ledesma Natera Edgar Giovanni, Fuentes Covarrubias Andrés Gerardo, Fuentes Covarrubias Ricardo.

**PUESTA EN FUNCIONAMIENTO DEL KIT LV-100 PARA INSTRUMENTACIÓN VIRTUAL USANDO LABVIEW** 34
Angulo Molina Ilda Fabiola, Torres Ruiz Hugo, Fuentes Covarrubias Andrés Gerardo, Fuentes Covarrubias Ricardo.

**DISEÑO E IMPLEMENTACIÓN DE UN CONTROLADOR DE RUIDO ACTIVO MEDIANTE UN ALGORITMO DE LAZO ABIERTO** 42
Elizondo Santos Christian Issachar, de los Santos Oscar Vicente, Nakay Carlos Cedillo, Sierra Peón Mónica Violeta, Fuentes Covarrubias Andrés Gerardo, Evangelista Salazar Martha Elizabeth.

**CLASIFICADOR DE PIEZAS POR TAMAÑO** 47
López Romero José María, Nava Aguirre José Ignacio, Torres Rangel Andrés, Vázquez Reina Jair, Solorio Alejandro, Fuentes Covarrubias Ricardo.

**ROBOT VOCACIONAL** 51
Ruiz Tadeo Ana Claudia, Nava Huitrón Débora Fabiola, Chávez Valdez Ramona Evelia, Saenz Rodríguez María Isabel, González Valladares Esteban.

**REPORTE TÉCNICO: INSTALACIÓN Y USO DE UN CLUSTER DE COMPUTADORAS** 57
Contreras Torres Victoria, Gutiérrez Morfín Roberto Carlos, Martínez Núñez Alberto Isaac, Verduzco Ramírez Jesús Alberto.

**REPORTE TÉCNICO: INSTALACIÓN, CONFIGURACIÓN Y PUESTA A PUNTO DE UNA NUBE PRIVADA CON UNA ESTRUCTURA (IaaS)** 61
Gutiérrez Morfín Roberto Carlos, Verduzco Ramírez Jesús Alberto, Rocha Medrano Pedro Contreras Torres Victoria.

# ÍNDICE

**EL ENTORNO URBANO COMO FACTOR EN EL COMPORTAMIENTO SOCIAL, CASO INFONAVIT - LA ESTANCIA**
Covarrubias Ruesga María Silvia del Rocío , Leyva Palacios Alejandrina, Flores Toscano Oscar Alejandro. **69**

**FACTIBILIDAD ECONÓMICA DE LA ELABORACIÓN DE VIVIENDAS A BASE DE CONTENEDORES METÁLICOS PARA EL ESTADO DE COLIMA**
Moreno Peña Ricardo, Aviña Iglesias Ramón , Guerrero Garcia Isabel , Ruiz Tadeo Ana Claudia. **77**

# APLICACIÓN PARA CALCULAR LA TRAYECTORIA DE LA LUZ SOLAR

[1]Moreno Pe¶a Ricardo, [2]Puente García Simón Pedro , [3]Rebolledo Peregrino Luis Antonio, [4]Verduzco Ramírez Jesús Alberto

*[23]Universidad de Colima, Facultad de Ingeniería Mecánica y Eléctrica, Km. 9 Carr. Colima-Coquimatlán, Colima, México.*

*[3]Instituto Tecnológico de Colima. Av. Tecnológico, Villa de Álvarez, Colima. México.,*

*[1]rmoreno@infonavit.org.mx,[2]simonppg@gmail.com,[3]LARP_03@hotmail.com, [4]averduzco@itcolima.edu.mx*

## Abstract

*In Architecture it has given great importance to the location of the sun for the best use of its light and heat in building construction. For this it is essential to know the placement of the sun throughout the solar cycle, which is used in the construction plains by the architects, however it is slow to calculate all relevant positions manually. To avoid wasting time calculating the above, the decision was made to make a program that displays the entire path of the sun during solar cycle, At the beginning of the project provided a program made in Excel which could see some facts about the Azimut[1] and solar altitude at a certain time and day, it also acquired a solar chart. The objective is to create an application that serves to visualize the entire journey of the Sun.*

## I. Introducción

En la arquitectura se le ha dado una gran importancia a la ubicación del sol para el mejor aprovechamiento de su luz y calor en la construcción de edificios. Para esto es fundamental conocer la colocación del Sol durante todo el ciclo solar, lo cual es usado en los planos de construcción por los arquitectos, no obstante es tardado calcular todos los posicionamientos relevantes de manera manual. Para evitar perder el tiempo calculando lo anterior, se tomo la decisión de hacer un programa que permite visualizar toda la ruta del Sol durante su ciclo solar. Al inicio de el proyecto se aportó un programa hecho en Excel que permitía ver algunos datos acerca del Azimut[1] y altura solar en determinada hora y día, también se adquirió una gráfica solar. EL objetivo es crear una aplicación que sirva para poder visualizar todo el recorrido del Sol.

## II. Material y Métodos

En vista del cumplimiento del objetivo de este sistema se desarrolló una aplicación que permitiera a los usuarios manipularlo de manera sencilla, rápida y eficaz, proporcionándoles una imagen en 2D del recorrido del sol que ellos requirieran para facilitarles su trabajo. Durante el desarrollo del mismo se pensó en una estrategia que posteriormente permitiera ponerlo al alcance de todos los arquitectos estableciendo que el sistema pudiera adaptarse a una pagina Web, para ello se utilizo un lenguaje de programación que facilitara a otros programadores adaptar el código fuente para montarlo en un servidor Web.

El lenguaje de programación fue Java por considerarse un lenguaje de programación y la primera plataforma informática creada por Sun Microsystems en 1995. Es la tecnología subyacente que permite el uso de programas punteros, como herramientas, juegos y aplicaciones de negocios. Java se ejecuta en más de 850 millones de ordenadores personales de todo el mundo y en miles de millones de dispositivos, como dispositivos móviles y aparatos de televisión.

Se acordó que el sistema fuera desarrollado en un lenguaje que posteriormente pudiera ser adaptado para montarlo en una pagina Web por tanto fue en Java el lenguaje elegido para esto, para programar en Java se requiere conocer las clases disponibles en la biblioteca de Java, para crear objetos que son necesarios para una mejor estructuración del programa, estos objetos poseen atributos que podemos utilizar para nuestros propósitos, pudiendo acceder a ellos mediante los métodos de estos mismos.

Conforme al objetivo de plasmar el recorrido de el sol en una IU, se vio la necesidad de usar una clase que permitiera utilizar sus métodos para el trazo de gráficos, la clase java.awt.Canvas me permite crear un objeto que puede dibujar en la pantalla, se necesita sobrescribir esta clase para dibujar de manera personalizada.

Se requería tener una estructura para nuestra gráfica por lo cual primeramente se intento dibujar una serie de círculos que representarían el ángulo de la altura solar, sobre el cual se graficarían las trayectorias de el sol. Se presento el problema de que al dibujar el círculo no tomaba el centro del círculo como base para ser dibujado, como lo podemos observar en la Figura 1.

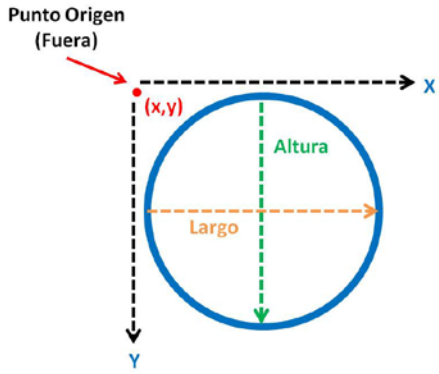



**Figura 1. Círculo fuera de su punto base u origen.**

Esto sucede porque el método drawOval() se le envían como parámetros las coordenadas de un punto cualquiera y el largo y alto de el circulo y a partir de ese punto el circulo se dibujaba. Para solucionarlo se utilizo un método SetCirculo() (Cuadro 1) de la clase Graficas (clase desarrollada para solucionar detalles gráficos) el cual recibe como parámetro un punto y un radio para poder dibujar un círculo como se aprecia en la Figura 2.

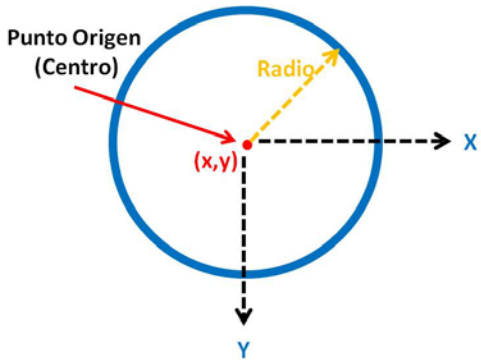



**Figura 2. Círculo dibujado en base a un punto y el radio.**

```
int j=0;
String gra;
//círculos que representa la altura solar
   while(j<=90)
   {
      gra=Integer.toString(90-j);
            //método que obtiene el origen del
                                  circulo y el radio
       obj.SetCirculo(xOrigen, j);
           //dibuja el circulo bien con el origen
                          en el centro del círculo
       g.drawOval(obj.posicion, obj.posicion,
obj.Largo, obj.Largo);

       g.drawString(gra+"º", xOrigen,
obj.posicion);
          j+=10;
      }
```

**Cuadro 1. Dibuja círculo que representa la altura solar.**

Las líneas que representan el ángulo (Azimut) se trazan cada 10 grados partiendo del origen de los círculos antes mencionados hasta la circunferencia del círculo mayor, para trazar una línea en java se utiliza el método drawLine() al cual se le envían 4 parámetros que son las coordenadas de dos puntos (el origen de los círculos y el punto sobre la circunferencia del círculo mayor con respecto del ángulo) el problema es que se desconocen las coordenadas del segundo punto, para ello se creó el método SetPunto() que calcula ese punto, usando el ángulo y la distancia que existe desde el origen

hasta la circunferencia del cículo mayor, en el Cuadro 2, se observa la implementación del método SetPunto().

```
int cont=-180;
//divide el circulo cada 10º
while(cont<=180)
{
    //método que obtiene un angulo
                                y un radio
    obj.SetPunto(cont, radio);

    //dibuja la línea correspondiente
                            al angulo cont
    g.drawLine(xOrigen, yOrigen,
(int)(xOrigen+obj.x1),
(int)(yOrigen+obj.y1));
    cont+=10;
}
```

**Cuadro2. Segmento de codigo que divide el esquema cada 10 grados.**

En el Cuadro 3 se presentan las formulas utilizadas en el método SetPunto() para calcular cada una de las coordenadas X y Y con base en sus ángulos.

//expresión algebraica

$$radian = \frac{ang * \pi}{180}$$

$$m = tan(radian)$$

$$y = \sqrt{\frac{r}{m^2+1}}$$

$$x = \sqrt{r^2 - y^2}$$

```
//exprecion en codigo java
radian=(ang*Math.PI)/180;
pendiente=Math.tan(radian);
y1=Math.sqrt((Math.pow(radio,
2))/(Math.pow(pendiente, 2)+1));
x1=Math.sqrt(Math.pow(radio, 2)-
Math.pow(y1, 2));
```

**Cuadro 3. Formulas aplicadas para la obtención de las coordenadas X y Y de un punto.**

De la misma manera como se dividió el esquema base para nuestra gráfica utilizando el método SetPunto(), este también fue útil para graficar los puntos obtenidos por las formulas que calculan el ángulo de el azimut y la altura solar, formulas que se muestra a continuación en el Cuadro 4.

```
//expreciones utilizadas para obtener los
 angulos correspondientes a la posicion solar

altura=asin(((segundoV)*cos((cont)*0.0174))
+(primerV))/0.0174;

 acimut=asin((tercerV)*sin((cont)*0.0174)/
cos((altura)*0.0174))/0.0174;
```

**Cuadro 4. Formulas que calculan la posición del sol.**

## III. Conclusiones

Con respecto a lo que se desarrollo podemos concluir que la automatización de el calculo de la trayectoria de el sol tiene problemas, como que al plasmarlo en la gráfica creada, presentaba unos problemas en el trazo de la trayectoria cuando las líneas dibujadas rebasaban el eje de las X tal y como se muestra en la Figura 3 donde se observa como la línea rebota al tratar de pasar al siguiente cuadrante en el esquema. Cuando se trató de resolver este problema no se logro un buen resultado de visualización en la gráfica.

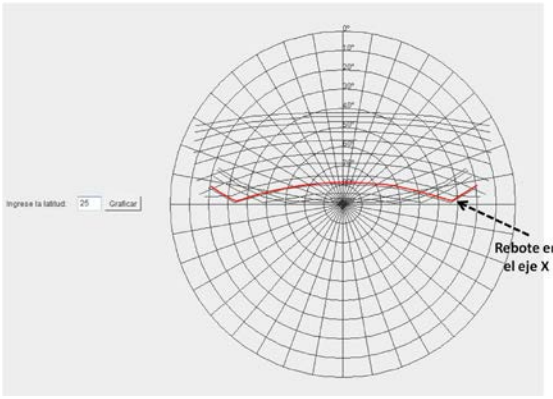



**Figura 3. Rebotó al toparse con el eje x del plano.**

# DESARROLLO DE ALGORITMOS DE LENGUAJE NATURAL EN C# PARA EL CONTROL DE POSICIONAMIENTO DEL BRAZO DE ROBOT MITSUBISHI RV-M1

[1]Camargo López Jesús Amor , [2]Cerón Valle Grissel Angélica , [3] Fuentes Covarrubias Ricardo, [4]Fuentes Covarrubias Andrés Gerardo

*[1 2]Universidad Autónoma de Sinaloa, Facultad de Ingeniería Mochis*
*[3 4]Universidad de Colima, Facultad de Ingeniería Mecánica y Eléctrica, Km. 9 Carr. Colima-Coquimatlán, Colima, México.*
*[1]amor.camargo@yahoo.com, [2]grisselceron@yahoo.com, [3]fuentesr@ucol.mx, [4]fuentesg@ucol.mx*

**Resumen**

*Este proyecto consiste en diseñar e implementar, por ende, el controlador embebido de posicionamiento del brazo de robot Mitsubishi RV-M1 aplicando técnicas de lenguaje natural por medio del Speech Recognition Kit SR-06/SR-07, así como utilizar el entorno de desarrollo Arduino para programar la interfaz entre la tarjeta y el brazo.*

**Palabras clave.** Controlador embebido, Arduino, brazo de Robot.

***Abstract.*** *The project consists to design and implement, therefore, the embedded controller of the robot arm positioning Mitsubishi RV-M1 applying natural language techniques through as well as use the Arduino development environment for programming the interface between the card and the arm.*

**Keywords.** *Embedded controller, Arduino, robot arm.*

## I. Introducción

El lenguaje natural, como disciplina de la Inteligencia Artificial, poco se ha utilizado en aplicaciones de control robótico, más bien ha tenido su advenimiento la visión artificial debido al amplio desarrollo del tema en la industria. Sin embargo, lo anterior representa un área de oportunidad para la aplicación de los sistemas embebidos al desarrollo de pequeños sistemas de control de dispositivos robóticos. Por lo que se pretende, en este proyecto, diseñar e implementar el controlador embebido de posicionamiento del brazo de robot Mitsubishi RV-M1 aplicando técnicas de lenguaje natural por medio del Speech Recognition Kit SR-06/SR-07. Así mismo, diseñar la interfaz entre el brazo y la tarjeta Arduino, utilizando el entorno de desarrollo de Arduino.

### 1.1 Reconocimiento de Voz

El reconocimiento de voz se clasifica en dos categorías, dependiente del hablante y el independiente del hablante.

Los sistemas dependientes del hablante son entrenados por la persona que va a utilizar el sistema. El inconveniente de este enfoque es que el sistema sólo responde con precisión sólo para el individuo que se formó el sistema. Este es el método más común empleado en software para ordenadores personales.

El sistema independiente del hablante, es un sistema preparado para responder a una palabra,

independientemente de quién habla. Por lo tanto, el sistema debe responder a una gran variedad de patrones de lenguaje, las inflexiones y la enunciación de la palabra objetivo. El número de palabras de comando es generalmente más bajos que el de los sistemas dependientes del hablante, sin embargo todavía se puede mantener dentro de los límites de procesamiento [1].

## Reconocimiento de estilos de voz

Los sistemas de reconocimiento de voz tienen otra restricción sobre la forma de hablar que puede reconocer. Son tres estilos de expresión: aislados, conectados y continua.

El estilo aislado de los sistemas de reconocimiento de voz sólo puede manejar palabras que se pronuncian por separado. Se trata de los sistemas de reconocimiento del habla más comunes hoy en día. El usuario debe hacer una pausa entre cada palabra o un comando hablado. El circuito de reconocimiento de voz está configurado para identificar palabras aisladas de longitudes de 0,96 segundos.

El estilo conectado es un punto a medio camino entre la palabra aislada y de reconocimiento de voz continua. Permite a los usuarios hablar varias palabras. El HM2007 se puede configurar para identificar las palabras o frases en 1,92 segundos de duración. Esto reduce el número de reconocimiento de palabras de vocabulario para 20.

El estilo continuo es la expresión natural de la conversación que estamos acostumbrados en la vida cotidiana. Es extremadamente difícil para un reconocedor de cambio, ya que las palabras tienden a fusionarse. Por ejemplo, "Hola, ¿cómo estás?" suena como "Hola, howyadoin".

### Speech Recognition Circuit

El Speech Recognition Kit es un circuito de reconocimiento de voz programable, en el sentido que se puede tratar con las palabras que se desea que el circuito reconozca.

El corazón del circuito es el circuito integrado de reconocimiento de voz HM2007. Este IC. Puede reconocer 40 palabras, cada una con un espacio de .96 segundos o 20 palabras, cada una con un largo de 1.92 segundos.

El chip tiene dos modos de operación; el modo manual y el modo CPU. El modo CPU está diseñado para permitir al chip trabajar en un equipo host. Este es un enfoque atractivo para el reconocimiento de voz para ordenadores debido a que el chip de reconocimiento de voz funciona como un co-procesador de la CPU principal. El trabajo de escuchar y el reconocimiento no ocupan ningún tiempo de CPU de la computadora. Cuando el HM2007 reconoce un comando que puede ser señal de una interrupción a la CPU del host y luego transmitir el código de comando. El chip HM2007 puede conectarse en cascada para proporcionar una biblioteca de reconocimiento de palabras mayores.

El SR-07 del circuito que se construyó funciona en modo manual. El modo manual puede permitir construir una junta independiente de reconocimiento de voz que no requiere un ordenador y puede ser integrado en otros dispositivos que utilizan el control del habla. [1]

### Arduino

Arduino es una plataforma de prototipos electrónica de código abierto (open-source) basada en software y hardware flexibles y fáciles de usar. Puede sentir el entorno mediante la recepción de entrada desde una variedad de sensores y puede afectar a su alrededor mediante el control de luces, motores y otros artefactos. El microcontrolador de la placa se programa usando el "Arduino Programming Language" (basado en Wiring) y el "Arduino Development Environment" (basado en Processing) (Fig. 1). [2].

Esta placa cuenta con las siguientes características:

Tiene 16 pines digitales de Entrada/salida
Puerto serial USB, para conectarlo a una PC
Transistores
Botón de reset
Memoria Flash
Transistor Regulador de voltaje (7805CT).
Microcontrolador
1 cristal
Resistencias de voltaje
Fotodiodos

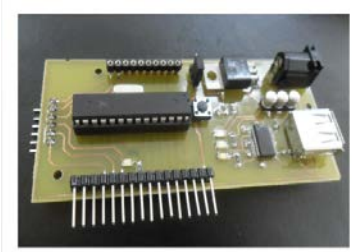


**Fig.1 Placa Arduino**

Esta tarjeta de Arduino tiene un número de infraestructuras para comunicarse con un ordenador, otro Arduino u otro microcontrolador. Una de las ventajas de esta tarjeta (además de ser única), es que provee una comunicación serie UART TTL (5 V), la cual está disponibles en los pines digitales 0 (Rx) y 1 (Tx). El software Arduino incluye un monitor serie que permite a datos de texto simple ser enviados a y desde la placa Arduino. [2]

## Mitsubishi RV-M1

El Brazo de Robot Mitsubishi RV-M1 (Fig. 2) es un brazo industrial, el cual cuenta con 5 grados de libertad, posee una capacidad de carga de 1.2 Kg, la longitud del brazo es de 400 mm, tiene un peso de 19 Kg y una velocidad de 1000 mm/s. Puede realizar prácticamente cualquier tarea, desde recolectar y mover componentes de secuencias de manipulación compleja que puede ser programado y controlado con una consola de programación, así como un PC, un PLC u otro sistema de control de procesos.

La unidad motriz se comunica con el Mitsubishi RV-M1 a través de dos cables. Todos los puertos de interfaz y las conexiones para equipos periféricos y los datos de procesos y de E/S se encuentran en el panel posterior. [3].



**Fig. 2 Brazo de Robot Mitsubishi RV-M1**

## II. Metodología.

Se comenzó con la construcción del Speech Recognition Kit SR-06/SR-07. Primeramente se armó el Main Circuit Board (SR-07) (Fig. 3). Se comenzó la construcción montando y soldando los 3 circuitos integrados, el HM2007 (que es el corazón del circuito), la RAM y el circuito 74LS373. Se prosiguió montando y soldando los resistores, los diodos, el cristal XTA (de 3.53 MHz), el LED rojo y los capacitores, se montaron y soldaron el regulador 7805 de voltaje y un Switch On-Off, se montó y soldó la toma del micrófono y el botón de

soporte de la batería, así como la betería de 9 voltios. Se finalizó con el Main Circuit Board montando y soldando los 10 pines en la esquina superior izquierda del tablero, así como los 7 pines en la esquina inferior izquierda del tablero y los dos pines en la locación WD.

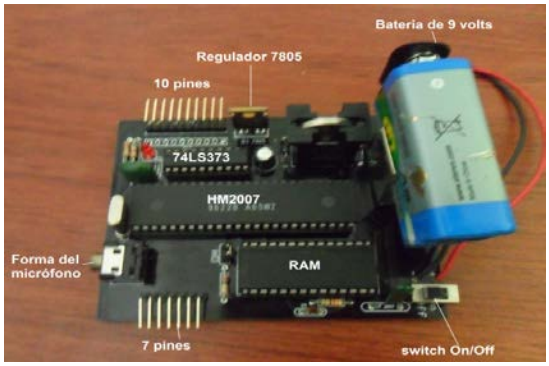



**Fig.3 Main Circuit Board**

Posteriormente, se continuó construyendo el Display Board (Fig.3) comenzando por montar y soldar los 16 resistores (de 220 ohm), en segundo plano se soldando los 14 pines de enchufe para los circuitos integrados del LED del Display, así como con la instalación de los LED del Display en los enchufes. Se continuó montando y soldando los 16 pines de enchufe para el circuito integrado 4511, así como instalando el circuito mismo. Se finalizó soldando un puente en 3 puntos de soldadura, comenzando en el punto T y terminando en el punto C, así mismo se montaron y soldaron los 10 pines en la parte inferior del tablero. (Fig. 4)

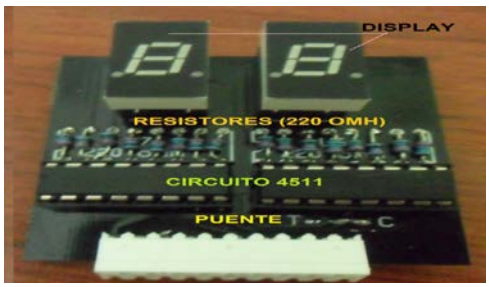



**Fig.4 Display Board**

Por último, se continuó con la construcción del Keypad. Comenzando por montar y soldar 12 switches de contacto, cada uno montado en una determinada posición, así como también conectando los 7 pines en la parte superior del tablero. (Fig. 5).

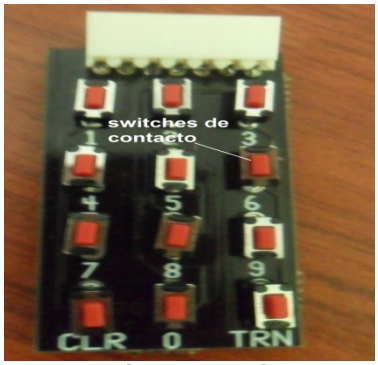



**Fig. 5 Keypad**

Se finalizó la construcción del Speech Recognition kit montando el Display Board a los 10 pines ubicados en la parte superior izquierda del Main Circuit Board, al igual que montando el Keypad a los 7 pines ubicado en la parte inferior del Main Circuit board.

## Pruebas con el Speech Recognition Kit SR-07

Se probó el Speech Recognition Kit comenzando primeramente en poner en On el switch, mostrándose en los Displays "00" y el LED rojo encendido. Posteriormente se presionó el número (posición), de la palabra que se deseaba guardar, en el Keypad (se puede utilizar cualquier número entre 1 y 40). Por ejemplo, se comienzó presionando el número 1 (el Display muestró "01" y el LED rojo se apagó) seguido de la tecla TRN (el LED se enciendó) para que el circuito esté listo para tratar con una palabra. Se prosiguió diciendo un comando de robot en el micrófono, el circuito aceptó y entendió la señal de voz, por lo que se apagó y encendió el LED rojo. La palabra (comando) quedó identificada como el comando "01". Se realizaron los mismos pasos para grabar los siguientes comandos en las posiciones determinadas:

Arriba -> 01
Abajo -> 02
Derecha ->03
Izquierda ->04
Detente -> 05
Cierra -> 06
Levanta -> 07

Para probar que dichos comandos quedaron grabados, éstos se repitieron en el micrófono y el número de palabra donde fueron grabados se mostró en el Display. Por ejemplo, si el comando fue "Detente" (que fue grabada en la posición 05) y al repetirlo en el micrófono, en el Display se mostró el número 05.

## Pruebas con Arduino

Se realizaron las pruebas con la tarjeta de Arduino (fig. 4). Lo primero que se realizó fue la configuración entre la placa Arduino y la PC. Para ello se abrió el menú "tools", la opción "serial port" (en la cual se seleccionó el puerto serial en el que se encontraba conectada dicha placa).

Para comprobar que todo estaba bien, así como para familiarizarse con la interfaz de desarrollo se comenzó por inicializar los programas precargados en la tarjeta. Se accedió al menú File- Sketchbook – Examples – Digital - Blink. Este ejemplo (Blink), de lo único que se encarga de hacer es de parpadear un LED que está conectado en el pin número 13 de la tarjeta (Fig. 5). De la misma manera se hicieron las pruebas con todos los programas pregrabados en la placa Arduino, cada uno con una función diferente.




**Fig. 5 Entorno Arduino**

**Mitsubishi RV-M1**

Se grabaron posiciones previas en el Brazo de Robot Mitsubishi RV-M1, con el fin de establecer una rutina al momento de comandar al brazo por comando de voz. Cada rutina estará enlazada a un comando de voz.

Una de las rutinas grabadas fue la siguiente:

1.- Nido……..….[1]
2.- Descanso…….....[2]
3.- Izquierada (aprox)…….....[101]
4.- Abajo (baja por la pieza)....….....[102]
5.- Cerrar (cerrar griper)……....[103]
6.-Arriba (Volver a aprox)…….….[107]
7.-Derecha (aprox 2)…….....[104]
8.-Baja (Baja a dejar pieza)……....[106]
9.-Abrir (abrir Griper)……..….[108]
10.-Sube (Vuelve a aprox. 2 con griper cerrado) [109] (Fig. 6).



**Fig.6 Mitsubishi RV-M1**

## III. Resultados

Los primeros resultados que se obtuvieron fueron con el Speech Recognition Kit SR-06/SR-07 resultaron exitosos. Se logró probar que efectivamente, por medio del uso de un sistema embebido, se pueden aplicar técnicas de lenguaje natural. El kit logró reconocer todos los comandos que fueron grabados para su posterior uso, mostrando en el Display de la tarjeta el número de posición en el cual se grabó dicho comando. (Fig. 7).

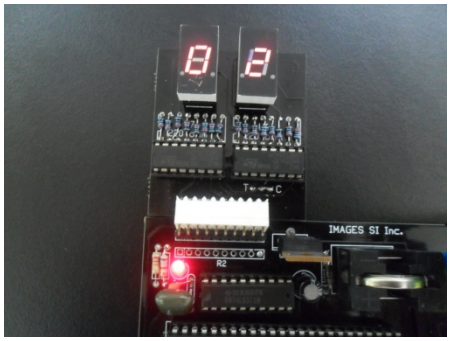



**Fig. 7 Resultados impresos en el Display**

Los siguientes resultados fueron dados por la Tarjeta de Arduino, probando que su funcionalidad era la que se deseaba obtener, gracias a que demostró un excelente desempeño cuando se realizaron las pruebas con las aplicaciones hechas en su entorno.

Por otra parte, no se logró terminar el proyecto como se deseaba o esperaba; pero se dio un gran paso en la investigación del tema.

## IV. Discusión

No cabe duda, que la aplicación del lenguaje natural en los sistemas de control robótico es una gran aportación a la robótica. Por lo tanto, cuando se llega a aplicar los sistemas embebidos a los sistemas de control robótico para el posicionamiento de un brazo robótico resulta ser, aún, una aportación mayor.

Este proyecto se trató de enfocar en ello. Sin embargo, debido a que es un tema nuevo en el campo de la robótica y a que no existe gran cantidad de información sobre el tema, no se pudo concluir con dicho proyecto. Sin contar con que el diseño de la tarjeta de Arduino es auténtico y no existe una similar en el mercado.

Por ende, se pretende que en un futuro este proyecto se retome y llegue a ser concluido, con ayuda del avance que se dio se espera que se llegue al objetivo.

## V. Agradecimientos

Se agradece a los M.C. Ricardo y Gerardo Fuentes Covarrubias por su cordial apoyo y por compartir su proyecto con nosotros. Sin contar la gran experiencia que logramos obtener con este proyecto, así como adentrarnos en la investigación científica.

Por otro lado también le agradecemos a la universidad autónoma de Sinaloa por darnos el apoyo para poder lograr esta experiencia donde perdimos poco, aprendimos mucho y ganamos demasiado.

A nuestros padres Francisco Cerón, Rosalba Valle, Verónica López y Jesús Camargo por habernos apoyado a hacer las grandes personas que hoy en día somos.

## VI. Bibliografía

[1] SR-06/SR-07 Speech Recognition Kit, Construction Manual & User Guide.
[2] Guía de Usuario de Arduino, Rafael Enríquez Herrador I.T.I. Sistemas Universidad de Córdoba i52enher@uco.es
[3] http://www.mitsubishirobot.com/r1.html

## VII. Autores

Camargo López Jesús Amor
Originaria de Los Mochis, Sinaloa. Actualmente se encuentra estudiando la Licenciatura en Ingeniería De Software en la Facultad de Ingeniería Mochis, en la Universidad Autónoma de Sinaloa.



Cerón Valle Grissel Angélica
Originaria de Los Mochis, Sinaloa. Actualmente se encuentra estudiando la Licenciatura en Ingeniería De Software en la Facultad de Ingeniería Mochis, en la Universidad Autónoma



# DESARROLLO DE ALGORITMOS DE CLASIFICACIÓN DE OBJETOS DEL SIMULADOR DE ESLABÓN CINEMÁTICO DE PRENSA POR MEDIO DEL BRAZO DE ROBOT INTELLITEK ESCORBOT

[1]Sergio Salas, [2]Capistrán Tomás, [3]Fuentes Covarrubias Andrés Gerardo, [4] Fuentes Covarrubias Ricardo

*[1 2] Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Culhuacán, Avenida Santa Ana #1000 Col. San Francisco Culhuacan, Deleg. Coyoacan C.P. 04430, México D.F.*

*[3 4]Universidad de Colima, Facultad de Ingeniería Mecánica y Eléctrica, Km. 9 Carr. Colima-Coquimatlán, Colima, México.*

*[1]bicho_lindo@hotmail.com, [2]tomas.capistran@hotmail.com, [3]fuentesg@ucol.mx, [4]fuentesr@ucol.mx*

## Resumen

*El presente material se desarrollo los algoritmos del Brazo de Robot Intellitek Scorbot para la clasificación de una pieza metálica que pasa por tres etapas. En el cual consiste en cuando el sensor detecte la pieza dentro del contenedor localizado en la etapa de estampación, el mandara una señal capaz de iniciar el programa, como consecuencia el Robot Scorbot traslade la pieza a la banda transportadora, enseguida el brazo saldrá del área de trabajo de la banda, esta trasladara la pieza a otro punto de la banda, para después el Robot valla por la pieza, la tome y la lleve al área de almacenamiento. Por último el brazo Scorbot tendrá que irse a la posición de espera, hasta que el sensor le indique que haga su rutina.*

## 1. Introducción

Durante la evolución de las máquinas industriales, el hombre se ha sentido fascinado por las maquinarias y dispositivos capaces de realizar actividades y movimientos de los seres vivos. De esta manera, hoy se tienen en muchas de las industrias del mundo, robots diseñados con particularidades humanas, como es el caso de los brazos robots.

Al involucrarse en la utilización de robots, uno de los aspectos importantes que hay que tomar en cuenta es "las leyes de la robótica" [1], que se expresa de la siguiente manera:

1. Un robot no puede perjudicar a un ser humano, ni con su inacción permitir que un ser humano sufra daño.

2. Un robot ha de obedecer las órdenes recibidas de un ser humano, excepto si tales órdenes entran en conflicto con la primera ley.

3. Un robot debe proteger su propia existencia mientras tal protección no entre en conflicto con la primera y segunda ley.

Por otro lado, si se desea definir el concepto de robot industrial, la Federación Internacional de Robótica (FIR) lo determina de la siguiente manera: Por robot industrial de manipulación se entiende a una máquina de manipulación automática, reprogramable y

multifuncional con tres o más ejes que pueden posicionar y orientar materias, piezas, herramientas o dispositivos especiales para la ejecución de trabajos diversos en las diferentes etapas de la producción industrial, ya sea en una posición fija o de movimiento.

En este sentido se cuenta con un simulador de prensa de estampación neumático que requiere la integración de un brazo de robot Intellitek Scorbot para integrar una cadena cinemática con ambos eslabones. En la actualidad es una aplicación trivial la integración de brazos de robot a las aplicaciones de clasificación de objetos, esto debido a su naturaleza repetitiva. En esta aplicación se deberá diseñar e implementar los algoritmos de posicionamiento

y control para realizar las tareas de pick&place de objetos utilizando el simulador de prensa de estampación.

## 2. Características del Brazo Robot Scorbot-ER IX

El Scorbot-ER IX es robot articulado vertical, similar a un brazo, con 6 articulaciones (ejes) para su movimiento, efectuado por motores de cd presentado en la figura y tabla 1.



Figura 1. Ejes y movimientos

El brazo robot tiene un área de operación definida por sus dimensiones físicas como muestra la figura 2.

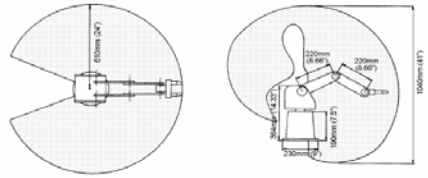


Figura 2. Área de operación.

## 3. Controlador B

El controlador se encarga de todos los procesos que lleva acabo el brazo robótico a continuación se dará una breve explicación de los principales interruptores y puertos que contiene.

### 3.1 Interruptor de Encendido

Localizado en la parte trasera del controlador

### 3.3 Interruptor Motores

Este interruptor conecta y desconecta el voltaje DC a los motores, El interruptor lleva un led que se ilumina cuando esta pulsado y hay tensión en los motores

### 3.4 Emergencia y Led

Con este botón se desactiva la potencia de los motores, se activa el control off, se ilumina el led de emergencia, aparece en el monitor un mensaje de emergencia, todos los programas que se estén ejecutando son abortados, las entradas y salidas son congeladas.

### 3.4 Entradas

El controlador B dispone de 16 entradas, divididas en dos bloques aisladas ópticamente (bloque C y D). Cada bloque es flotante y puede referenciarse a un elemento externo diferente o a la tierra del controlador. Además, cada entrada puede actuar como fuente o sumidero de corriente.

Las entradas se activan por medio de salidas de un elemento externo, que están conectadas a las correspondientes entradas del controlador.

Si la entrada está configurada como Fuente:

Un voltaje externo de 0 a 1.5 activara la entrada a 1.

Un voltaje de 3VDC desactivara la entrada (0).

Es en caso contrario cuando están configuradas como SUMIDERO (SINK)[2].

## 3. Salidas

Tabla 1. Ejes y movimientos

| No. del eje | Nombre articulación | Movimiento |
|---|---|---|
| 1 | Base | Rota el cuerpo |
| 2 | Hombro | Levanta y baja el ante brazo |
| 3 | Codo | Levanta y baja el brazo |
| 4 | Elevación | Levanta y baja la pinza |
| 5 | Giro | Gira la pinza |
| 6 | Pinza | Abre y cierra la pinza |

El controlador B tiene 16 salidas. Cuatro son salidas de relé con un contacto normalmente abierto y normalmente cerrado. Las otras salidas son del tipo colector abierto y están divididos en 2 bloques ópticamente asilados.

Cada bloque es eléctricamente flotante y puede ser referenciado a un elemento externo individual o al común del propio controlador además cada salida puede trabajar como fuente o como sumidero

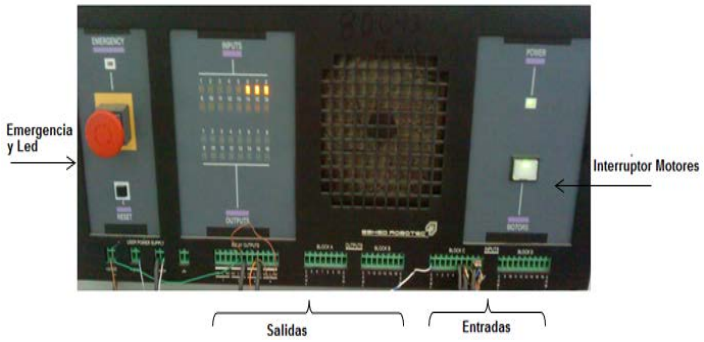


Figura 3.Controlador B

## 4. Botonera de Enseñanza

Botonera de enseñanza, que permite el control de las articulaciones, la definición de posiciones por medio del método de enseñanza y la ejecución de programas que se encuentren en el controlador.



Figura 4. Botonera de Enseñanza

### 3.1 Control de Teach y Auto

Selecciona si el control será con la botonera o por medio de la computadora.

### 3.2 Botón de Tipo de Movimiento.

Selecciona si el movimiento será en:

Coordenadas cartesianas (XYZ), especifica la posición de la punta de la pinza como la distancia en milímetros desde el punto central de la base del robot en los tres ejes lineales XYZ.

Coordenadas de robot (JOINTS). Especifican cada posición de cada eje con pulsos de encoger. Cuando el robot se mueve los encoders ópticos generan una serie de impulsos. El número de pulsos generados es proporcional a la distancia realizada. Cada controlador cuenta los impulsos realizados y determina donde se encuentra cada eje. La posición de cualquier periférico conectado al robot siempre se especifica en pasos de encoder.

### 3.3 Run

Al presionar este botos seguido del número de programa requerido y al presionar enter, correrá dicho programa.

### 3.4 Open/Close

Abre y cierra la pinza

### 3.5 Grabar Posición

Para grabar una posición primeramente se utilizan las teclas de movimiento para llevar al robot a la posición deseada, de ahí se pulsa la tecla record position seguido de un número y al final enter

Aparecerá en el display de la caja de enseñanza: Here #

De esta manera se graba la posición absoluta.

### 3.6 Move

Mueve el brazo a la posición deseada.

### 3.7 Banda Transportadora

La banda transportadora es un accesorio que funciona por medio de un motor de 24 Vcd para desplazar objetos sobre ella. El movimiento lo puede realizar en ambas direcciones. Cuenta con un sensor óptico para indicar que un objeto ha llegado a la posición deseada.

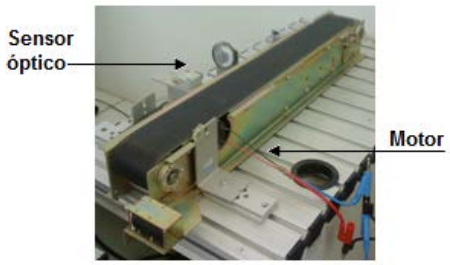


Figura 5. Banda transportadora

## 4. Modelo Cinemático del Robot

La cinemática de un brazo robot estudia la geometría del movimiento del mismo con respecto a un sistema de referencia fijo sin tener en cuenta las fuerzas que los originan. Relaciona la posición y orientación del extremo final del robot con permitiendo la descripción analítica del movimiento espacial del robot como una función del tiempo, y en particular por las relaciones entre la posición y la orientación del extremo final del robot con los valores que toman sus coordenadas articulares [1] [4][5]. En el estudio de la cinemática se plantean dos problemas:

Determinación de la posición y orientación del elemento terminal a partir de las coordenadas articulares del robot (Cinemática directa)

Determinación de las coordenadas articulares a partir de la posición y orientación del elemento terminal (Cinemática inversa)

Existen distintos métodos para resolver estos problemas. En el que se ha utilizado para el desarrollo de la aplicación se basa en la representación de Denavit-Hartenberg (D-H), que sitúa de forma sistemática un sistema de coordenadas en cada articulación del robot, de forma que se pueda relacionar el sistema Si de la articulación i con el sistema de la articulación anterior, Si-1, utilizando una matriz homogénea generada por el producto de cuatro transformaciones básicas: rotación alrededor del eje zi-1 un determinado ángulo

Θi, traslación a lo largo del eje zi-1 una distancia di, traslación a lo largo del xi una distancia ai y rotación alrededor del eje xi un ángulo αi. La posición del elemento terminal vendrá determinada por el producto de matrices homogéneas de transformación que dependerán del valor de cada una de las coordenadas articulares del robot, así como de sus características físicas (longitud, tipo de articulaciones, etc.).

La Figura 6 muestra la situación de los sistemas de coordenadas siguiendo la representación de D-H para el robot manipulador Scorbot ER–IX y en la Tabla2 se muestran los parámetros de D-H de cada articulación del robot. Sustituyendo estos parámetros se obtienen 5 matrices homogéneas que permiten el cálculo de la posición de la pinza del robot a partir delos ángulos de cada una de sus articulaciones.

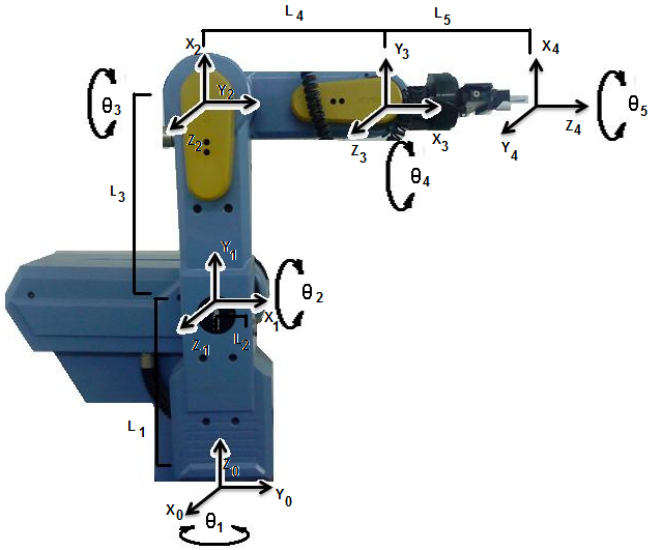


Figura 6. Representación D-H del Scorbot.

Tabla2. Parámetros D-H de Scorbot ER-IX

| Articulación | $\Theta i$ | $d_i$ | $a_i$ | $\alpha_i$ |
|---|---|---|---|---|
| Articulación1 | $\Theta 1$ | L1 | L2 | 90 |
| Articulación2 | $\Theta 2$ | 0 | L3 | 0 |
| Articulación3 | $\Theta 3$ | 0 | L4 | 0 |
| Articulación4 | $\Theta 4+90°$ | 0 | 0 | 90 |

Para el estudio de la cinemática inversa se utiliza la técnica denominada desacoplo cinemático [4], que consiste en dividir el problema en dos más simples:

- Problema cinemático inverso de la posición, en el que obtienen los valores de las primeras tres articulaciones que posicionan al robot
- Problema cinemático inverso de orientación, en el que se obtienen los valores de las dos últimas articulaciones que orientan al extremo del robot.

## 5. Generación de Trayectorias

El generador de trayectorias se ocupa de las posiciones que debe seguir cada articulación a lo largo del tiempo para lograr los objetivos fijados por el usuario (punto de destino, tipo de trayectoria, velocidad, etc.). El generador implementado en la aplicación que se describe en este artículo incorpora dos tipos de trayectorias:

- Trayectorias punto a punto, en la que cada articulación evoluciona desde un punto inicial a un punto final sin realizar consideración alguna sobre el estado o evolución de las demás articulaciones [1].
- Trayectoria continúa definida por el usuario que debe seguir el extremo del robot desde el punto inicial al final, por ejemplo, lineal.

Los algoritmos de generación de trayectorias aproximan el camino a seguir mediante funciones polinomiales. En el caso de la aplicación desarrollada se han utilizado interpoladores cúbicos, que aseguran que la trayectoria que une los puntos por los que tiene que pasar cada articulación presente continuidad en velocidad [5]. Por tanto, la trayectoria de la articulación i estará definida por el polinomio:

$$q_i = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \qquad (1)$$

De este modo, al tener cuatro parámetros disponibles se podrán imponer cuatro condiciones de contorno:

Condiciones iniciales de posición, qi0, y velocidad, qi0'

$$q_{io}(t_f) = q_{io} \qquad q_{io}(t_f)' = q_{io}' \qquad (2)$$

Condiciones finales de posición, qif, y velocidad, qif'

$$q_{if}(t_f) = q_{if} \qquad q_{if}(t_f)' = q_{if}' \qquad (3)$$

Para calcular la velocidad deseada de una articulación, sólo hay que derivar la ecuación de la trayectoria articular:

$$q_1' = a_1 t + a_2 t^2 + a_3 t^3$$

(4)

Combinando las ecuaciones y las condiciones de contorno se obtienen las cuatro ecuaciones siguientes

$$q_{io} = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

$$q_{io}' = a_1 + 2a_2 t + 3a_3 t^2$$

$$q_{if} = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

(5)

$$q_{if}' = a_1 + 2a_2 t + 3a_3 t^2$$

Donde los parámetros a0, a1, a2 y a3 definirán la función que describe la trayectoria que debe seguir la articulación a lo largo del tiempo.

Las trayectorias continuas que se han implementado en esta aplicación son la trayectoria lineal y la circular. Para implementar la trayectoria lineal, conocido el punto inicial y el final, se calcula la ecuación de la recta que los une. A continuación se obtiene un vector de posiciones muestreando la ecuación de la recta obtenida. Para conocer las posiciones articulares en cada instante, sólo hay que aplicar las ecuaciones obtenidas en el estudio cinemático inverso. La trayectoria que se debe seguir entre los puntos muestreados será un polinomio de tercer orden. El resultado final será una trayectoria cuasi-lineal. El mismo procedimiento se utiliza para la trayectoria circular, sólo que la curva a calcular necesita un punto intermedio entre el inicial y el final.

## 6. Clasificación de Objetos

Una pieza metálica rectangular tendrá que pasar por una clasificación el cual cuenta con tres etapas: estampación, transportación y almacenaje. (Vease en la figura 7).

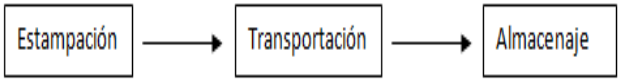



Diagrama de flujo de la cadena cinemática de la pieza metálica

### 6.1 Etapa de Estampación

Esta etapa el simulador de prensa de estampación neumático tomara la pieza metálica y la colocara en un contenedor de tal manera que el brazo de robot Intellitek Scorbot pueda tomarla y colocarla en el inicio de la etapa de transportación.

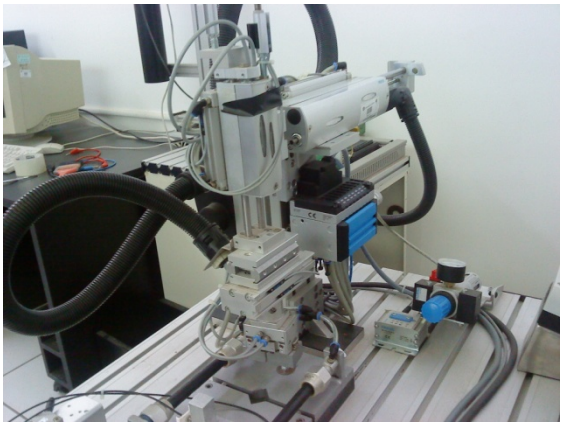


Figura 7. Simulador de eslabón cinemático de prensa

Para lograr lo anterior se tiene que obtener primero la tabla de entradas (Tabla3) y de salidas (Tabla4) Así se podrá desarrollar el programa el cual se cargara en el PLC para su control.

Tabla 3. Entradas del PLC

| Puerto | Entrada | Comentarios |
|---|---|---|
| I0.3 | Botón rojo | |
| I0.2 | Botón amarillo | |
| I0.1 | Botón verde | |
| I0.4<br>I0,5 | Roll sensor | Pick<br>Place |
| I2.2<br>I2.1 | Gripper sensor | Si =1 close<br>Si=1 close |
| I0.7 | X int sensor | Si=0 inactivo<br>Si=1 activo |
| I0.6 | Y int sensor | Si =0 inactivo<br>Si =1 activo |
| I1.3<br>I1.2 | X sensor | Inicio Fin Medio<br>Si =1 Si=0 Si=0<br>Si=0 Si=1 Si=0 |
| I1.0<br>I1.1 | Y sensor | Inicio Fin Medio<br>Si =1 Si=0 Si=0<br>Si=0 Si=1 Si=0 |

Tabla 4. Salidas del PLC

| Puerto | Entrada | Comentarios |
|---|---|---|
| O0.1 | Prensa | |
| O0.2 | Roll | Si =1 place<br>Si =0 pick |
| O0.3 | Xint | |
| O0.4 | Yint | |
| O0.5 | Gripper | Si =1 open<br>Si =0 close |
| O0.6 | X | Si =1 inicio<br>Si =0 fin |
| O0.7 | Y | Si =1 inicio<br>Si =0 fin |

Para realizar el programa de control se debe se seguir una estructura de programación para el listado de instrucciones:

(Paso)

STEP (Etiqueta)

(Frase)

IF (Parte condicional)

THEN (Parte ejecutiva)

OTHRW (Ejecución opcional)

(Pasos) ...

(Pasos) ...

STEP

Tabla 5. Comandos para la estación de estampación

| Comando | Función |
|---|---|
| NOP | No hacer nada, sin condiciones. |
| OR | Realiza la operación lógica or (suma) |
| OTHRW | Aplica cuando la condición es falsa. |
| RESET | Para cambiar a estado lógico "0". |
| ROL | Gira a la izquierda los bits de un acumulador. |
| ROR | Gira a la derecha los bits de un acumulador. |
| SET | Para cambiar a estado lógico "1". |
| THEN | Indica el inicio de la parte ejecutiva. |
| WITH | Para indicar parámetros de funciones. |
| LOAD | Carga un valor en el acumulador |

### 6.2 Etapa de Transportación

En la etapa de transportación, una vez que el brazo Scorbot deje la pieza metálica, y se posicione en un estado de descanso (el cual no permite alguna alteración en esta etapa), la banda transportadora se activa un determinado tiempo para posicionar la pieza a una posición para que el brazo Scorbot pueda agarrarla y enviarla a la etapa de almacenaje.

Para que el brazo Scorbot sepa cuando comienza su rutina, es necesario de la colocación de un sensor láser, de tal manera que detecte la pieza metálica cuando sea colocada en el pequeño contenedor de la etapa de estampación.

## 7. Etapa de Almacenaje

Esta es la última etapa de la clasificación de objetos, la cual es la encargada de tener un lugar específico para guardar la pieza metálica.

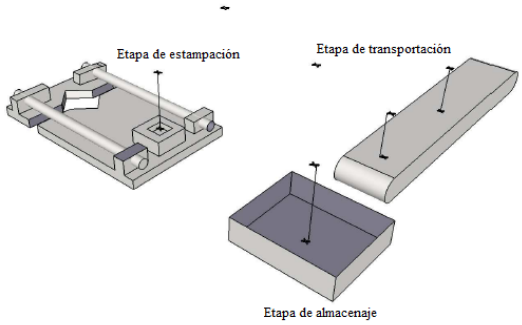



Figura 8 Cadena cinematica

## 8. Manipulación del brazo de robot Intellitek Scorbot

La comunicación entre la aplicación y el controlador del robot se realizan mediante una conexión RS-232.

El controlador trabaja internamente con el lenguaje de programación ACL , por lo que la aplicación debe enviar la orden al puerto serie con la sintaxis ACL. El controlador la interpreta y la ejecuta. Si la ejecución se ha realizado correctamente, envía a la aplicación la cadena de caracteres "Done.". En caso contrario, devuelve una cadena con la descripción del error que ha ocurrido.

Para trasladar la pieza metálica entre las etapas de la clasificación de objetos se apoya del brazo de robot

Intellitek Scorbot., se guardaron 10 posiciones para manipular el traslado y la espera de la pieza metálica. Estas posiciones se manipularon y una vez estando en la posición se guardaron con la botonería de enseñanza, se utilizo ene el modo XYZ y JOINTS. En el modo JOINTS, nos permitía alcanzar las posiciones alejadas y adecuadas para la toma de la pieza. En el modo XYZ, nos permitía desplazarnos sobre el eje Z para tener las aproximaciones para tomar o dejar la pieza metálica.

En la figura 9 se muestran tres posiciones para la posición 73 es donde el brazo Scorbot espera hasta que una señal que indique que hay pieza por tomarla. La posición 72 es la aproximación para la tomar la pieza en la etapa de estampación y la posición 71 nos posiciona en el lugar adecuado para cerrar la pinza y tomar la pieza.

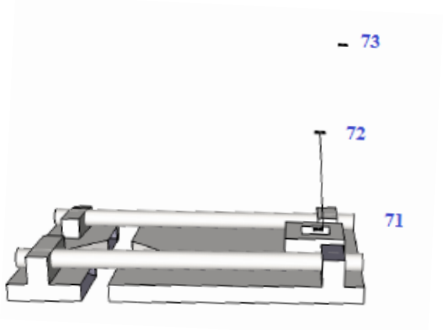



Figura 9 Posiciones en la etapa de estampación

En la figura 10 se muestra 5 nuevas posiciones y la posicion 73. La posicion 74 permite tener una posicion intermedia para pasar de la posicion 73 y 74, ademas de unir la posicion 80 y 73, la posicion 76 es la aproximación para dejar la pieza en la banda transportadora, la posicion 75 es el lugar adecado para dejar la pieza en la banda, la posicion 77 es el punto donde la pieza se quedara despues que sea trasladada por banda transportadora y la posicion 78 es la apriximación de la posicion 77.

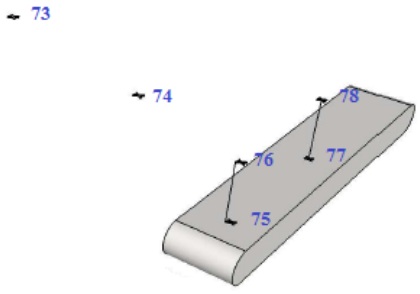



Figura 10 Posiciones en la etapa de transportación

En la figura 11 se presentan las dos últimas posiciones, estas permiten llegar al punto de almacén de la pieza metálica. La posición 79 es el punto para almacenar la pieza y la posición 78 es la aproximación de este punto.




Figura 11 Posiciones en la etapa de almacenaje.

## 9. Comandos

Para realizar el programa para controlar el brazo Scorbot se necesita saber cuáles son sus comandos que acepta y que función realiza.

Tabla 6. Comandos para programar el brazo Scorbot

| Comando | Función |
|---|---|
| LABEL <etiqueta n> | Marca el comienzo de una subrutina de programa que se ejecuta al leer el comando GOTO. |
| IF IN [n]= 1/0 | Comprueba el estado n de entrada, '1' prendido y '0' apagado, si es verdadera la entrada a '1'/ '0' ejecuta las sentencias del que contiene, si es falso sigue adelante. |
| MOVE <pos> {tiempo} | Desplaza al robot a la posición dentro el tiempo especificado (sus unidades son centésimas de segundo) |
| DELAY <var> | Demora la ejecución de un programa por número <var> de tics cada tics es igual a 10 ms |
| OPEN | Abre a pinza hasta el final de su movimiento |
| CLOSE | Cierra la pinza hasta el final de su movimiento |
| SET OUT [n]=1/0 | Fija el estado n de salida especificada con '1' prendido, o '0' apagado |
| ENDIF | Indica el fin del condicional IF |
| GOTO <n> | Ejecuta un salto incondicional hasta el LABEL <n> |
| END | Indica el fin del programa |

## 10. Resultados

Teniendo las posiciones adecuadas pregrabadas en la memoria del Brazo Scorbot, con el siguiente programa se unen las posiciones para que valla y tome la pieza y la deje, pero también necesita un tiempo para poder ejecutar adecuadamente el programa:

```
LABEL 10
IF IN [8] = 1
DELAY 600
MOVE  73  300
MOVE  72  300
MOVE  71  300
DELAY  900
CLOSE

DELAY  50
MOVE  72  300
MOVE  73  300
MOVE  74  200
MOVE  76  150
MOVE  75  200


DELAY  1150
OPEN
DELAY  50
MOVE  76  200
MOVE  74  150
DELAY  350
SET  OUT [3] =1
DELAY 500
SET  OUT [3]=0
DELAY  100
MOVE  78  400
MOVE  77  200
DELAY  600
CLOSE

DELAY  50
MOVE  78  200
MOVE  80  200
MOVE  80  150
DELAY  550


OPEN
DELAY  50
MOVE  80  150
MOVE  74  300
MOVE  73  400
DELAY  850
ENDIF

GOTO 10
END
```

El programa solo cuando detecta un uno lógico en la entrada 8 comenzara su ejecución en caso contrario seguirá preguntando hasta tener un uno en la entrada 8. Las instrucciones se fueron ligando las posiciones de tal manera que no choque o intervenga en alguna etapa, cuando se encuentra en posición de agarrar o dejar la pieza se colocó un retardo de 0.5 segundos para que el brazo Scorbot deje la pieza con tranquilidad.

Para la estación de estampación se obtuvo el siguiente algoritmo (Diagrama 1) para su control y en base a este, se desarrolló el programa para el control.

Programa para la estación de estampación

```
STEP UNO
        IF                  I0.1
        THEN    SET         F0.0

                SET         O0.5
              RESET         O0.4


STEP TRES
```

```
    IF              F0.0
            AND     I1.3
            AND N   I1.1
    THEN    SET     O0.1
            SET     O0.5
            SET     O0.3
            SET     O0.6

            SET     O0.5
            SET     T1
             WITH   2s
STEP CUATRO

    IF          N       T1
    THEN        SET     O0.7
                SET     T2
                WITH    2s
STEP CINCO
    IF          N       T2
    THEN        RESET   O0.5
                RESET   O0.1
                RESET   O0.7
                 SET    T4
                   WITH 2s
STEP SEIS
    IF          N       T4
   THEN         RESET   O0.3
                SET     O0.2
                SET     O0.4
                SET     T6
                   WITH 2s
STEP OCHO
    IF          N       T6
    THEN        SET     O0.7
                SET     O0.5
                SET     T3
                   WITH 2s
STEP SIETE
     IF         N       T3
     THEN       SET     O0.7
                SET     T9
                  WITH  2s
STEP DIEZ
     IF         N       T9
     THEN       RESET   O0.6
                SET     T7
                  WITH  2s
STEP ONCE
     IF         N       T7
     THEN       RESET   O0.2
                RESET   F0.0
```

Diagrama1.-Algoritmo para control de la estación de estampación




## 11. Conclusiones

La colocación en los procesos repetitivos de los brazos de robot por mencionar al Intellitek Scorbot, es de gran utilidad, ya que estos nos permiten realizar secuencias repetitivas en tiempos exactos para la toma de las piezas y la colocación dentro de un área específica. Pero presentan un ligero problema para algunos trabajos, ya que como solo se siguen rutinas establecidas, se requiere agregar señales que indiquen el inicio y el final de cada algoritmo de control del eslabón de una cadena cinemática.


## Agradecimientos

Se le agradece por la atención y amabilidad, por los conocimientos aportados para desarrollar un buen articulo y por recibirnos en colima.

**M. en C.** Ricardo Fuentes Covarrubias
**M. en C.** Andrés Gerardo Fuentes Covarrubias


## 12. Bibliografia


[1] Barrientos, Peñín, Balaguer, Aracil; Fundamentos deRobótica; Ed. Mc Graw Hill, 1997.

[2] Eshed Robotec, Manual de usuario del SCORBOT -ER IX. 1994.

[3] Barrientos, Peñín, Balaguer, Aracil; Fundamentos de Robótica; Ed. Mc Graw Hill, 1997.

[4] Fu K.S.; et al..; Robótica. Control, detección, visión e Inteligencia. Ed. Mc Graw Hill, 1990.

[5] Spong M.W. and M. Vidyasagar. Robot Dynamic and Control. John Wiley and Sons, 1989.

# DESARROLLO DE ALGORITMOS DE VISIÓN ARTIFICIAL EMBEBIDA EN C# PARA LA CLASIFICACIÓN DE OBJETOS POR MÉTODOS MORFOLÓGICOS PARA EL BRAZO DE ROBOT MITSUBISHI RV-M1

[1]Ledesma Natera Edgar Giovanni [2] Fuentes Covarrubias Andrés Gerardo , [3] Fuentes Covarrubias Ricardo

*[1]Avenida Tecnológico numero 2000, Colonia Solidaridad, Fresnillo Zacatecas.*
*[2 3] Universidad de Colima, Facultad de Ingeniería Mecánica y Eléctrica, Km. 9 Carr. Colima-Coquimatlán, Colima, México.*
[1]giovanniln_45@hotmail.com, [2]fuentesg@ucol.mx, [3]fuentesr@ucol.mx

## Resumen

*En este artículo se presenta una metodología para la manipulación de un robot Mitsubishi RV-M1. En donde se involucra el diseño e implementación del software para la selección de objetos por medio de sus atributos, necesario para la ordenación de dichos objeto. La captura de un frame en tiempo real incluye, la segmentación y la binarizacion para poder tratar la imagen y realizar las técnicas de inteligencia para poder realizar la trayectoria correcta de ordenación encontrada.*

Palabras clave: Frame, Segmentación, Binarización, Imagen, Inteligencia, Ordenación.

.

## Abstract

*This article presents a methodology for handling a Mitsubishi RV-M1 robot. Where it involves the design and implementation of software for the selection of objects by their attributes required for the management of such object. Capturing a real-time frame includes segmentation and binarization to treat the image and conduct intelligence techniques to make the correct management ordenation found.*

**KEY WORDS**: Frame, Segmentation, Binarization, Image, Intelligence, Ordenation.

## 1. Introducción

La visión humana es el sentido más versátil, ya que es la que en mayor proporción nos aporta información muy precisa. Por ende de ahí surge el interés de realizar algoritmos para conocer el mundo mirando al igual que el ser humano, la única diferencia que se ocupara para la visión artificial una computadora en vez de un cerebro para adquirir el conocimiento.

La visión artificial aplicada a la industria está más orientada a realizar tareas de inspección, control de calidad y control de procesos. Mediante el procesamiento de imágenes es posible detectar la presencia o ausencia de atributos de objetos tales como defectos, dimensiones, formas y colores.

En el ámbito industrial, estas plataformas tienen múltiples aplicaciones, como, por ejemplo, medición de precisión y calibración de piezas, que van desde partes mecánicas hasta frutas y moluscos de exportación; análisis de color en pinturas, licores, termografías, códigos de colores, separación de piezas por color; detección de defectos en etiquetas, envases, logotipos, maderas, láminas continuas de cartón y acero; conteo de elementos según calibre, color, forma, tamaño; verificación y reconocimiento de caracteres, códigos de barra 1D, 2D y Data Matrix, fechas de vencimiento, número de lote.

## 2. Robot Mitsubishi RV-ML

### 2.1 Instalación general

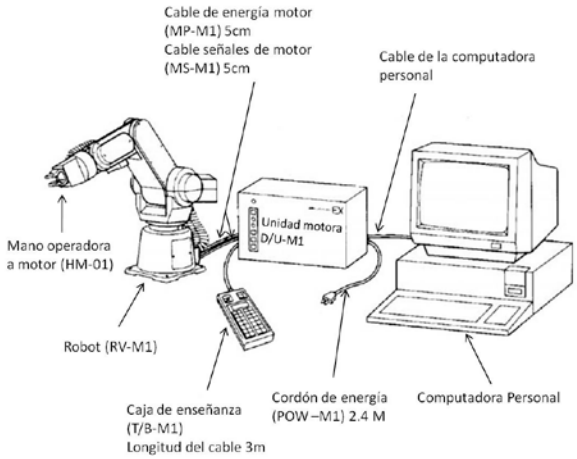



**Figura 1.** Instalación General

### 2.2 Nomenclatura

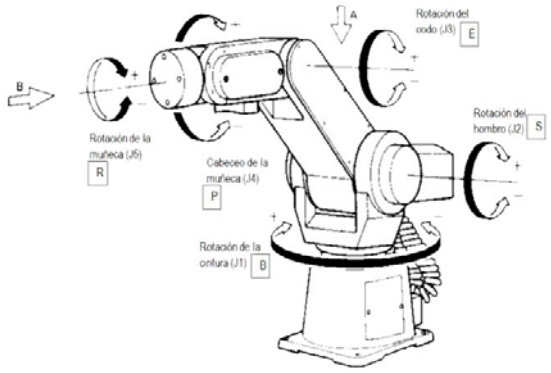



**Figura 1.1.** Nomenclatura (Vista externa).

### 2.4 Espacio de operación

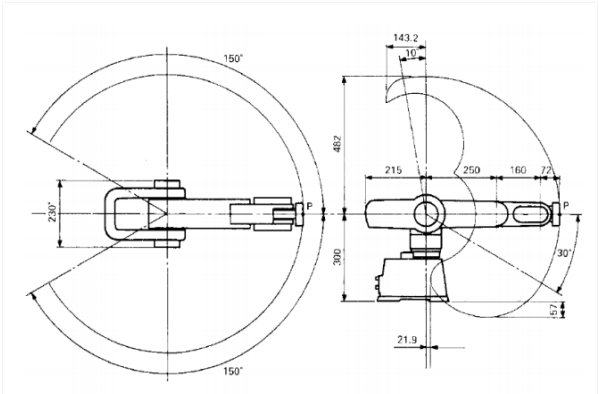



**Figura 1.2.** Nomenclatura (Vista externa).

El espacio de operación supone que la mano no está instalada en el robot. (Trazo del punto P).

La operación de cabeceo de la muñeca se puede restringir en algo del área dependiendo de las posiciones del brazo superior y antebrazo.

La operación manual (caja de enseñanza) debe realizarse con especial cuidado porque la muñeca puede interferir con la base del robot y la superficie del piso.

### 2.3 Operaciones básicas

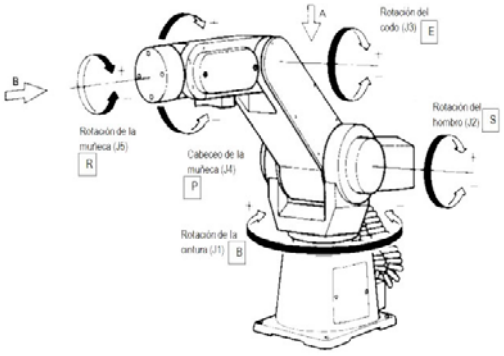



**Figura 1.3.** Operaciones del sistema articulado.

El sentido positivo de operación de los ejes J1 Y J5 es el de las manecillas del reloj visto desde las flechas A Y B, respectivamente.

El sentido positivo de operación de los ejes J2, J3 Y J4 es en la dirección hacia arriba del brazo y la muñeca.

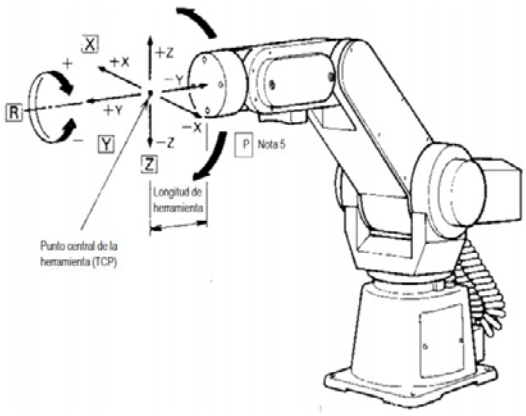



**Figura 1.3.1.** Operaciones en el sistema cartesiano de coordenadas.

El TCP (punto central de la herramienta) se mueve en línea recta en el sistema cartesiano.

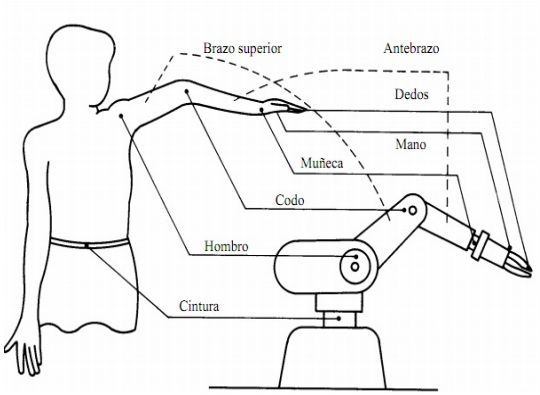



**Figura 1.3.2.** Comparación brazo humano y brazo robótico.

## 2.5 Unidad motora

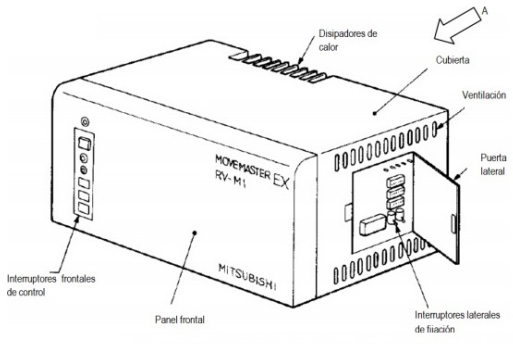



**Figura 1.4** Nomenclatura (Vista externa).

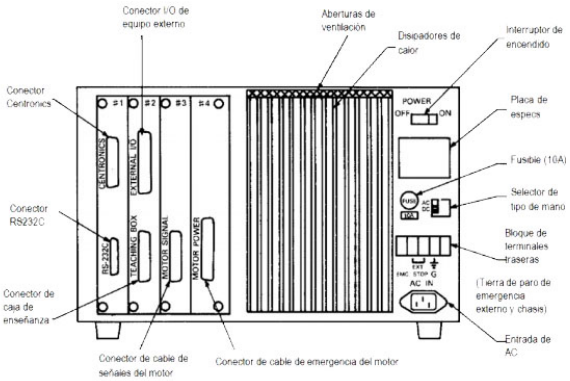



**Figura 1.4.1.** Nomenclatura (Unidad motora).

**Tabla 1.** Especificaciones Estándar de la Unidad Motora

| Item | Especificaciones |
|---|---|
| Método de Enseñanza | Sistema de lenguaje de programación. |
| Método de Control | Sistema de control de posición PTP usando servo motores DC. |
| Numero de Ejes de Control | 5 ejes ( +eje opcional) |
| Detección de Posición | Sistema codificador de pulsos |
| Retorno al Origen Fijación de Origen | Interruptores límite y codificadores de pulsos. |
| Función de Interpolación | Interpelación articulada, interpolación lineal. |
| Fijación de Velocidad | 10 pasos (max. 1000 mm/s) |
| Numero de posiciones | 629 (6 KB) |
| Numero de Pasos de Programa | 2048 (16 KB) |
| Almacenamiento de Datos | Escriba en el EP-ROM usando el escritor interconstruido de EP-ROM o almacene en la RAM estática respaldada por batería. |
| Equipo de Enseñanza de Posición | Caja de enseñanza (opción) o computadora personal |
| Equipo de Programación | Computadora personal |
| I/O Externa | I/O de propósitos generales, 8 puntos cada uno. Señales sincrónicas de propósitos generales I/O no dedicado. Energía eléctrica para estas I/O deberán prepararse por el usuario (12v a 24v DC) |
| Interfaces | 1 interface paralela (conformada a Centronics) 1 interface serial (conformada a RS232C) |
| Paro de Emergencia | Usándolo cualquiera de los interruptores de: control frontal, caja de enseñanza y block de terminales trasero. |
| Control de Mano | Mano operada a motor o mano operada neumáticamente (usando un solenoide AC). |
| Control de Freno | Eje J2 (hombro), eje J3 (codo). |
| Fuente de Energía | 120V/220V230V/240V AC 0.5KVA |
| Temperatura Ambiente | 5°C a 40°C |
| Peso | Aprox. 23 Kgf |
| Tamaño | 380 (W) x 331 (D) x 246 (H) mm |

## 2.6 Caja de enseñanza

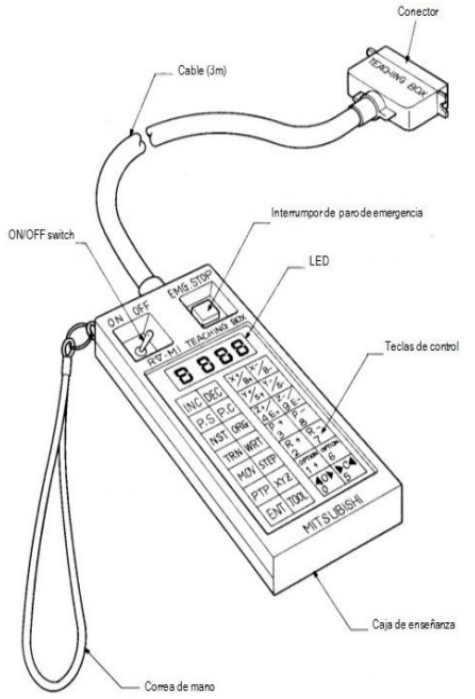



**Figura 1.4.2.** Nomenclatura (Caja de enseñanza).

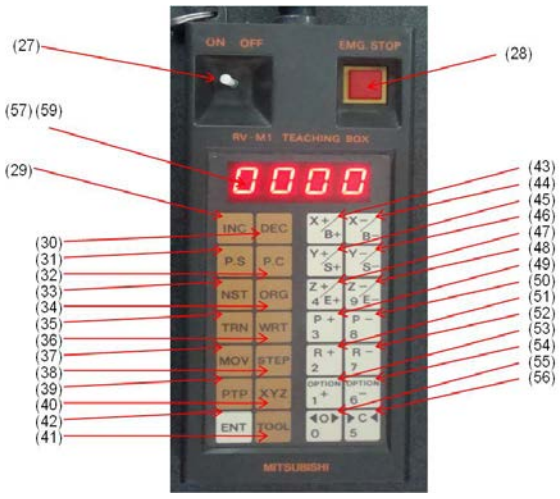



**Figura 1.4.2.** Caja de enseñanza

**(27) ON/OFF (Interruptor de energía)**

Selecciona el habilitar o deshabilitar las teclas de la caja de enseñanza. Cuando el robot va a ser operado usando la caja de enseñanza, ponga este interruptor en ON. Durante la corrida del programa o cuando el robot está siendo controlado por medio de comandos enviados desde una computadora personal, apague el interruptor en OFF. También puede borrarse una entrada errónea de tecla apagando el interruptor a OFF.

**(28) EMG. STOP (Interruptor de paro de emergencia)**

Botón interruptor usado para un paro de emergencia del robot (la señal internamente queda bloqueada cuando este botón se oprime). Cuando se oprime el interruptor, el robot inmediatamente se detiene y el indicador LED de error parpadea (modo de error 1) El LED4 dentro de la puerta lateral de la unidad motora también se enciende.

**(29)**

INC + ENT

Mueve el robot a una posición predefinida con un número de posición mayor que el presente. Para mover el robot a través de una cierta secuencia, repita la secuencia de tecleo. (Comando "IP.").

**(30)**

DEC + ENT

Mueve el robot a una posición predefinida con un número de posición menor que el presente. Para mover el robot a través de una cierta secuencia, repita la secuencia de tecleo. (Comando "DP.").

**(31)**

P.S. + NUM + ENT

Define las coordenadas de la posición presente del robot en una posición con el número especificado. Si el mismo número es asignado a dos posiciones diferentes, la definida a la última toma precedencia. (Comando "HE.").

**(32)**

P.C + NUM + ENT

Elimina el contenido de una posición con el número especificado.

**(33)**

NEST + ENT

Regresa el robot al origen. (Comando "NT.").

**(34)**

ORG + ENT

Mueve el robot a la posición de referencia en el sistema de coordenadas cartesianas. (Comando "OG.").

**(35)**




Transfiere el contenido del EPROM del usuario (programa y datos de posición) instalado en el SOC2 del panel lateral en la unidad motora a la RAM de la unidad motora. (Comando "TR.").

**(36)**




Escribe el programa y los datos de posición escritos en la RAM de la unidad motora en el EPROM del usuario instalado en el SOC2 del panel lateral en la unidad (Comando "CR. ").

**(37)**




Mueve el extremo de la mano a una posición especificada. (Comando MO.") La velocidad de movimiento es equivalente a SP4.

**(38)**




Ejecuta el programa paso por paso empezando en el número de línea especificado. Para hacer que el programa sea ejecutado en secuencia de un paso a otro, repita la secuencia de tecleo. Note que, en este caso, no es necesario dar "enter" a los números. Se causa un error modo II si ocurre un error mientras los pasos están siendo ejecutados.

**(39)**




Selecciona la operación de la caja de enseñanza articulada. Cuando se oprime esta tecla, la operación de cualquier tecla de la caja en lo sucesivo efectúa un movimiento en cada articulación. En la condición inicial cuando se enciende la caja de enseñanza, se establece este estado PTP.

**(40)**




Selecciona la operación de la caja de enseñanza cartesiana. Cuando se oprime esta tecla, la operación de cualquier tecla de la caja en lo sucesivo efectúa un movimiento axial en el sistema cartesiano de coordenadas.

**(41)**



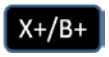

Selecciona la operación joq de la herramienta. Cuando se oprime esta tecla, la operación jog de cualquier tecla en lo sucesivo efectúa un movimiento axial en el sistema de coordenadas de la herramienta (movimiento de avance/retracción en la dirección de la mano).

**(42)**




Completa la entrada de cada tecla de (29) a (38) para efectuar la operación correspondiente.

**(43)**



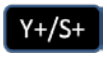

Mueve el extremo de la mano en el sentido positivo del eje X (a la izquierda si se está viendo hacia el frente del robot), en la operación jog cartesiana y barre la cintura en sentido positivo (a favor de las manecillas del reloj si se ve desde arriba del robot) en la operación jog articulada

**(44)**



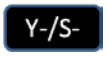

Mueve el extremo de la mano en el sentido negativo del eje X (a la derecha si se está viendo hacia el frente del robot), en la operación jog cartesiana y barre la cintura en sentido negativo (contralas manecillas del reloj si se ve desde arriba del robot) en la operación jog articulada.

**(45)**



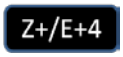

Mueve el extremo de la mano en el sentido positivo del eje Y (hacia el frente del robot), en la operación jog cartesiana y gira el hombro en el sentido positivo (hacia arriba) en la operación jog articulada.

**(46)**




Mueve el extremo de la mano en el sentido negativo del eje Y (hacia atrás del robot), en la operación jog cartesiana y gira el hombro en el sentido negativo (hacia abajo) en la operación jog articulada.

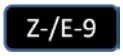
**(47)**




Mueve el extremo de la mano en el sentido positivo del eje Z (directamente hacia arriba), en la operación jog cartesiana, voltea el codo en sentido positivo (hacia arriba) en el jog articulado y avanza la

mano en el jog de herramienta. Sirve también como la tecla numérica “4”.

**(48)**




Mueve el extremo de la mano en el sentido negativo del eje Z (directamente hacia abajo), en el jog cartesiano, voltea el codo en sentido negativo (hacia abajo) en el jog articulado y retrae la mano en el jog de herramienta. Sirve también como la tecla numérica "9.".

**(49)**




Voltea el extremo de la mano, manteniendo su posición presente determinada por el comando "TL", en el sentido positivo (hacia arriba) en el jog cartesiano y dobla la muñeca (cabeceo de muñeca) en el sentido positivo (hacia arriba) en el jog articulado. Sirve también como la tecla numérica “3”.

**(50)**




Voltea el extremo de la mano, manteniendo su posición presente determinada por el comando "TL", en el sentido negativo (hacia abajo) en el jog cartesiano y dobla la muñeca (cabeceo de muñeca) en el sentido negativo (hacia abajo) en el jog articulado. Sirve también como la tecla numérica "8".

**(51)**




Tuerce la muñeca (balanceo de muñeca) en el sentido positivo (a favor de las manecillas del reloj viendo hacia la superficie de montaje de la mano). También como la tecla numérica "2".

**(52)**




Tuerce la muñeca (balanceo de muñeca) en el sentido negativo, (contra las manecillas del reloj) viendo hacia la superficie de montaje de la mano) Sirve también como la tecla numérica "7.".

**(53)**




Mueve el eje opcional en sentido positivo. Sirve también como la tecla numérica "1.".

**(54)**




Mueve el eje opcional en sentido negativo. Sirve también como la tecla numérica "6.".

**(55)**




Abre el agarrador de la mano. Sirve también como la tecla numérica"0.".

**(56)**




Cierra el agarrador de la mano. Sirve también como la tecla numérica "5.".

## 3 C#

C# (pronunciado si sharp en inglés) es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET, que después fue aprobado como un estándar por la ECMA (ECMA-334) e ISO (ISO/IEC 23270). C# es uno de los lenguajes de programación diseñados para la infraestructura de lenguaje común.

Además c-sharp permite una buena interacción con el brazo robótico para la realización del proyecto, ya que como el sistema de visión artificial involucra funciones graficas y conexión por puertos, el lenguaje se presta para realizar dichas funciones.

## 3.1 EMGU CV

Es una librería que es compatible con C-Sharp la cual contiene una gama de funciones para la búsqueda de objetos de un frame, y hace una mayor exactitud en los procedimientos de localización de objetos.

## 3.2 MySQL

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL de la GNU. Su diseño multihilo le permite soportar una gran carga de forma muy eficiente. MySQL fue creada por la empresa sueca MySQL AB, que mantiene el copyright del código fuente del servidor SQL, así como también de la marca.

Aunque MySQL es software libre, MySQL AB distribuye una versión comercial de MySQL, que no se diferencia de la versión libre más que en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario, ya que de no ser así, se vulneraría la licencia GPL.

Este gestor de bases de datos es, probablemente, el gestor más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración.

Para este proyecto resulto ser una herramienta importante, ya que nos permite almacenar los puntos requeridos para realizar la calibración.

## 3.3 Descripción de los materiales y equipos a emplear

- Objetos para el procedimiento de ordenación.
- Web Cam
- Entorno de Trabajo donde se encuentran situados los objetos y la cámara para la obtención de la imagen.
- Robot Mitsubishi RV-M1
- Teaching Box
- Computadora para el control de visión.
- Base de datos para guardar la calibración del entorno de trabajo.

# 4. Método

## 4.1 Entorno de trabajo

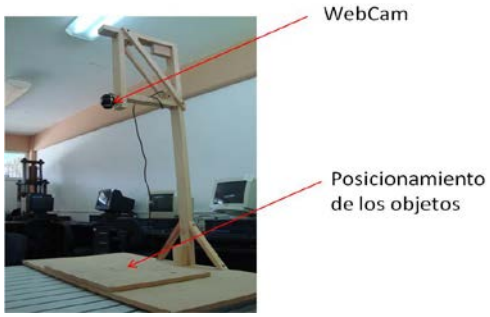



**Figura 2.1.** Entorno de trabajo lateral.

Se tiene el entorno en el cual por medio de la webcam obtendrá los frame en tiempo real de la posición de los objetos.

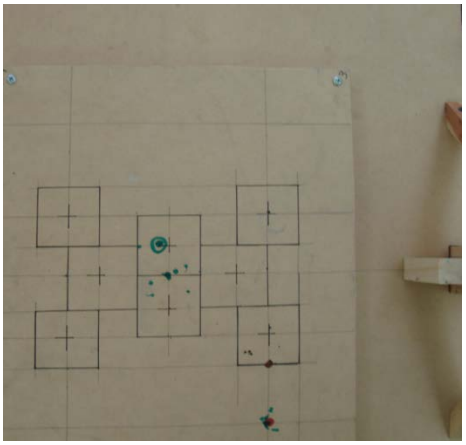


**Figura 2.1.2.** Vista Webcam.

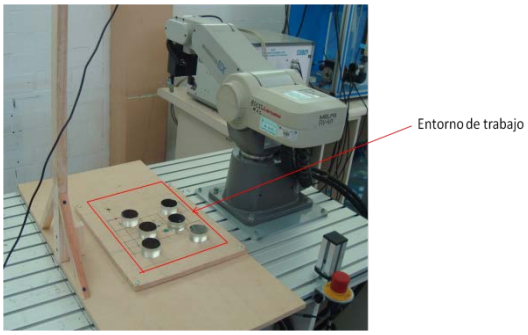



**Figura 2.1.3.** Entorno de trabajo en tiempo real.

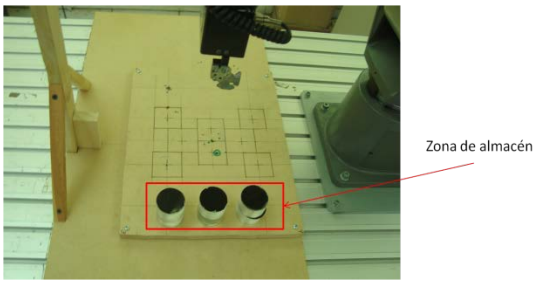



**Figura 2.1.4.** Zona de almacén

## 4.2 Aproximaciones y posiciones

Para realizar el proceso de ordenación de objetos, se tiene que tener varias posiciones del brazo robótico guardas para poder realizar el procedimiento y tener en cuenta las aproximaciones dentro del entorno de trabajo.

Estas aproximaciones nos sirven para delimitar el espacio del robot y así evitar coaliciones que se puedan dar dentro del desarrollo de ordenamiento.

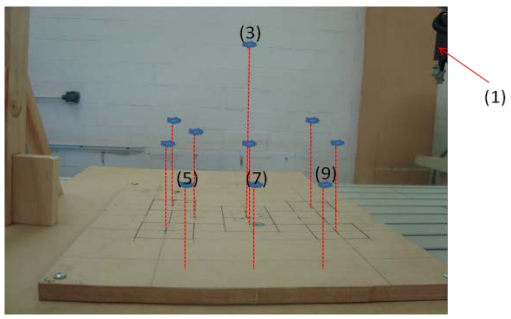



**Figura 3.** Aproximaciones.

(1) Es la posición del descanso del robot, para que la cámara pueda obtener el frame de los objetos y el brazo no obstruya en la detección de los objetos. Ya que si el brazo se deja en el centro del frame, la cámara al capturar el frame se podrá encontrar en el mismo objetos, ocasionado así un error ya que en la posición del brazo buscara objetos que no son mismos del entorno de trabajo.

(3) Es la posición central, en donde el brazo robótico se dirigirá para encontrar y tomar el objeto, al realizar esta acción vuelve a esta posición para pasar a la posición de acomodo.

(5) En esta posición se acomoda la primera columna de dos objetos, el total de columnas de es de 3 con dos objetos por cada columna respectivamente.

(7) En esta posición se acomoda la segunda columna de dos objetos.

(9) En esta posición se acomoda la tercera columna de dos objetos.

Las demás aproximaciones que no se tienen enumeradas es debido a que el brazo robótico en esas partes funciona inteligentemente, esas posiciones no se tiene guardadas si no que se le indican de acuerdo al objeto encontrado.

## 4.3 Segmentación de imagen

Obteniendo un frame de video en tiempo real, se realiza el procesamiento de segmentación que pasa por varias fases, este proceso consiste en tener la imagen y tratarla por varios métodos para realizar un correcto análisis de los objetos y así una mayor exactitud en el ordenamiento.

Para estos procesos ya mencionados se utilizo el modelo RGB que es fundamental para realizar la segmentación este proceso es un modelo de color basado en la síntesis aditiva, con el que es posible representar un color mediante la mezcla por adición de los tres colores luz primarios. El modelo de color RGB no define por sí mismo lo que significa exactamente rojo, verde o azul, por lo que los mismos valores RGB pueden mostrar colores notablemente diferentes en diferentes dispositivos que usen este modelo de color. Aunque utilicen un mismo modelo de color, sus espacios de color pueden variar considerablemente.

Para indicar con qué proporción mezclamos cada color, se asigna un valor a cada uno de los colores primarios, es frecuente que cada color primario se codifique con un byte (8 bits). Así, de manera usual, la intensidad de cada una de las componentes se mide según una escala que va del 0 al 255.

Por lo tanto, el rojo se obtiene con (255, 0,0), el verde con (0, 255,0) y el azul con (0, 0,255), obteniendo, en cada caso un color resultante monocromático. La ausencia de color lo que nosotros conocemos como color negro se obtiene cuando las tres componentes son 0, (0, 0,0).

Para la obtención de niveles de color de una imagen, basta con sacarlo por cada pixel, se debe tener claro que una imagen en sí, es una matriz determinada por el tamaño que nosotros le damos.

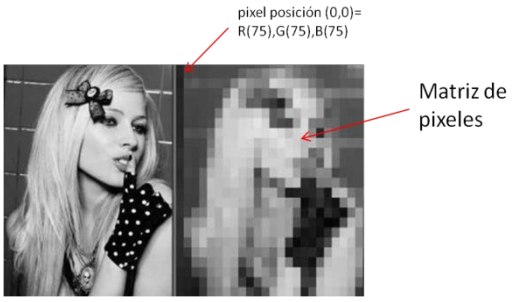



**Figura 3.1.** Valores pixeles.

**Escala de Grises:** Obteniendo los valores del modelo RGB se procede a calcular la escala de grises, aplicando la siguiente fórmula:

$$Y = R * 0.59 + G * 0.3 + B * 0.11$$

En donde el valor 0.59 que se le otorga al color rojo es el porcentaje de percepción por parte del ojo humano, para el color verde es el 0.3 y para el color azul es el 0.11.




**Figura 3.2.** Imagen escala de grises.

**Binarizacion:** Esta técnica consiste en calcular la intensidad de gris que correspondería a cada píxel,

pero definiendo previamente un umbral a partir del cual se saturaría el color.

Si la intensidad fuera mayor al umbral saturará a blanco, y si fuera menor o igual saturará a negro.

Esta técnica se utiliza buscando un umbral con el cual se muestren todos los objetos que se identificarán, no pudiendo distinguir uno de otro, pero si distinguiéndolos de la información que no es útil del área de trabajo.

Si el valor del pixel en grises es mayor al umbral, esa posición del pixel la convertirá de gris a blanco de lo contrario lo convertirá de gris a negro, eliminando así la información que no es necesaria para el proceso.

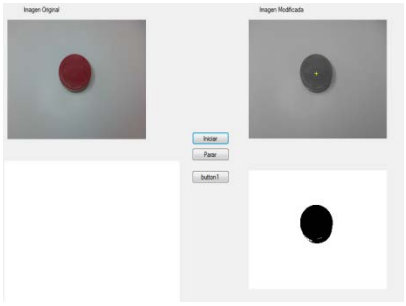



**Figura 3.2.1.** Imagen binarizada.

## 4.4 Centroide

El centroide es un punto que define el centro geométrico de un objeto. Su localización puede determinarse a partir de formulas semejantes a las utilizadas para determinar el centro de gravedad o el centro de masa del cuerpo, el centroide nos ayuda a encontrar el punto en el que se concentra las fuerzas que actúan sobre una figura irregular, o figuras geométricas no muy conocidas, por ejemplo el centroide nos ayudaría a encontrar el punto en el que se concentran las fuerzas de un puente.

Determinar el centroide nos permitirá conocer las coordenadas de x,y dentro del frame para con ello, realizar los cálculos pertinentes para su localización.

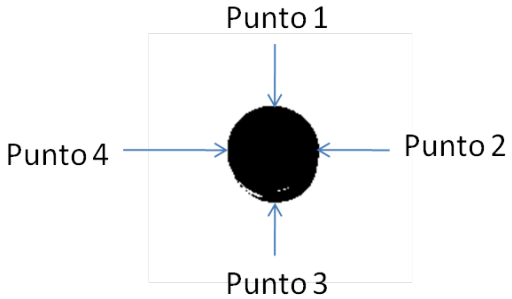



**Figura 3.3.** Determinación de puntos utilizados para la ubicación del centroide.

Por medio de la binarizacion de un frame se podrá obtener los puntos que rodean el objeto realizando las siguientes operaciones para el cálculo del centroide.

x= Punto4+ ( ( Punto4 – Punto2) / 2)

y= Punto1+ ( ( Punto1 – Punto3) / 2)

Con las siguientes formulas se obtuvo el cálculo del centroide, que se suma la distancia entre los puntos entre dos para encontrar el centroide x,y del objeto.

# 5. Sistema de Visión Artificial

## 5.1 Calibración

Para el comienzo de nuestro sistema de visión artificial, se realizo la calibración correspondiente, para determinar cuántos milímetros corresponden a cada pixel, para ello en el entorno de trabajo se marcaron dos puntos, que obtuvimos el valor en milímetros y el valor en pixeles de cada uno.

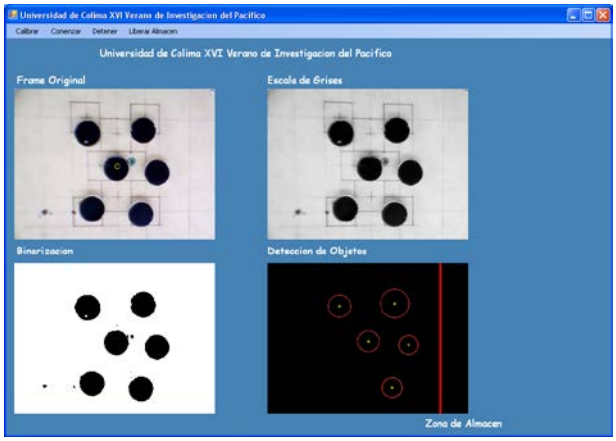



**Figura 4.1.** Ventana principal, obteniendo el frame original, escala de grises, binarizacion y los objetos encontrados.

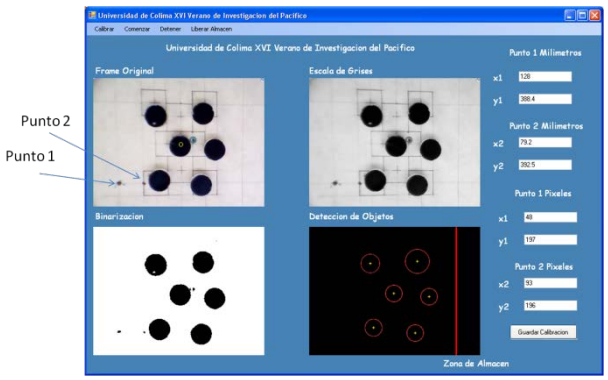



**Figura 4.1.1.** Ventana para el comienzo de calibración.

El punto 1 y 2 en milímetros para “x” y “y” se introducen manualmente situando el robot en esos puntos, y por medio del comando “WH” se le indica al robot que posición esta en x, y. Enseguida obteniendo esos valores se introducen en las cajas en blanco correspondiente en milímetros.

El punto 1 y 2 en pixeles se obtiene automáticamente, solo se debe de situar el cursor sobre el frame original y dar un clic en el primer punto, mismo caso para el punto número 2 y automáticamente obtendremos los valores en las cajas en blanco.

Ya teniendo estos puntos obtuvimos la escala, esta escala nos proporcionara el valor de cuantos milímetros corresponden a un pixel obteniéndolos con la siguientes fórmulas:

$$\Delta milimetros = Punto2mmx - Punto1mmx$$

$$\Delta pixeles = Punto2pixx - Punto1pixx$$

**Donde:**

Punto2mmx: Es el valor del punto dos respecto a x en milímetros.

Punto1mmx: Es el valor del punto uno respecto a x en milímetros.

Punto2pixx: Es el valor del punto dos respecto a x en pixeles.

Punto1pixx: Es el valor del punto uno respecto a x en milímetros.

$$escala = \Delta milimetros / \Delta pixeles$$

Al momento de tener la escala, esta misma se guarda dentro de la base de datos, para ya no tener que hacer el mismo proceso cada vez que se ejecute el programa. Si se desea realizar otra calibración el sistema automáticamente actualiza la calibración, dejando la última agregada por default.

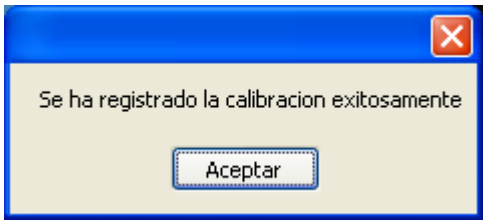



**Figura 4.1.2.** Mensaje en el cual todos los mensajes se guardaron exitosamente en la base de datos.

## 5.2 Dirección del brazo hacia los objetos

Con la escala encontrada se procedió a realizar el cálculo de dirigir el robot hacia el objeto encontrado, para ejecutar la acción, el robot se dirigió en la posición central del entorno de trabajo, en este caso a la aproximación 3 que se encuentra en el centro del entorno de trabajo.



**Figura 4.2.** Posición o aproximación 3.

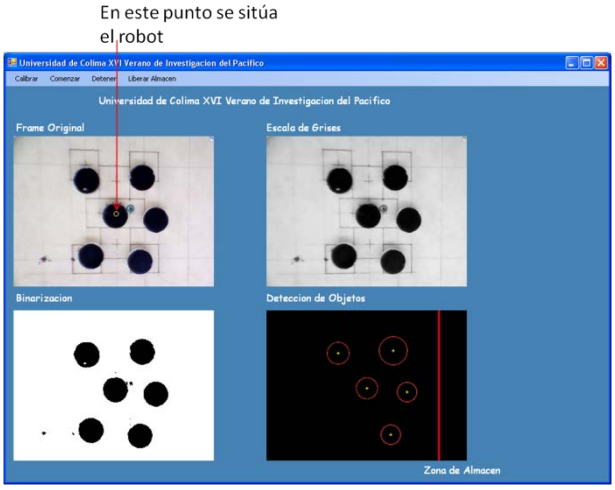



**Figura 4.3.** Posición del robot para selección de objetos.

Como el punto central es conocido en milímetros, lo que se necesita es encontrar el centroide de los objeto. Los centroides de los objetos los obtenemos en pixeles, que es el valor donde se encuentran en el frame. El paso siguiente es convertir el valor del centroide de pixeles a milímetros y solo sacar la distancia de la aproximación o posición 3 a la del centroide del objeto.

Esa distancia es la que el robot recorrerá del punto amarillo al cenotride del objeto y en ese punto recogerá la pieza y la llevara a la columna correspondiente dependiendo de la cantidad que haya en cada columna.

Las columnas pueden ser la aproximación o posición 5, 7 o 9, que se tienen ya almacenadas para el ordenamiento.

El proceso de ordenación se realiza en tiempo real, con un máximo de 6 objetos para el almacen, cada columna ocupa dos objetos.



**Figura 4.4.1** Posición del robot.

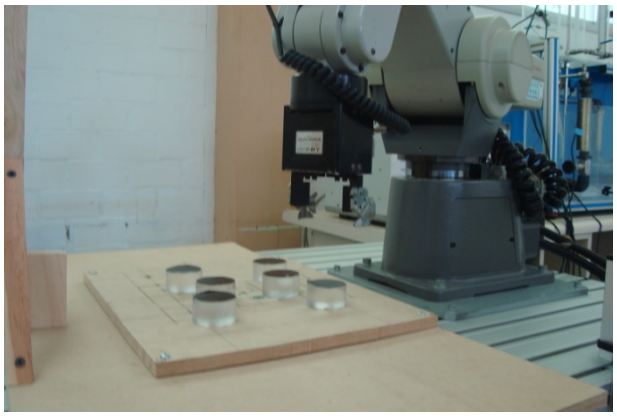


**Figura 4.4.2.** Aproximación al objeto

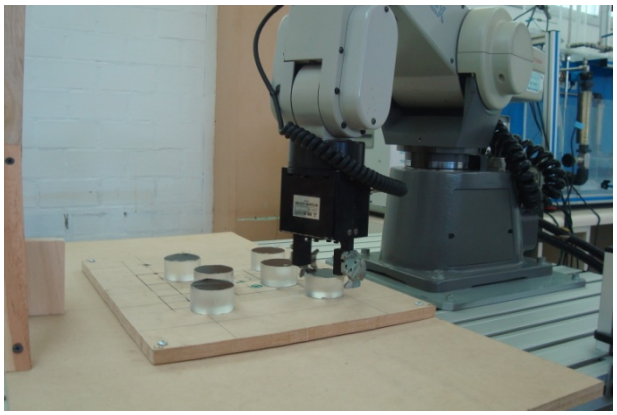


**Figura 4.4.3.** Obtención del objeto



**Figura 4.4.4.** Regreso a la aproximación.



**Figura 4.4.5.** Regreso a la aproximación 3

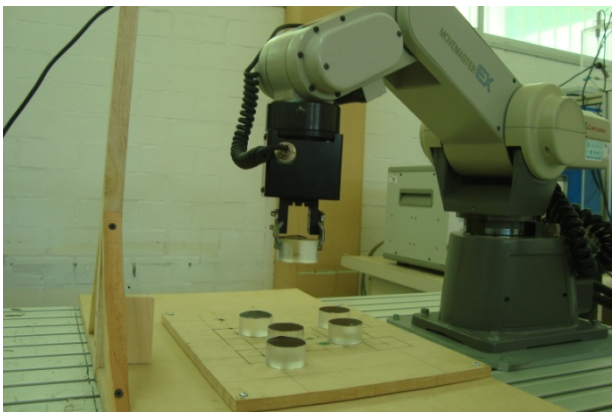


**Figura 4.4.6.** Regreso a la aproximación 3



**Figura 4.4.7.** Aproximación para acomodo.

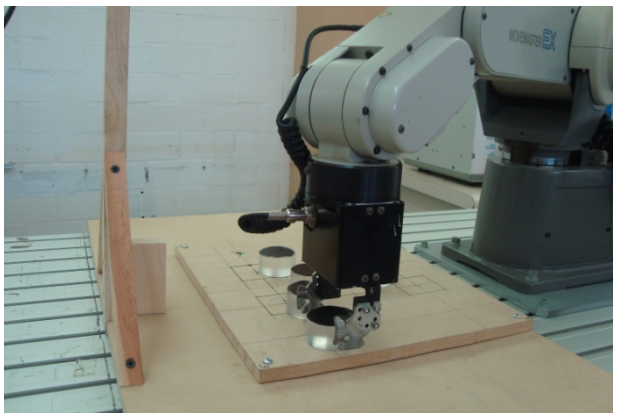


**Figura 4.4.8.** Aproximación para acomodo en almacén.

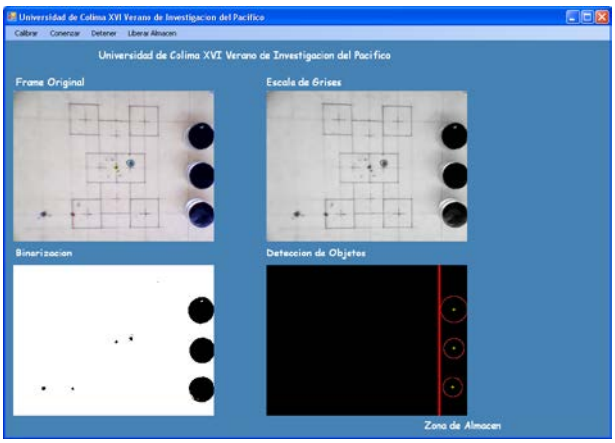



**Figura 4.4.9.** Ordenación de objetos terminada. (Vista desde el sistema de visión).

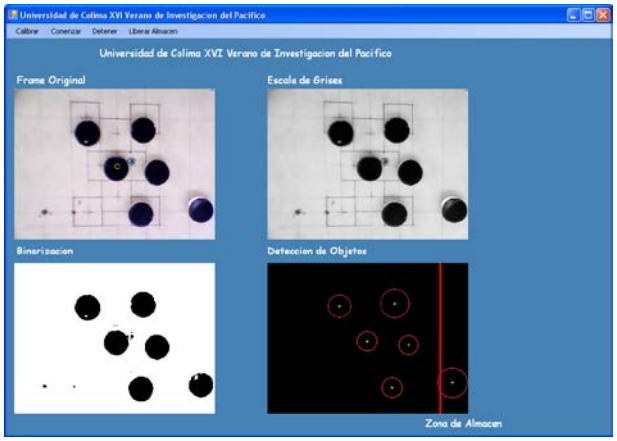



**Figura 4.4.9.** Vista desde el sistema de visión.

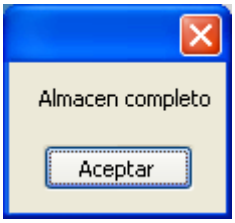



**Figura 4.4.11.** Mensaje donde el almacén esta completo.

Al tener el almacén completo, el robot se trasladara a la posición o aproximación 1, esperando a liberar el almacén, ya que si detecta alguna pieza, el robot se quedara inactivo hasta que sean liberados los objetos.



**Figura 4.4.10.** Ordenación de objetos terminada.

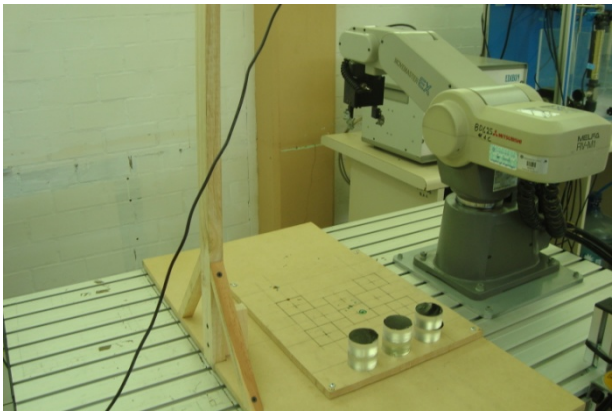


**Figura 4.4.12.** Robot en posición de descanso esperando liberación de almacén.

Esta posición de descanso también se da si en algún momento se cuentan con pocos objetos, el robot se irá a esa posición hasta encontrar un objeto llevándolo a su área correspondiente.

## 6. Conclusiones

Con el sistema de visión artificial embebida, se ha podido tener un mayor calculo en la obtención de características de los objetos para la operación de clasificación de objetos, con estos cálculos es más preciso realizar procedimientos, ya que con la visión humana pueden existir errores de geometría del objeto y así impedir un procedimiento correcto.

## AGRADECIMIENTOS

Agradezco enormemente a mis asesores M.C. Andrés Gerardo Fuentes Covarrubias, M.C. Ricardo Fuentes Covarrubias, por todo el conocimiento que me otorgaron, en verdad muchas gracias a los dos por que aprendí muchísimo en este verano de investigación, sin ellos este proyecto no sería un éxito, gracias maestros por la dedicación y el tiempo que me dieron muchas gracias.

A mi familia que estando tan lejos, me apoyaron en todos los sentidos, como en lo económico y en lo sentimental, otro agradecimiento a la universidad de colima por facilitarnos los materiales necesarios para l realización de este proyecto.

Y ya por ultimo a todas las personas de colima y de otros estados que también sirvieron de apoyo para la realización del proyecto en verdad a todos los mencionados muchas gracias.

## REFERENCIAS

[1] http://docs.gimp.org/es/gimp-image-convert-grayscale.html. Imágenes en Escala de Grises.

[2 ]http://dev.mysql.com/doc/refman/5.0/es/index.html. Manual de MySQL.

[3] http://www.emgu.com/wiki/index.php/Main_Page. EMGU

[4] http://www.cnad.edu.mx/sitio/matdidac/md/mecanica/opeba srv-m1.pdf
Manual de Operación Básico Robot Mitsubishi RV-M1
Ing. Uriel Gutiérrez Salazar.

[6] http://www.cnad.edu.mx/sitio/matdidac/md/mecanica/opeba srv-m1.pdf
Manual de Operación Básico Robot Mitsubishi RV-M1
Ing. Uriel Gutiérrez Salazar.

[7] Manejo de Imágenes Visual C# Studio 2010 Express Tutorial. A. Gerardo, Ricardo Fuentes C.

[8] EMGU CV Tutorial 1, 2, 3.
Andrés Gerardo Fuentes Covarrubias – Ricardo Fuentes Covarrubias.

## BIOGRAFÍA DEL AUTOR

### Ledesma Natera Edgar Giovanni

Nacido Fresnillo, Zacatecas el dia 3 de noviembre de 1989, estudio el bachillerato en CBtis #1 En fresnillo, Zacatecas, Actualmente se encuentras estudiando la carrera de Ingeniería en Sistemas Computacionales en el Instituto Tecnológico Superior de Fresnillo, cursando el 9no Semestre.

# PUESTA EN FUNCIONAMIENTO DEL KIT LV-100 PARA INSTRUMENTACIÓN VIRTUAL USANDO LABVIEW

[1]Angulo Molina Ilda Fabiola [2]Torres Ruiz Hugo [3]Fuentes Covarrubias Andrés Gerardo , [4]Fuentes Covarrubias Ricardo.

*[1]Universidad Autónoma de Nayarit, Edificio de la Unidad Académica de Economía. Ciudad de la Cultura Amado Nervo, Tepic, Nayarit. México. C.P. 63155.*
*[2]Instituto Tecnológico de Estudios Superiores de Zamora, Km. 7 Carretera Zamora – La Piedad, Zamora, Michoacán.*
*[3 4] Facultad de Ingeniería Mecánica y Eléctrica - Universidad de Colima*
*Km 9 Carr. Colima – Coquimatlán C. P. 28400 – Coquimatlán, Colima.*
*[1]fabiola.uan.2010@gmail.com,[2]hutorz.25@hotmail.com,*
*[3]fuentesg@ucol.mx, [4]fuentesr@ucol.mx*

## Resumen

*El siguiente documento presenta el desarrollo de una serie de prácticas de control utilizando el software de programación visual LabVIEW y como elemento principal de interfaz de comunicación, el kit LV-100, mismo que contiene todos los elementos de entrada y salida necesarios para el envío y recepción de datos. Para comunicar la PC con el kit se usaron los puertos de comunicación serial y paralelo. Así mismo se muestra el proceso por el cual se creó una fuente de alimentación para dicho kit y finalmente se hace una demostración de un VI que realiza las funciones de procesamiento de imágenes a través de una cámara externa.*

Palabras clave: instrumentación virtual, LED, switch, display, fuente, puerto serial, puerto paralelo, entrada/salida, arreglo.

## Abstract

*The following document presents the development of a series of control practices using the LabVIEW visual programming software and as a main communication interface, the LV-100 kit, it contains all the elements of input and output necessary for sending and receiving data. To communicate PC with the kit were used serial communication ports and parallel ports. It also shows the process by which created a power source for the above kit and finally a demonstration of a VI that performs the image processing functions via an external camera.*

Key words: virtual instrumentation, LED, switch, display, source, serial port, parallel port, Input/Output, array.

## 1. Introducción

El desarrollo de aplicaciones, en especial de electrónica implica el uso de una gran variedad de elementos que llegan a complicar su desarrollo, en especial por el espacio físico que pueden llegar a ocupar y el costo de cada componente. Es por eso que actualmente se recurre al uso de nuevas tecnologías que faciliten la tarea del desarrollo de aplicaciones. Uno de los softwares empleados es el creado por la empresa National Instruments, llamado LabView, que es con el cual se trabajará, dicho programa permite simular el funcionamiento de componentes físicos reales y permite el desarrollo de programas de control, estos son conocidos como instrumentos virtuales. Lo anterior es posible apoyándose en hardware, tales como tarjetas de adquisición de datos, instrumentos accesibles vía GPIB (General Purpose Interface Bus) y RS-232. Este programa tiene varias facilidades como funciones de un alto nivel de programación y elementos gráficos que simplifican la tarea de programación y elaboración del panel frontal como cualquier software de programación pero gráfico. El presente proyecto pretende mostrar el funcionamiento del kit LV-100 por medio del software de programación visual LabVIEW a través de la representación de algunas aplicaciones prácticas, mismas que fueron tomadas de referencia del manual entrenador para la interfaz de entrada/salida (kit LV-100).

## 2. Desarrollo

Las prácticas de instrumentación virtual fueron elaboradas para poner en marcha y probar la funcionalidad del kit LV-100. Este consta de una tarjeta que se divide en varias áreas de acuerdo al tipo de práctica que se vaya a realizar y al medio de comunicación empleado.

### 2.1. Descripción de las áreas del kit LV-100

Las áreas del kit pueden visualizarse en la figura 1.1. y cada una corresponde a alguna práctica en específico de acuerdo a lo siguiente:

*Tabla 1.1. Áreas del kit LV-100.*

| Área | Descripción |
|---|---|
| 1 | Alimentación |
| 2 | Conexión del puerto paralelo. |
| 3 | Prácticas para el puerto serial. |
| 4 | Prácticas para el puerto paralelo. |
| 5 | Puerto paralelo (primera área del 8255). |
| 6 | Puerto paralelo (segunda área del 8255). |

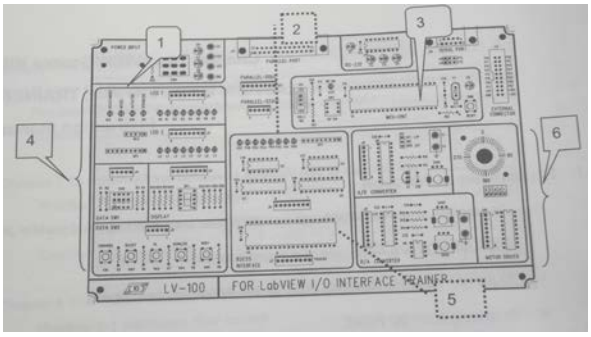



*Figura. 1.1. Plano del Kit LV-100.*

### 2.2. Diseño e implementación de una fuente de alimentación para el kit LV-100

Antes de iniciar con las prácticas fue necesario armar una fuente de voltaje simétrica para la alimentación del kit, cuyas salidas fijas son de +12, -12 y 5 V.

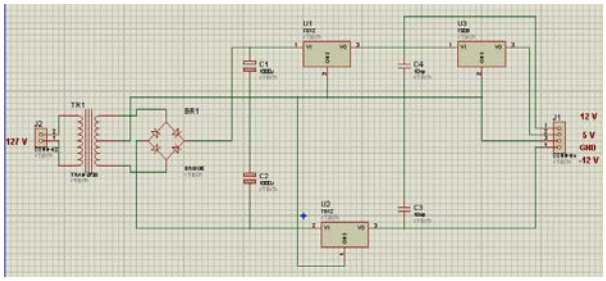



*Fig. 1.2 a. Diseño de la fuente de alimentación.*

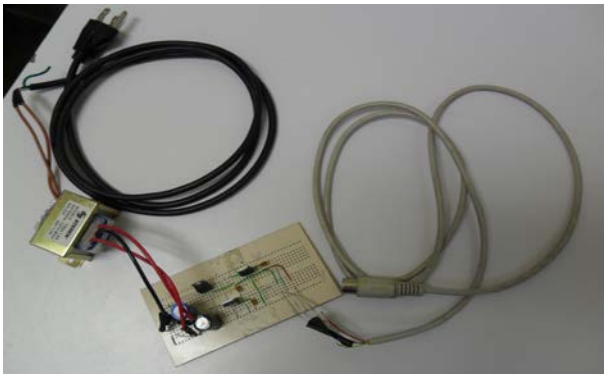


*Fig. 12 b3. Fuente de alimentación concluida.*

## 2.3. Uso de LabVIEW

Los programas creados en LabVIEW son llamados instrumentos virtuales o Vis, los cuales constan de tres elementos: panel frontal, diagrama de bloques, así como un icono y un conector.

El panel frontal se construye mediante controles e indicadores que contienen la interfaz gráfica de las aplicaciones. Por su parte, el diagrama de bloques es aquel que contiene el código fuente gráfico que controla el flujo de las aplicaciones.

### 2.3.1. Equipo empleado

- Computadora de escritorio.
- Kit LV-100.
- Cables serial y paralelo.
- Fuente de voltaje.
- Cámara de video.

## 2.4. Descripción detallada de las áreas del kit

A continuación se describen las prácticas realizadas por áreas y el medio de comunicación usado según sea el caso.

### 2.4.1. Área 4 del kit LV-100. (Puerto paralelo)

El primer bloque de prácticas corresponde al uso del puerto paralelo; la primera de ellas es verificar si este funciona adecuadamente y se revisa en las propiedades del equipo, una vez configurado correctamente se procede a usarse en las prácticas de transmisión de datos con elementos externos.

Como segunda práctica se hizo encender y apagar un LED este se visualiza en el panel frontal y a su vez físicamente en el kit. Dentro de la programación se usaron iconos de comunicación I/O (Input output). En las siguientes imágenes de muestra la funcionalidad de una de las prácticas de ésta área.

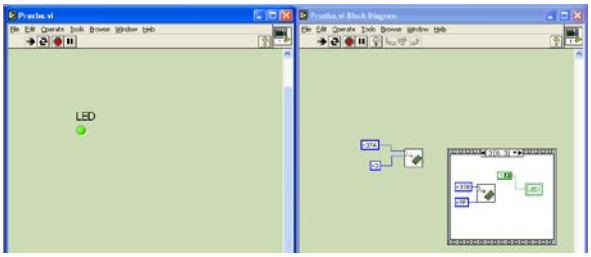



*Figura. 1.3a. Práctica 2.2. (Vista en LabVIEW).*

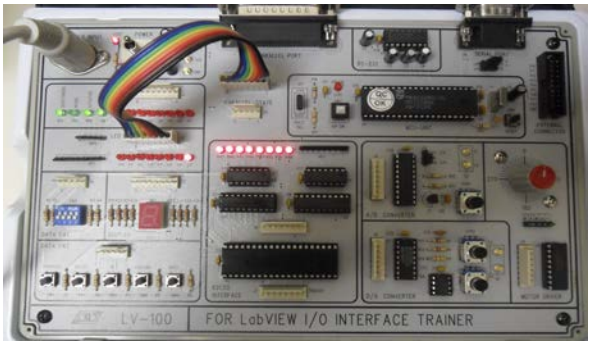



*Figura. 1.3b. Práctica 2.2. (Vista en el kit LV-100).*

#### 2.4.1.1. Método de salida para el puerto paralelo

En esta parte se desarrollan aplicaciones con una serie de LEDs, mismos que se manipulan de diversas formas como son el encendido y apagado de forma secuencial además del control mediante dispositivos de entrada como push buttons. Dentro de la programación se usan estructuras conocidas como el ciclo for y el bucle while, que permiten una mejor manipulación del arreglo de LEDs. En la figura 1.4a se aprecia la imagen del panel frontal y en la figura 1.4b la imagen del kit en funcionamiento.

*Tabla 1.2. Configuración de las salidas.*

| MSB LSB | 8 bit LEDs output | HEX |
|---|---|---|
| ●○○○○○○○ | 011111111 | 0x7F |
| ○●○○○○○○ | 101111111 | 0xBF |
| ○○●○○○○○ | 110111111 | 0xDF |
| ○○○●○○○○ | 111011111 | 0xEF |
| ○○○○●○○○ | 111101111 | 0xF7 |
| ○○○○○●○○ | 111110111 | 0xFB |
| ○○○○○○●○ | 111111011 | 0xFD |
| ○○○○○○○● | 111111101 | 0xFE |

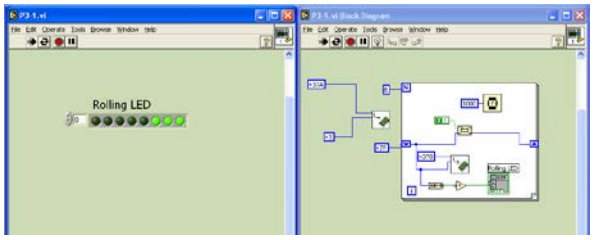



*Figura 1.4 a. Esquema de programación.*

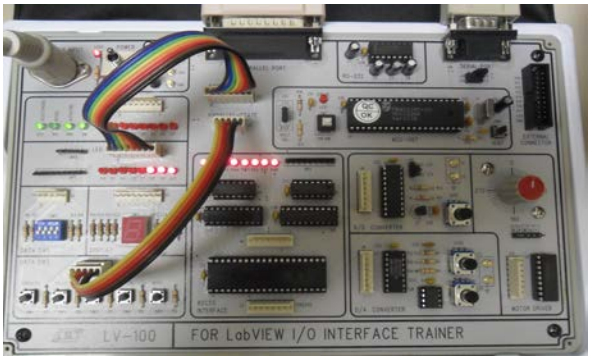



*Fig. 1.4b. Funcionamiento físico del programa.*

## 2.4.1.2. Método de entrada del puerto paralelo

En este apartado se utilizaran los elementos de entrada para modificar el control del arreglo de LEDs en diversas maneras, como puede ser pausar, iniciar o limpiar. También se usan dichos controles para controlar un cronometro que se visualiza en el panel frontal. Véase figura 1.5a. y figura 1.5b.

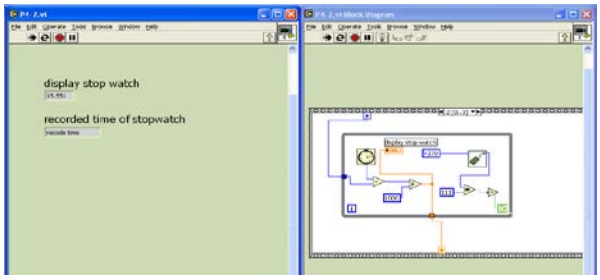



*Figura 1.5a. Control del cronómetro. (Vista del panel frontal).*

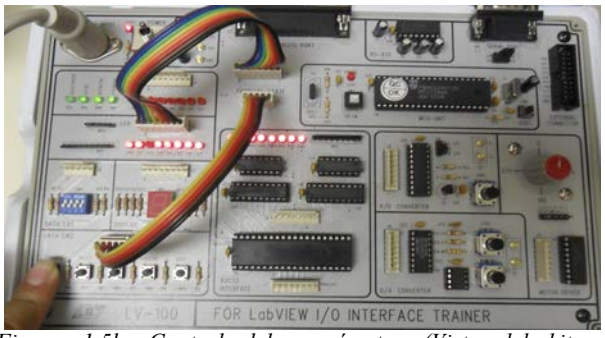



*Figura 1.5b. Control del cronómetro (Vista del kit en funcionamiento)*

## 2.4.1.3. Entradas y salidas del puerto paralelo

Ahora se trabajará sobre la misma área pero esta vez se controlará un display, los datos que en este caso se verán como números son grabados en el programa en formato hexadecimal y se usarán botones como entrada para manipular el display. Para una mejor comprensión vea la figura 1.5. En el primer programa el display muestra una serie de números en forma aleatoria, esta acción puede ser detenida mediante un botón de paro ubicado en el kit. Para el segundo ejemplo se representan los números hexadecimales desde 0 hasta F usando un switch, con el cual cada combinación genera una salida de ocho bits que ya estaba configurada dentro del programa. Las combinaciones son las mostradas en la tabla 1.3 y el diagrama queda como en la figura 1.6a. y 1.6b.

*Tabla 2.1. Configuración de salida para el display.*

| Switch de 4 entradas B3B2B1B0 | Salida de 8 bits para el display. A7A6A5A4A3A2A1A0 | No. Hexadecimal. |
|---|---|---|
| **0 0 0 0** | 1 1 0 0 0 0 0 0 | C0 |
| **0 0 0 1** | 1 1 1 1 1 0 0 1 | F9 |
| **:** | : | : |
| **:** | : | : |
| **1 1 1 1** | 1 0 0 0 1 1 1 0 | 8E |

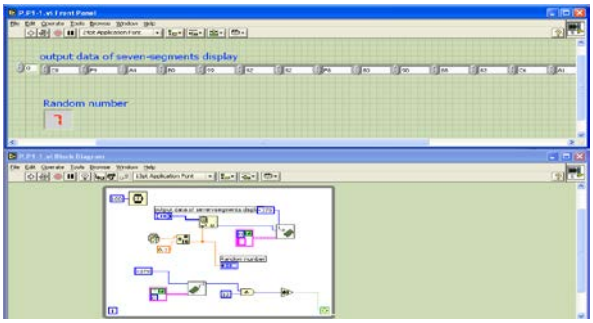



*Fig. 1.6a. Números aleatorios en el display (Vista en PC).*

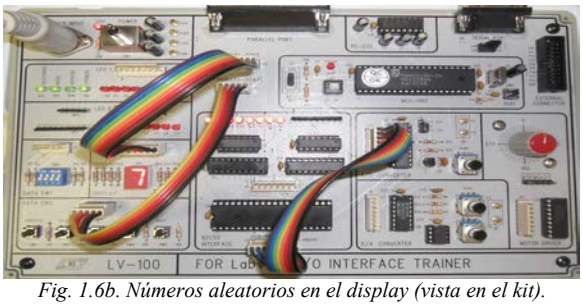



*Fig. 1.6b. Números aleatorios en el display (vista en el kit).*

## 2.4.2. Área 3, puerto serial

### 2.4.2.1. Introducción al puerto serie

La primera práctica para el puerto serial es probar si el puerto serial está configurado correctamente. Ello se hace conectando el icono de la transmisión serial con un indicador para que nos diga si existe algún problema con el puerto. Si no hay complicaciones, el indicador que en este caso es de error, será de 0.

Con la comunicación serial es posible grabar programas que se pueden descargar directamente del software WinISP dentro del microcontrolador 8051 y así ver las distintas aplicaciones que puede tener.

## 2.4.3. Área 5, integración del LV-100 con 8255 PPI

### 2.4.3.1. Introducción al chip 8051

En esta parte se usa el componente 8255 de control programable para expandir los puertos de I/O, el componente cuenta con tres puertos A, B, y C que tienen función Input/Output. Para reducir la complejidad de los programas se crearon dos SubVIs: uno de entrada y uno de salida para poder usarse como interfaz periférica programable con el componente 8255, así esta es una de las ventajas de usar LabVIEW, puesto que se pueden crear programas y usarse dentro de otros como Subprogramas que pueden modificarse en apariencia y también sus conexiones. Las figuras 1.7a. y 1.7b muestran los diagramas de los subprogramas creados.

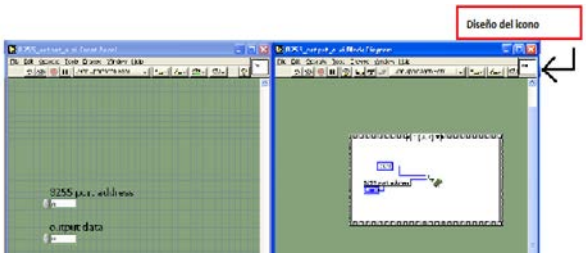



*Fig. 1.7a. Diseño de un SubVI de salida.*

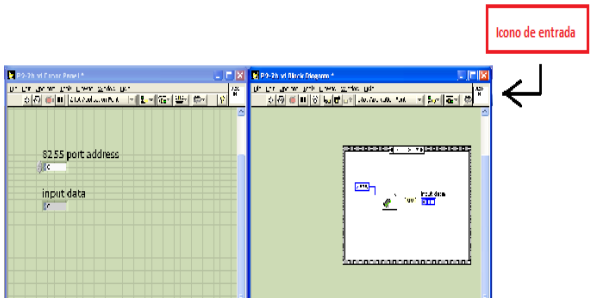



*Fig. 1.7b. Icono de entrada como SubVI.*

Para indicar si se quiere escribir o leer con alguno de los puertos se tiene que señalar dentro de los bits de control, para ello véase la siguiente tabla.

*Tabla 2.3. Control de operaciones para el componente 8255.*

| A1 | A0 | /RD | /WR | /CS | Función |
|---|---|---|---|---|---|
| **0** | 0 | 0 | 1 | 0 | Lee puerto A. |
| **0** | 1 | 0 | 1 | 0 | Lee puerto B. |
| **1** | 0 | 0 | 1 | 0 | Lee puerto C. |
| **0** | 0 | 1 | 0 | 0 | Envía datos al puerto A. |
| **0** | 1 | 1 | 0 | 0 | Envía datos al puerto B. |
| **1** | 0 | 1 | 0 | 0 | Envía datos al puerto C. |

Como primer ejemplo y para probar la funcionalidad de los iconos creados se hizo un ejercicio práctico con un arreglo de LEDs similar a los realizados anteriormente pero ahora con el 8255.

### 2.4.4. Área 5 y 6

### 2.4.4.1. 8255 Aplicado como un ADC

Continuando con el uso de los iconos de I/O creados, enseguida se describe un medidor de voltaje. Este voltímetro puede medir el voltaje interno o externo; por su parte, interno de un POT incluido en el área de trabajo y para el externo tiene dos terminales donde se pueden conectar otros dispositivos para medir su voltaje. Una representación del medidor de voltaje se ve en la figura 1.8a. y 1.8b.

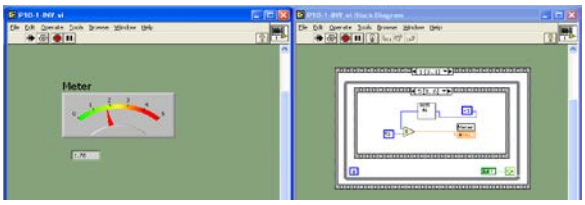



*Fig. 1.8a. Medidor de voltaje.(visualización en LabVIEW).*

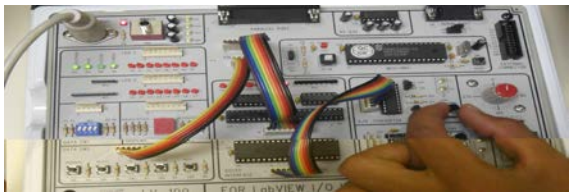


*Fig. 1.8b. Control del medidor de voltaje en el kit.*

Esta área se puede aprovechar para realizar mediciones de otras magnitudes físicas, con distintos transductores; cabe aclarar que esto va a depender de la magnitud a medir.

*Tabla 2.4. Ejemplos de transductores comunes.*

| Magnitud | Componente |
|---|---|
| **Posición** | Potenciómetro |
| | Encoders |
| **Temperatura** | Termopar |
| | Resistencia NTC |
| **Fuerza y par** | Galgas |
| **Velocidad** | Tacómetro |
| | Encoders |

LabView cuenta con indicadores de nivel para representar visualmente las magnitudes examinadas.

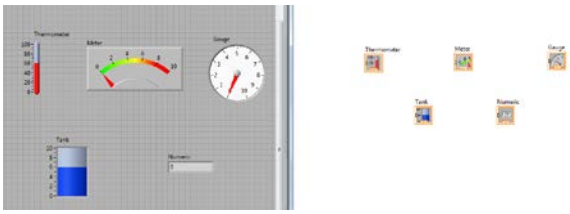


*Fig. 1.9. Indicadores de magnitudes fisicas.*

En ocasiones es necesario hacer acondicionamiento de señales para tener una buena representación de los valores. Como ejemplo Suponga que tiene un termopar que genera 10 mV/°C, aquí se tendría que hacer un arreglo de tal forma que se visualice ese grado en el indicador y no 10 mV. Esto se logra multiplicando por una constante de 100 en este caso y poner un indicador con un valor maximo de:

$$10mV \longrightarrow 1\ °C$$
$$5\ V \longrightarrow 500°C$$

### 2.4.4.2. Control de un motor paso a paso

Aquí se realiza el control de un motor paso a paso contenido dentro del kit, aunque se puede realizar el control de un motor externo conectado en las terminales incluidas en la misma área de trabajo. Los motores paso a paso tienen 4 fases y generalmente cada paso del motor es de 1.8 grados, por lo que se necesitarán 200 pulsos para completar una revolución. El motor incluido en el kit tiene un paso de 7.5 grados por pulso, esto quiere decir que se necesitan 48 pulsos para completar un giro. Hay diversas formas de hacer girar un motor paso a paso y estas son: de 1 fase, 2 fases y una combinación de las dos anteriores que es 1-2 fases. Este último es el método empleado en la práctica para poner en marcha el motor, ya que con ello se tiene un paso más pequeño y preciso, esta forma se describe mejor en la tabla siguiente.

*Tabla 2.5. Modo de operación del motor.*

| STEP | A | B | A′ | B′ |
|---|---|---|---|---|
| **1** | 1 | 0 | 0 | 0 |
| **2** | 1 | 1 | 0 | 0 |
| **3** | 0 | 1 | 0 | 0 |
| **4** | 0 | 1 | 1 | 0 |
| **5** | 0 | 0 | 1 | 0 |
| **6** | 0 | 0 | 1 | 1 |
| **7** | 0 | 0 | 0 | 1 |
| **8** | 1 | 0 | 0 | 1 |

Para la realización del programa se usaron los iconos creados de I/O con el 8255, y para indicar el orden de activación de las fases tal como se describieron en la tabla anterior se usa una serie de combinaciones en hexadecimal las cuales se pueden apreciar en la figura 1.10. El control del motor implica invertir el giro y variar su velocidad, para cambiar la velocidad se modifica la frecuencia de envío de datos y para invertir el giro se envían los valores en orden inverso.

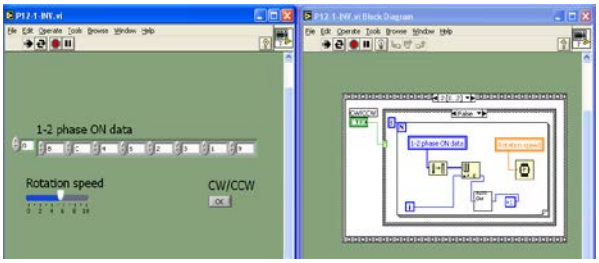



*Fig. 1.10. Control del motor paso a paso.*

## 2.3. Cámara de video

Como práctica final se hizo una pequeña aplicación con una cámara de video usando LabVIEW que consistía en realizar una grabación con un límite de tiempo predeterminado. El entorno de programación cuenta con iconos que permiten usar una cámara externa y tomar video, además se pueden hacer modificaciones en las imágenes o tomar simplemente ciertas características. En la figura 2.1a y 2.1b. se ilustra el programa realizado para tomar video y a su vez salvarlo en una unidad de grabación.

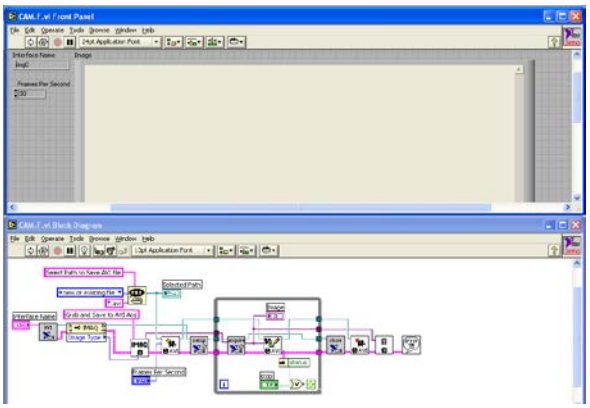


*Fig. 2.1a. Diseño del programa para la cámara de video.*

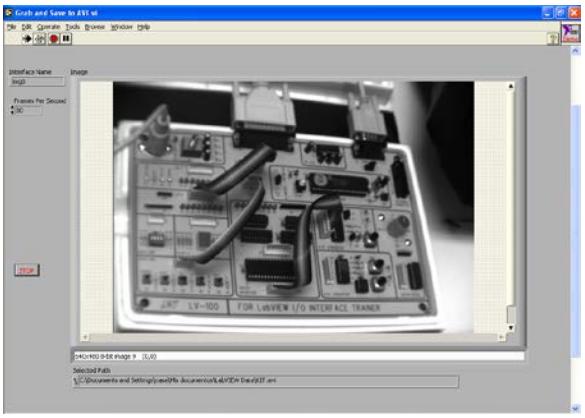


*Fig. 2.1b. Imagen tomada con la captura de video en curso.*

# 3. Conclusiones

El desarrollo de estas prácticas fue para probar que el kit LV-100 funcionara correctamente; para operarlo se necesitó armar una fuente de voltaje con las características de salidas necesarias para alimentar el kit. Se comprobó la función de transmisión y recepción de datos con LabVIEW a través de los puertos de comunicación y también se probaron otro componente muy importante como lo es una cámara de video.

La idea principal era que a todas las practicas realizadas pudieran ser de aplicación industrial, como controlar un proceso de selección de algún fruto usando tanto la cámara de video como una tarjeta de adquisición de datos que controle una banda y un actuador para que realice la selección del producto. Esta es una de tantas aplicaciones que podrían hacerse con los componentes y el software de programación LabVIEW.

Con el desarrollo de este proyecto aprendió más sobre el alcance de estos trabajos en una industria o en cualquier lugar donde se requiera algún proceso de control y se quedó con una idea más clara sobre cómo realizarlas aplicando lo aprendido.

Así entonces, con el uso de LabVIEW como entorno gráfico de programación para las etapas de control fue posible la comunicación externa para enviar y recibir datos a través de los puertos serial y paralelo que se enlazan con el kit LV-100 y; además con el uso de otros dispositivos externos como lo fueron la tarjeta de adquisición de datos y una cámara para la adquisición de video se obtuvo como resultado el procesamiento de imágenes. Por lo anterior, gracias al empleo de este tipo de entornos de programación se ahorra tiempo, es mucho más sencillo y fácil de realizar aplicaciones comparado con la programación de tipo textual, y otros métodos tradicionales.

## 4. Referencias

[1] Kandh. (s.f.). LV-100 for I/O LabVIEW Interface Trainer Experiment Manual. Taiwan.
[2] Lázaro, A. M. (2001). LabVIEW 6i. Madrid, España: Paraninfo.
[3] National Instruments. (2003). LabVIEW 7 Express. Austin, Texas, USA: National Instruments.
[4] National Instruments. (2003). LabVIEW 7 Express (Measurements Manual). Austin, Texas, USA: National Instruments.

[1] National Instruments. (2003). *LabVIEW 7 Express (User Manual)*. Austin, Texas, USA: National Instruments.

[2] National Instruments. (s.f.). *National Instruments - Pruebas y Medidas*. Recuperado el 8 de Agosto de 2011, de http://www.ni.com/es/

## 5. Biografía

Ilda Fabiola Angulo Molina, originaria de Badiraguato, Sinaloa; al momento se encuentra estudiando el 5° semestre de la licenciatura en informática en la Universidad Autónoma de Nayarit

Hugo Torres Ruiz, originario del estado de Michoacán, actualmente se encuentra realizando sus estudios de Ingeniería Electrónica en el Tecnológico de Zamora en el 7° semestre.

## *Agradecimientos*

Por su apreciable apoyo como asesores de este proyecto y el material de apoyo proporcionado, se agradece al M.C. Ricardo Fuentes Covarrubias y al M.C. Andrés Gerardo Fuentes Covarrubias.
Así mismo, se hace extensivo el agradecimiento a quienes guiaron el curso de "Introducción a la investigación" la M.C. Mónica Talla Sierra Peón y al M.C. Carlos Cedillo Nakay.
Agradecemos también a las instituciones de las cuales pertenecemos por habernos brindado la oportunidad de participar en este proyecto tan enriquecedor para nuestro aprendizaje científico: a la Universidad Autónoma de Nayarit y al Instituto Tecnológico de Estudios Superiores de Zamora.

# DISEÑO E IMPLEMENTACIÓN DE UN CONTROLADOR DE RUIDO ACTIVO MEDIANTE UN ALGORITMO DE LAZO ABIERTO

[1]Elizondo Santos Christian Issachar , [2]Vicente de los Santos Oscar
[3]Cedillo Nakay Carlos, [4]Sierra Peón Mónica Violeta , [5]Fuentes Covarrubias Andrés Gerardo
[6]Evangelista Salazar Martha Elizabeth

[1 2]Universidad de Colima, Facultad de Ingeniería Mecánica y Eléctrica, Km. 9 Carr. Colima-Coquimatlán, Colima, México.
talis271@hotmail.com mtvsierra@yahoo.com.mx
carlosn@ucol.mx monike@ucol.mx fuentesg@ucol.mx maevar@hotmail.com

## 1. ABSTRACT

The active control of noise, that is to say, the extenuation of acoustic noise by means of the pressure waves superimposition in contra-fase, went back to the years 30 when Paul Lueg, in Germany in 1936 and Coanda in France they patented the plans in which the active control is supported.

Currently speak of contamination by environmental noise is somewhat routine. The development of materials of construction insulating al environmental noise has proposed good solutions, nevertheless, imposes the one that a construction be customized at client to implement a solution since the phase of planning. In the case of a construction, be for house-room, industry or offices that already is found in use the setting to reduce the environmental noise by means of materials of construction requires modifications to walls and coverings with special insulating materials

The environmental noise extenuation technologies by means of techniques of digital prosecution of signs propose a by means of electronic solution that does not require modification of housing infrastructure, on the other hand, they propose an also applicable and portable solution to the automobile industry, of aviation and other environments.

At present the technological development and the compression in the integrated circuits integration scale has permitted that the Digital Processors of Signs (DSP) are applied to the control algorithms implementation, filters, controllers and actuators; and with it to achieve that this technology emerge.

## 2. Introducción

El presente proyecto trata sobre la implementación de los algoritmos LMS para la cancelación de ruido ambiental. El algoritmo propuesto filtered-X/U LMS en tiempo real se propone implementar en la tarjeta de procesado de señal de bajo coste: DSK TMS320C26 o C50 de Texas Instruments Se efectuará un análisis de los recursos consumidos por el

procesador (memoria y tiempo de proceso) en la implementación de los algoritmos para la elección de la tarjeta adecuada.

Dado que las aplicaciones actuales se enfocan resolver problemas de la industria automotriz y de aviación, este proyecto estará enfocado a solucionar dicho problema mediante la implementación habitacional y de auditorios

## 3. Material y Métodos

Conversión de señales de audio del dominio analógico al discreto

Una señal se define como una cantidad física que varia con el tiempo, el espacio o cualquier otra variable o variables dependientes. Matemáticamente, describimos una señal como una función de una o más variables independientes. Por ejemplo, la función:

$$s(t) = A\cos(2\pi f_0 t - \theta) \quad (1)$$

Describe una señal senoidal que varia en función de la variable independiente t (tiempo), a esta señal se le conoce como "de tiempo continuo" y define matemáticamente una señal analógica. Con frecuencia una señal de tiempo discreto se obtiene tomando muestras de una señal de tiempo continuo; en el espectro de señales audibles una senoidal de frecuencia constante o a frecuencia fundamental; o si es una señal compuesta como lo son en el mundo real, una suma de señales senoidales.

$$\sum_{i=1}^{N} A_i(t)sen[2\pi F_i(t)t + \theta_i(t) \quad (2)$$

Tanto en (1) como en (2), A representa la amplitud de la señal, f0 es la frecuencia en Hz y θ es la fase inicial. Si se obtiene una señal de tiempo discreto tomando regularmente muestras de (1) o (2) a la frecuencia de muestreo fs=1/T, conseguiremos transformar (1) del dominio analógico al discreto o discretizar la señal, (1) se transforma en:

$$s[n] = s(t)\Big|_{t=nT} = A\cos(2\pi f_0 Tn + \theta) \quad (3)$$

Donde el termino s[n] se refiere a una representación discreta de una muestra de s(t). Para el caso de (3) {Ai(t)}, {Fi(t)}, y {θi(t)} son los conjuntos de amplitudes, frecuencias y fases de la familia de senoides presentes en una señal de audio compuesta.

Dado que se efectúa el control mediante un controlador discreto en necesario efectuar el cambio de dominio en tiempo real.

## 4. Método de Cancelación de Ruido Ambiental

Por cancelación activa de ruido ambiental se entiende aquella técnica capaz de atenuar el ruido acústico, existente en un determinado recinto, mediante la emisión de otro ruido acústico. Esto es posible cuando las ondas de presión acústica de ambos ruidos estén en oposición de fase (interferencia destructiva).

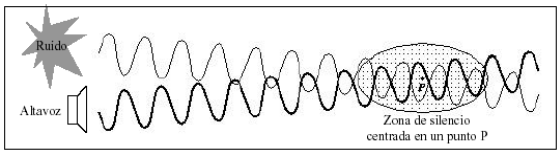



Figura 1. Método de cancelación de ruido ambiental por suma de señal en contrafase

La técnica de control a emplear permitirá situar estas "zonas de silencio", mas o menos numerosas y de mayor o menor tamaño, en aquellos lugares del recinto que se desee. Los elementos que constituyen todo el conjunto controlador son los siguientes: un sensor (micrófono) de referencia acústica, un controlador que varíe las características de la señal referencia, un actuador (altavoz) que genere ondas de presión y un sensor de realimentación (micrófono) que sirva al controlador para modificar su funcionamiento y conseguir disminuciones de presión acústica justamente donde esté colocado este sensor.

Cuando se desear atenuar o eliminar el ruido que genera una fuente acústica lo mas inmediato es actuar allí donde se encuentra dicha fuente, y en caso de no ser posible, actuar entonces sobre el lugar donde se requiere una disminución de ruido. En ambos casos, es necesaria una barrera que impida el paso de la energía acústica desde la fuente al recinto de interés.

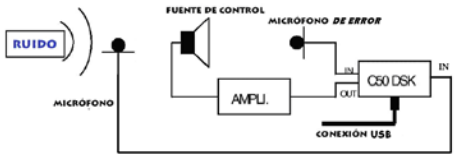



Figura 2. Cancelador Adaptivo Activo de Ruido de un sólo Canal

.

Las líneas que representan el ángulo (Azimut) se trazan cada 10 grados partiendo del origen de los círculos antes mencionados hasta la circunferencia del círculo mayor, para trazar una línea en java se utiliza el método drawLine() al cual se le envían 4 parámetros que son las coordenadas de dos puntos (el origen de los círculos y el punto sobre la circunferencia del circulo mayor con respecto del ángulo) el problema es que se desconocen las coordenadas del segundo punto, para ello se creó el método SetPunto() que calcula ese punto, usando el ángulo y la distancia que existe desde el origen hasta la circunferencia del circulo mayor, en el Cuadro 2, se observa la implementación del método SetPunto().

## 5. Programa Implementado

```
#include "tonecfg.h"
#include "dsk5510.h"
#include "dsk5510_aic23.h"

#define SINE_TABLE_SIZE   48

DSK5510_AIC23_Config config = { \
   0x0017,
   0x0017,
   0x01f9,
   0x01f9,
   0x0011,
   0x0000,
   0x0000,
   0x0043,
   0x0081,
   0x0001
};
```

```
Int16 sinetable[SINE_TABLE_SIZE] = {
   0x0000, 0x10b4, 0x2120, 0x30fb, 0x3fff,
0x4dea, 0x5a81, 0x658b,
   0x6ed8, 0x763f, 0x7ba1, 0x7ee5, 0x7ffd,
0x7ee5, 0x7ba1, 0x76ef,
   0x6ed8, 0x658b, 0x5a81, 0x4dea, 0x3fff,
0x30fb, 0x2120, 0x10b4,
   0x0000, 0xef4c, 0xdee0, 0xcf06, 0xc002,
0xb216, 0xa57f, 0x9a75,
   0x9128, 0x89c1, 0x845f, 0x811b, 0x8002,
0x811b, 0x845f, 0x89c1,
   0x9128, 0x9a76, 0xa57f, 0xb216, 0xc002,
0xcf06, 0xdee0, 0xef4c
};

void main()
{
   DSK5510_AIC23_CodecHandle hCodec;
   Int16 msec, sample;

  DSK5510_init();

   hCodec = DSK5510_AIC23_openCodec(0,
&config);

   for (msec = 0; msec < 5000; msec++)
   {
      for (sample = 0; sample <
SINE_TABLE_SIZE; sample++)
      {
         while
(!DSK5510_AIC23_write16(hCodec,
sinetable[sample]));

         while
(!DSK5510_AIC23_write16(hCodec,
sinetable[sample]));
      }
   }
   DSK5510_AIC23_closeCodec(hCodec);
}
```

## 6. Conclusión

El desarrollo de un sistema de control activo de ruido acústico esta limitado por el fenómeno acústico, los transductores electro-acústicos y el controlador electrónico el cual se implementara por sus características. Según se ha expuesto anteriormente, la atenuación acústica por técnicas activas sólo es aplicable, a efectos prácticos, en bajas frecuencias hasta los 500 Hz, complementando el control menos eficiente que ofrecen las técnicas pasivas. Además, el control se realiza en zonas de pequeñas dimensiones o en conductos por donde se propagan ondas acústicas planas.

Así como también la selección y ubicación de los transductores electro-acústicos determina el buen funcionamiento del sistema, para esto, debe realizarse un análisis previo del mapa de ruido para determinar las líneas y superficies nodales de la presión acústica de ruido y situarlos lejos de estas posiciones. Este factor no puede pasarse por alto y tiene que tener un gran peso en el diseño final del sistema. En algunos casos, una determinada ubicación de los transductores puede llegar a suponer, no sólo la incapacidad de atenuar el ruido acústico, sino incluso el incremento del ruido acústico inicial. La posición óptima de los transductores no es una cuestión obvia.

Además de la colocación de los transductores también tenemos que tener en cuenta el desarrollo de los parámetros para el diseño del software, esto parámetros los podemos calcular

con la ayuda del MatLab para simplificar en gran medida el desarrollo del software, los módulos M generados son utilizados en los algoritmos adaptativos de control.

Otra cosa que se tomo en cuenta para el desarrollo de esta investigación es conocer que tipo escenario tenemos para poder lograr una cancelación de ruido mucho mas efectiva.

El uso de la tarjeta DSK TMS320VC5510 de Texas Instruments, nos facilitó en gran medida el diseño de este trabajo, ya que el software que viene incluido en el kit, cuenta con una amplia gama de librerías y software ejecutivo.

El desarrollo de esta tecnología si se continúa su investigación podría beneficiar a muchas personas, ya que el ruido es una de las causas de estrés en las personas y sobre todo de las que viven en la ciudad. Solo que en estos momentos este tipo de tecnologías tiene costos muy elevados y no se pueden adquirir muy fácilmente, además que se ven afectadas por la amplia gama de frecuencias de ruido que existen, ya que este tipo de sistema solo es efectivo para ruidos de baja frecuencia y sus áreas de cancelación son muy pequeñas.

**Bibliografía**


(Baumann 1992) In: Baumann D.C. and Greiner R.A., "Number of error microphones for multi-modal cancellation", Proceedings of Inter-Noise 92, Tornto, Canada, July 20-22, 1992, pp. 345-348.

(Lueg 1936) In: Lueg P., "Process of silencing sound oscillations," U.S. Patent 2,043,416, June 19, 1936.

(Bullmore 1987) In: Bullmore A. J., Nelson P. A., Elliott S. J., Evers J. F., Chidley B., "Models for evaluating the performance of propeller aircraft active noise control systems", AIAA, Aeroacoustics Conference, 11th, Sunnyvale, CA, Oct. 19-21, 1987.

(Coanda 1932) En: Coanda, French Patent 722.274, 1932.

(Conover 1957) En: Conover W.B. and Bray F.M., U.S. Patent 2,776,020, Jan. 1, 1957.

(Costin 1989) In: Costin M.H. and Elzinga D.R., "Active reduction of low-frequency tire impact noise using digital feedback control", IEEE Control System Magazine, pp. 3-6, August 1989.

(Elliott 1988) In: Elliott S.J., Syothers I.M., Nelson P.A., McDonald A.M., Quinn D.C., and Saunders T., "The Active Control of Engine Noise Inside Cars", Proc. of Inter-Noise 88, pp. 987-990, 1988.

(Elliott 1993) In: Elliot S.J. and Nelson P.A., "Active noise control," IEEE Signal Processing Magazine, pp. 12-35, Octuber 1993.

(Guicking 1985) In: Guicking D., Karcher K. and Rollwage M., "Coherent Active Methods for Applications on Room Acoustics", J. Acoustic. Soc. Am., 87, pp. 2251-2254, 1990.

(Joseph 1988) In: Joseph P., Elliott S.J., and Nelson P.A., "Active Cancellation at a Point in a Pure Tone Diffuse Sound Field, Zones of Silence and Potencial Energy Statitics", Proc. of Inter-Noise 88, pp. 971-974, 1988.

# CLASIFICADOR DE PIEZAS POR TAMAÑO

[1]López Romero José María , [2]Nava Aguirre José Ignacio , [3]Torres Rangel Andrés,
[4]Vázquez Reina Jair, [5]Solorio Alejandro , [6] Fuentes Covarrubias Ricardo
[1, 2, 3, 4,5] *Instituto Tecnológico de Colima. Av. Tecnológico 1, Villa de Álvarez, Colima. México.*
[6]*Universidad de Colima, Facultad de Ingeniería Mecánica y Eléctrica, Km. 9 Carr. Colima-Coquimatlán, Colima, México.*
[1]*Colima15@blogspot.com,* [6]*fuentesr@ucol.mx*

**Resumen**

*Se realizó el diseño y construcción de un clasificador de materiales por tamaño, así como la implementación del control y la interfaz gráfica, la cual por medio de la ayuda de un sensor infrarrojo, identifica la altura de las piezas, posteriormente pasa la información a un microncontrolador el cual es el encargado de llevar a cabo todas las tareas para el completo funcionamiento del prototipo, entre ellas el posicionamiento correcto de los servomotores indicando en qué depósito corresponden.*

Palabras clave: Automatización, Control de Posición, Control Sistema Experto, Sensor Infrarrojo, Microcontrolador, Servomotor.

## 1. INTRODUCCIÓN

El proyecto está basado prácticamente en la implementación de varios elementos y su precisa sincronización ya que hablamos de un clasificador de materiales los cuales al momento de ser identificados serán depositados en distintos recipientes dependiendo su única característica que en este caso es su altura. Cabe mencionar que para su selección se utilizó un sensor infrarrojo y posteriormente para su depositado unos servomotores, cada

elemento siendo comandado por un microcontrolador, todo el sistema es monitoreado y controlado mediante una interfaz realizada en el paquete llamado Visual Studio C#.

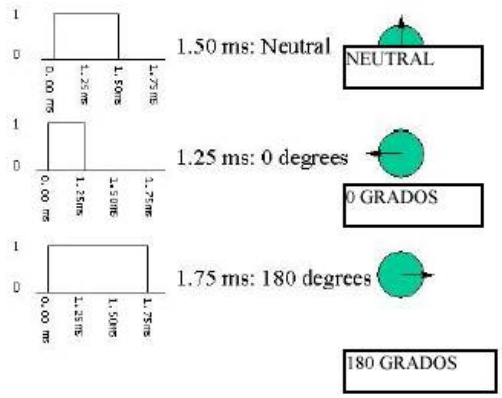
## 2. DESARROLLO

2.1. Marco Teórico

Sistema Experto(SE): es básicamente un programa de computadora basado en conocimientos y raciocinio que lleva a cabo tareas que generalmente sólo realiza un experto humano; es decir, es un programa que imita el comportamiento humano en el sentido de que utiliza la información que le es proporcionada para poder dar una opinión sobre un tema en especial.

Lógica de Servomotor: El motor del servo tiene algunos circuitos de control y un potenciómetro (una resistencia variable) esta es conectada al eje central del servo motor. Este potenciómetro permite a la circuitería de control, supervisar el ángulo actual del servo motor. Si el eje está en el ángulo correcto, entonces el motor está apagado. Si el circuito chequea que el ángulo no es el correcto, el motor girará en la dirección adecuada hasta llegar al ángulo correcto. El eje del servo es capaz de llegar alrededor de los 180 grados. Normalmente, en algunos llega a los 210 grados, pero varía según el fabricante.

Un servo normal se usa para controlar un movimiento angular de entre 0 y 180. La cantidad de voltaje aplicado al motor es proporcional a la distancia que éste necesita viajar. Así, si el eje necesita regresar una distancia grande, el motor regresará a toda velocidad. Si este necesita regresar sólo una pequeña cantidad, el motor correrá a una velocidad más lenta. A esto se le llama control proporcional. Como se observa en la figura, la duración del pulso indica o dictamina el ángulo del eje (mostrado como un círculo verde con flecha). Nótese que las ilustraciones y los tiempos reales dependen del fabricante de motor. El principio, sin embargo, es el mismo.



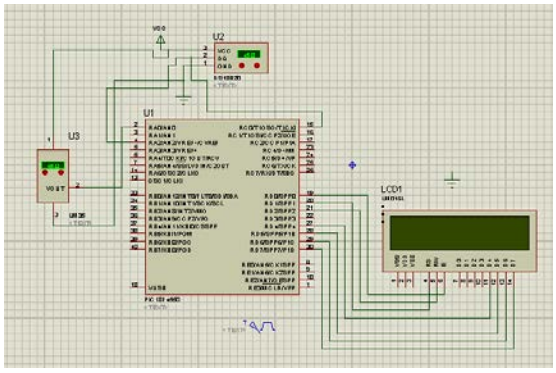



Figura 2. Diagrama de la Conexión USB

2.2. Cálculos

Debido a que estamos usando un sensor infrarrojo analógico y éste nos arroja una lectura expresada en voltaje de 0 a 5 volts nos vimos en la necesidad de usar el ADC del microcontrolador implementando la siguiente fórmula para obtener un resultado que nos fuera más sencillo manipular a la hora de programar.

----------Sensor Analógico----------
medicion =( Read_ADC() / 1024.0)*5000;//conversion

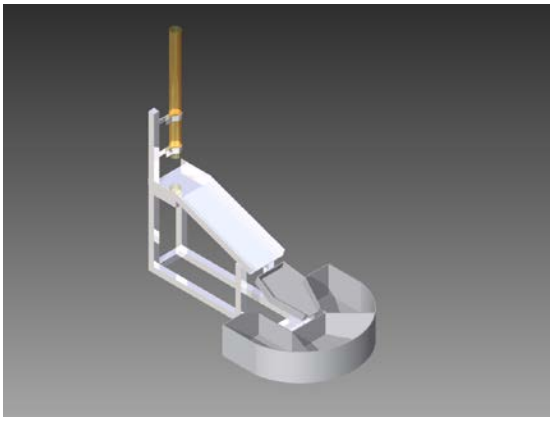


Figura 3. Prototipo Simulado en 3D

Se implementaron dos servomotores que como mencionamos anteriormente son controlados mediante un ancho de pulso. El primer servomotor requería una velocidad relativamente lenta en comparación al segundo, a continuación se muestra el ancho de pulso utilizados.

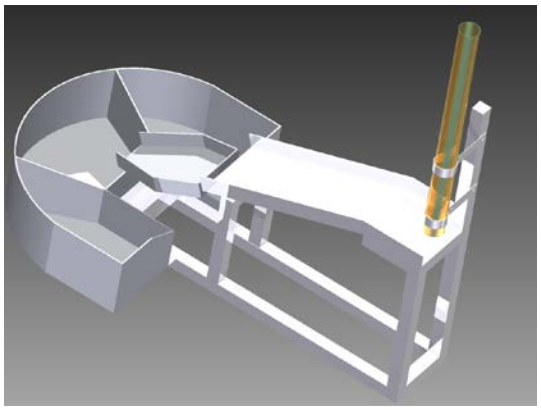


Figura 4. Prototipo Simulado en 3D(2daVista)

2.3. Diagramas Esquemáticos

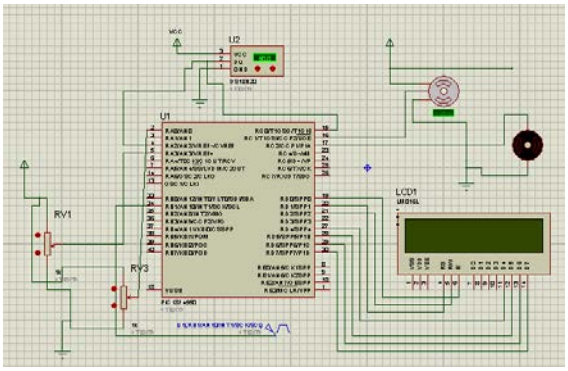


Figura 1. Diagrama Completo en Proteus

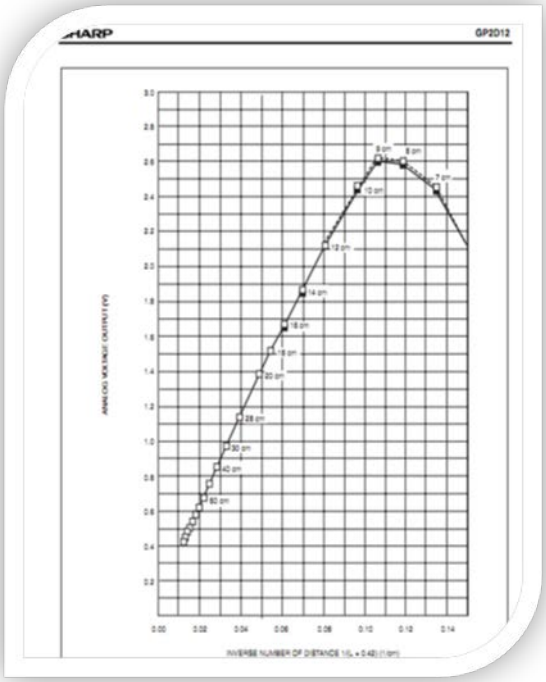
2.4. Dispositivos
El proyecto en general está constituido por varios dispositivos los cuales en conjunto dan lugar al buen funcionamiento de éste. A continuación una breve definición de cada uno de ellos:

- Servomotor: es un dispositivo pequeño que tiene un eje de rendimiento controlado. Este puede ser

llevado a posiciones angulares específicas al enviar una señal codificada. Con tal de que una señal codificada exista en la línea de entrada, el servo mantendrá la posición angular del engranaje. Cuando la señal codificada cambia, la posición angular de los piñones cambia. En la práctica, se usan servos para posicionar superficies de control como el movimiento de palancas, pequeños ascensores y timones. Ellos también se usan en radio control, títeres, y por supuesto, en robots.

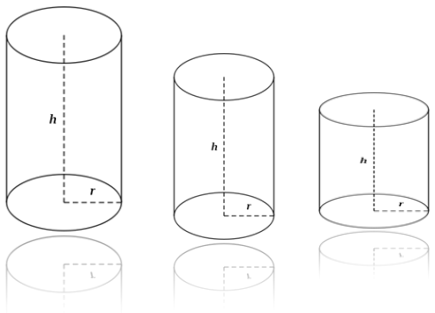
- Sensor Infrarrojo: El sensor es un dispositivo electrónico/mecánico/químico que mapea un atributo ambiental resultando una medida cuantificada, normalmente un nivel de tensión eléctrica. Particularmente, el sensor infrarrojo es un dispositivo electrónico capaz de medir la radiación electromagnética infrarroja de los cuerpos en su campo de visión. Todos los cuerpos reflejan una cierta cantidad de radiación, esta resulta invisible para nuestros ojos pero no para estos aparatos electrónicos, ya que se encuentran en el rango del espectro justo por debajo de la luz visible.
- Microcontrolador(PIC18F4550):
  Un microcontrolador (abreviado µC,UC o MCU) es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales, los cuales cumplen una tarea específica. Un microcontrolador incluye en su interior las tres unidades funcionales principales de una computadora: unidad central de procesamiento, memoria y periféricos de entrada y salida.
- LCD (Liquid Crystal Display): es una pantalla delgada y plana formada por un número de pixeles en color o monocromos colocados delante de una fuente de luz o reflectora. A menudo se utiliza en dispositivos electrónicos de pilas, ya que utiliza cantidades muy pequeñas de energía eléctrica.
- Fuente de Alimentación: es un dispositivo que convierte la tensión alterna de la red de suministro, en una o varias tensiones, prácticamente continuas, que alimentan los distintos circuitos del aparato electrónico al que se conecta.
- Visual Studio 2010: es un conjunto de herramientas completo que simplifica el desarrollo de aplicaciones para desarrolladores individuales o equipos, haciendo posible la creación de aplicaciones escalables y de alta calidad. Independientemente de si está escribiendo código, creando bases de datos, evaluando o depurando, podrá mejorar su productividad utilizando las potentes herramientas que trabajan como usted trabaja.

2.5. Gráficas




Gráfica característica que muestra la lectura entre la distancia de entrada y el voltaje de salida del sensor infrarrojo.

2.6. Datos




Las piezas utilizadas

y consideradas en la medición fueron tres cilindros de madera del mismo diámetro pero de distinto tamaño en altura (chica mediana y grande) utilizando 3, 4 y 6 cm consecutivamente. Las piezas fueron recubiertas por una capa de pintura blanca porque comprendimos que dicha pintura tiene un mayor reflejo óptico en la lectura del sensor infrarrojo.

## 3. RESULTADOS

Los resultados arrojados a la finalización de este proyecto fueron muy concretos y convincentes. Es sumamente importante tener un amplio conocimiento sobre el manejo de sensores en este caso infrarrojo ya que nos enfrentamos con diversos problemas al momento de realizar la sincronización de los servomotores junto con el sensado y a su vez el monitoreo mediante la interfaz realizada. Al momento de entrelazar lo mecánico, electrónico y la programación tuvimos que solucionar bastantes discrepancias con la ayuda de la pericia que se obtiene al combinar teoría y práctica. Cabe mencionar que a pesar de los inconvenientes que surgieron a través de la implementación de este prototipo obtuvimos un resultado bastante satisfactorio por la precisión en la cual las piezas fueron clasificadas.

## 4. LIMITACIONES

Al finalizar el proyecto nos encontramos con limitantes que debido a factores como tiempo, calidad y precio de materiales y falta del manejo de paquete programadores, provocaron varios inconvenientes al momento de estar realizando el prototipo:

- Descalibración debido al ruido de luz ambiental.
- Tamaño de la pieza.
- Sincronización y coordinación.
- Inversión.

## 5. REFERENCIAS

[1] http://www.sapiesman.com/controlmoderno/sistemaexpero.htm
[2] Ingenieria de Control Moderna, Ogata, Katsuhiko Editorial: Prentice Hall/Pearson
[3] Compilador CCS y simulador Proteus para Pic.
[4] Sistemas de Control Automático, Benjamin C. Kuo.

# ROBOT VOCACIONAL

[1]Ruiz Tadeo Ana Claudia, [2]Nava Huitrón Débora Fabiola, [3]Chávez Valdez Ramona Evelia, [4]Saenz Rodríguez María Isabel, [5]González Valladares Esteban.

[12345]*Instituto Tecnológico de Colima. Av. Tecnológico 1, Villa de Álvarez, Colima.*
[1]*aruiz@itcolima.edu.mx,* [2]*dnava@itcolima.edu.mx,* [3]*echavez@itcolima.edu.mx,*
[4]*isaenz@itcolima.edu.mx,* [5]*egonzalez@itcolima.edu.mx*

**Resumen:** En este artículo se presenta el funcionamiento de un Robot virtual que simula a un experto humano en Orientación Vocacional. Permite establecer una conversación con los usuarios para definir su vocación de acuerdo a sus capacidades e intereses personales. Con este proyecto se presenta un nuevo enfoque a este análisis vocacional ya que simula una conversación amigable entre ambos actores. La investigación se desarrolla dentro del área de la Inteligencia Artificial. La aportación principal es presentar el prototipo de un robot virtual capacitado para simular a un orientador vocacional que establece un dialogo con la intención de hacer creer que se está hablando con otra persona. El enfoque utilizado en nuestra investigación puede aplicarse a otras prácticas como la emulación de un psicólogo, pedagogo y otras aplicaciones posibles.

***Palabras Clave***: Inteligencia Artificial, Agente Inteligente, Sistema Conversacional, Orientación Vocacional.

**Abstract**: This research presents an application of Artificial Intelligence models and tools to develop an Intelligent Agent, which simulates a human expert on Career Counseling. This Agent establishes a conversation with users in order to define their career according to their personal skills and interests. This project will give a fresh approach to Career Analysis, by simulating a friendly conversation between the Intelligent Agent and the user. This research takes place in the field of Artificial Intelligence. A virtual Robot with the ability of "thinking" like a human being and the goal of making a person believe that he or she is talking with another person is the main contribution of this project. The approach used in our Artificial Intelligence research can be applied to other disciplines, like making a person believe he or she is talking with a Psychologist or a Pedagogue. There may be other possible applications as well.

***Keywords***: Artificial Intelligence, Intelligent Agent, Conversational System, Vocational Orientation.

## Introducción

Los Sistemas Inteligentes actualmente se aplican en el campo de la medicina, ingeniería, economía, la milicia y el entretenimiento. Con este proyecto se busca diseñar un agente inteligente en orientación vocacional para apoyar a los jóvenes a seleccionar la carrera de acuerdo sus características, intereses, aptitudes y necesidades personales. Un Agente Inteligente es un software que automatiza el proceso de razonamiento humano, en este documento denominaremos a este agente como robot o "*bot*" indistintamente.

El análisis de Orientación Vocacional se describe como un estudio que permite ayudar a los jóvenes a determinar las habilidades y capacidades que tienen para realizar actividades profesionales. Con la finalidad de que elijan la carrera adecuada a su personalidad, teniendo en cuenta las posibilidades trabajos existentes [1].

La elección de una carrera nos permite desempeñar una función profesional, así como una forma de vida, por tanto, la elección debe hacerse consciente de que con ella formamos nuestra identidad, a través de ella, asumimos un rol, un estatus y hasta elegimos una pareja [2].

El sistema que se propone está capacitado para poder pensar como un ser humano, un programa que pueda simular una conversación escrita, con la intención de hacer creer a un humano que está hablando con otra

persona. La conversación tendrá como finalidad determinar la orientación vocacional de los usuarios. Para su diseño utilizaremos el l enguaje de programación AIML (Lenguaje de Marcas para Inteligencia Artificial), que es una extensión de XML, este lenguaje es utilizado para crear chatbot o sistemas conversacionales.

La aplicación de los métodos y modelos de la inteligencia artificial nos permite desarrollar sistemas de utilidad a todos los sectores sociales como es el caso de nuestro chatbot que permite apoyar a l os jóvenes estudiantes a s eleccionar su carrera de acuerdo a s us preferencias, gustos, habilidades y deseos personales utilizando un agente inteligente conversacional desde la comodidad y privacidad de su casa o l ugar de trabajo. Estos sistemas inciden en una disminución de la deserción escolar en los distintos niveles académicos y coadyuvan a formar profesionistas útiles a l os diversos sectores económicos y sociales de nuestro entorno.

## Trabajos Relacionados

Turing (1950), en la revista *Mind* publica un artículo llamado *Computing Machinery and Intelligence*, donde propone la investigación desarrollado por Alan Turing (Test de Turing), para probar que las máquinas son inteligentes, actualmente son una de las prioridades para l os investigadores de Inteligencia Artificial. Se propone como hipótesis de que, si una máquina se comporta en todos los aspectos como inteligente, entonces se prueba que es inteligente. El desafío consiste en que la máquina debe hacerse pasar por humano en una conversación con una persona a través de una comunicación de texto estilo chat. A la persona n o se le avisa si está hablando con una máquina u otro ser humano. Si el sujeto es incapaz de determinar si la otra parte de la comunicación es humana o máquina, entonces se considera que la máquina ha alcanzado un determinado nivel de madurez entonces es inteligente. Todavía ninguna máquina puede pasar este examen en una experiencia con método científico [3].

A.L.I.C.E. es uno de los sistemas conversacionales más fáciles de utilizar para interactuar [4]. Su estructura está basada en d os partes importantes: el compilador del lenguaje y almacenamiento de datos en AIML [5].

El chatbot, **Eliza**, fue diseñado en 1966 por el Profesor **Joseph Weizenbaum** del Instituto de Tecnología *Massachusetts* [6] con la finalidad de analizar la comunicación con un idioma natural entre una persona y una computadora. Que simula a u n terapeuta al responder preguntas.

En el *Institute of Information Science*, Taiwan, se desarrollo un programa que emula una conversación humana y proporciona una opción para aprender el idioma inglés. E l chatbot inteligente pr oporciona un método de colaboración sincrónica. Desempeña el papel de un ayudante de instructor para prestar el servicio en cualquier momento y en cualquier lugar. Proporciona materiales de referencia, diccionario, responde a preguntas, el agente realiza registros y analiza la conversaciones para que el profesor pueda evaluar los progreso de los estudiantes.

Esta investigación tiene como objetivo desarrollar un Agente Inteligente Conversacional que simule a un experto humano en Orientación Vocacional, y será novedoso para los jóvenes estudiantes de secundaria, bachillerato y nivel superior y estaría a la vanguardia de lo que ellos utilizan actualmente, el chat.

En el contenido de este trabajo se plasma inicialmente los conceptos y teorías que soportan este proyecto, posteriormente se muestra e l modelo del sistema a desarrollar, para finalizar se presentan los resultados y conclusiones de esta investigación.

## Fundamentos Teóricos

En este apartado del artículo se presentan los conceptos teóricos que respaldan el desarrollo de un agente inteligente en orientación vocacional.

### Educación y Tecnología

Este trabajo se desarrolla en el campo educativo, por ello es importante comprender en primer lugar que la educación es un proceso a través del cual una persona adquiere algún tipo de habilidad o t oma conciencia de algo [7].

Se tendrá mayor i mpacto cuando se logre que la educación se pueda impartir desde las diferentes ramas de la informática, como es la inteligencia artificial puede ser mayor alcance cuando se incorporen procesos autónomos o inteligentes, para que los sistemas computacionales dejen de ser solo un administrador o presentador de recursos [7].

## Inteligencia Artificial

Esta investigación se encuentra dentro de la Inteligencia artificial por tanto una definición de inteligencia artificial por Minsky (1984) [14] es "el estudio de cómo programar computadoras que tengan la facultad de hacer aquello que la mente humana puede realizar". Este concepto resulta extenso ya que la inteligencia artificial es un amplio estudio de procesos realizados por los humanos, de tal modo que se distinguen dos grupos para los cuales se puede dictar una definición independiente en base a su naturaleza, estos grupos corresponden al procesamiento de la información y a la representación de la misma, siendo esta última la que se refiere a los sentidos y aspectos como el lenguaje y la comunicación [8]. La **inteligencia artificial** es la ciencia que desarrolla programas para máquinas que permitan la realización de tareas humanas y simulan la inteligencia humana.

La investigación en el campo de La Inteligencia Artificial se destaca principalmente por el desarrollo de máquinas que automatizan actividades que requieran un comportamiento inteligente.

## Agente Inteligente

Los agentes inteligentes son resultado del estudio e investigación hecha en el ámbito de la inteligencia artificial. Específicamente provienen de la inteligencia artificial distribuida (IAD) y de la vida artificial. Las primeras aplicaciones de la vida artificial aparecieron casi al mismo tiempo que la informática con los autómatas celulares de Von Neumann y las neuronas formales [5].

Dando paso desde entonces a una serie de evoluciones hasta llegar a las aplicaciones que ahora se están desarrollando tales como el Web Semántico, los asistentes virtuales, chatbot, entre otros. Un agente es una entidad de software con un propósito específico. La persistencia distingue a los agentes de las subrutinas; los agentes tienen sus propias ideas sobre tareas a ejecutar. Con objetivos específicos se distinguen los agentes de las aplicaciones multifunción, que son típicamente más pequeños [9].

Las características que debe cumplir un agente para ser considerado como tal son: autonomía, reactividad, iniciativa, habilidad social, etc. [10].

## Agente Conversacional o Chatbot

Un agente conversacional o chatbot es un programa que intenta imitar la conversación que puede proveer un ser humano [11].

Otra definición nos dice que el *chatbot* es un programa que tiene como finalidad simular la conversación escrita, aparentando inteligencia, hasta el momento, de manera temporal [18].

## La Aportación de ELIZA

Uno de los primeros trabajos que se desarrollaron acerca de software con capacidad de comunicarse a través del lenguaje natural en forma escrita es el realizado por Weizenbaum en el MIT, creó un programa al que denominó ELIZA el cual pretendía suplantar a un psicólogo en una sesión de trabajo. Este chatbot tuvo un gran éxito al momento de su creación ya que fue capaz de engañar a varias personas haciéndoles creer que en verdad hablaban con un psicólogo, sin embargo, su creador siempre estaba consciente de que ELIZA no representaba un ente inteligente ya que su principio está basado en la repetición de palabras presentes en una misma conversación constantemente, lo que no era tan susceptible de percibirse al momento de representar una sesión con un psicoanalista [12].

Este chatbot es el precedente y el cimiento de la tecnología conversacional en agentes y gran parte de los trabajos posteriores tomaron como base a ELIZA.

Weizenbaum en sus intentos por crear un programa capaz de comunicarse de forma escrita en el idioma inglés identificó la dificultad y complejidad de que su software pudiera almacenar en un solo sitio las palabras y con ellas emular la conversación con diferentes temas en una misma plática, de tal forma que en lugar de tener un solo almacén de datos, Weizenbaum dividió la base de conocimientos de ELIZA por módulos o temas lo que permitió al chatbot cambiar arbitrariamente de tema al moverse entre sus módulos de conocimiento [12].

## Introducción a Orientación Vocacional

Cada ser humano tiene características personales muy particulares, por lo que desarrolla habilidades únicas. Por ejemplo: Si ponemos a exponer un mismo tema a dos alumnos, cada uno va a describir el tema en la forma que cada uno de ellos interpreto. [2].

Estas diferencias hacen que se tengan destrezas intereses y habilidades muy específicas para cada ser humano. Una destreza o habilidad es la capacidad que se tiene para realizar una tarea específica, por ejemplo el trabajo de carpintería en la que se necesitan destrezas específicas, herramientas, capacidades de atención entre otras. Los intereses, gustos y habilidades personales nos permiten determinar alternativas de nuestra verdadera vocación.

Vocación es la atracción natural a un estado, profesión o carrera, según Patricia Cabrera Z., Docente y guía vocacional en la consultaría chilena. [2].

El concepto etimológico de vocación viene del latín "*vocatio*, *vocationis*" que significa "llamado", "invitación", mejor dicho, la vocación nace desde el exterior de nosotros, que nos involucra a participar en una específica tarea o relación laboral.[1].

Es importante mencionar lo esencial que es la orientación vocacional, porque si ésta es excelente y aparte, conocemos nuestros gustos personales, así como nuestros intereses, capacidades y habilidades esto nos permitirá seleccionar de manera correcta, nuestros estudios universitarios con más interés y gusto, lo que en un futuro se verá reflejado directamente en la calidad del trabajo profesional que desempeñemos, así como en la calidad de vida.

Los objetivos básicos de todo proceso de orientación vocacional están establecidos:

En primer lugar a conocer la personalidad de cada individuo, especificar sus capacidades, gustos, habilidades, motivaciones, intereses, su inteligencia y aptitud.

En segundo lugar, la orientación vocacional debe involucrar a los padres, ya que ellos apoyan en este proceso orientador, siendo debidamente informados de la realidad educativa y laboral que existe para apoyar a sus hijos, solo de manera informativa sin impedir la libre elección.

La orientación profesional tiene como finalidad ayudar a los jóvenes a encontrar su vocación y guiarlos a las actividades profesionales de acuerdo a sus aptitudes [1].

Se puede describir a la orientación vocacional como un proceso que permite ayudar a los jóvenes a conocerse a sí mismo y al mundo de trabajo al se enfrentaran. Para que se preparen y elijan la profesión adecuada a sus aptitudes, intereses y rasgos de personalidad, teniendo en cuenta las posibilidades de estudio y de trabajo existentes [1].

Haciendo un análisis sobre las posibles investigaciones relacionadas con Sistemas Inteligentes Conversacionales para la orientación vocacional, se determinó que en la actualidad no exista una aplicación igual a lo que se propone realizar.

Con base al marco teórico se muestra la estructura del modelo que representa el agente inteligente conversacional para la orientación vocacional a desarrollar.

## Modelo del Agente Inteligente Conversacional para La Orientación Vocacional.

Con base a los conceptos teóricos se propone el modelo del agente conversacional en orientación vocacional.

En la Figura 1 se plasman los elementos que integran el Agente Inteligente para apoyar La Orientación Vocacional el sistema requiere realizarse con el lenguaje de programación AIML, para implementarlo se utiliza el software "Program E" el cual es un intérprete de AIML el cual funciona sobre la plataforma de apache, php y mysql.

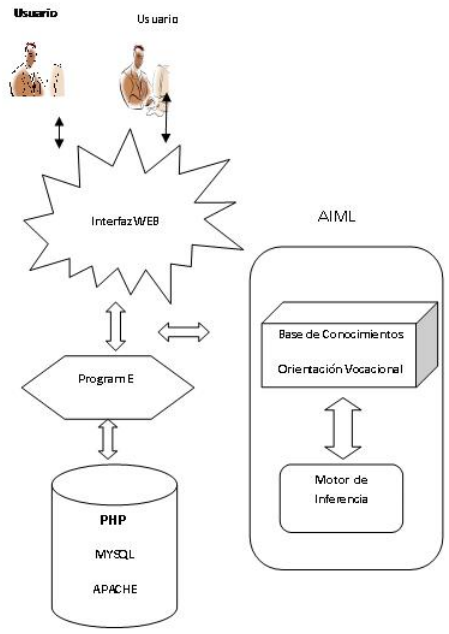



Fig. 1 Estructura del Agente Inteligente Conversacional.

Para desarrollar un chatbot en orientación vocacional se necesitan como *software* principal el lenguaje de programación AIML (lenguaje de Marcas para la Inteligencia Artificial), [17]. Es un leguaje en el cual se establecen categorías de las posible preguntas y expresiones que el usuario puede manifestar, a su vez se determinan las respuestas correspondientes para el usuario. Se apoya de una base de conocimiento que contiene la información y hechos sobre el tema específico a abordar durante el análisis. Otro elemento que apoya la estructura es el motor de inferencia, el que modela el razonamiento humano través de instrucciones. El componente final es la interfaz, la interacción se realiza en línea utilizando lenguaje natural.

El motor de inferencia se integra de un conjunto de instrucciones en el lenguaje de programación, que en este caso es el AIML, son instrucciones que funcionan como un IF o un CASE en todos los lenguaje de programación, la instrucción <pattern>valor de entrada <pattern> es la sentencia, que contiene el valor que espera el motor de inferencia, para proporcionar como salida una de sus respuestas en la instrucción <li>. Ejemplo si el usuario escribe hola, el motor da como respuesta de manera aleatoria una de las posibles indicadas en su código fuente. Por ejemplo:

```
<category>
<pattern>hola</pattern>  //dato de entrada
<template>
<random>
<li> Hola Bienvenido.</li>   //posibles respuestas
<li><suspecting/>HOLA. Gracias por tu visita</li>
<li><offended/>HOLA. Gracias </li>
<li><ironic/>Para un análisis escribe ORIENTACION</li>
</random>
</template>
</category>
```

Se trabaja con el lenguaje de marcas para inteligencia, el cual se basa en el estándar XML; este intérprete no incluye la interfaz con el usuario, sin embargo, dicho motor puede ser incluido en un servidor Web. También se puede agregar como un contacto más en un programa de mensajería instantánea o hacer la interacción vía *Shell*. La base de conocimientos se forma a partir del AIML que consiste en una serie de preguntas del usuario con sus respectivas respuestas; estas preguntas pueden ser agrupados en sets de conocimientos de acuerdo a un tema y además, dentro del AIML se proporcionan una serie de mecanismos que permiten al creador de la base de conocimientos implementar procedimientos para dar realismo a las conversaciones [15].

El lenguaje de programación de AIML fue desarrollado por el Dr. Richard Wallace y la comunidad del *software* de *Alicebot* libremente entre los años de 1995 y 2000. AIML consiste en una serie de etiquetas las cuales permiten manejar y organizar la información entrante y saliente, provee de una estructura única a partir de la cual se accesa a las bases de conocimiento, permitiendo recuperar e insertar texto. La escritura en AIML es costosa en tiempo a menos de que se reduzca la capacidad del chatbot para desarrollar diferentes temáticas [15].

Sintetizando el Agente Inteligente Conversacional se integra principalmente de una base de conocimientos que contienen la información sobre orientación vocacional que es el tema a tratar por este sistema, un motor de inferencia que contienen las instrucciones para que el robot actué y de decida sobre cuales acciones realizar de acuerdo a las expresiones del usuario, un program E que se utilizará como compilador del código, el PHP, mysql y APACHE que son la plataforma que permiten implementarlo en internet.

La psicología y pedagogía son dos ciencias que poyan y dan herramientas para apoyar la función de la orientación vocacional ya que es un proceso que permite ayudar a los jóvenes a conocerse a sí mismo y al mundo de trabajo al que se enfrentaran. Para que se preparen y elijan la profesión adecuada a sus aptitudes, intereses y rasgos personales, tomando en cuenta las posibilidades de estudio y de trabajos actuales. Se tienen que manejar técnicas de psicología y pedagogía para apoyar de manera más científica y formal.

Así mismo, la orientación profesional tiene como fin asesorar y ayudar al individuo a descubrir su vocación y orientarle hacia la actividad cultural o profesional en la que mejor se desenvuelva y ayudarlo a reconocer sus propias aptitudes y asesorarlo sobre cuál debe ser su carrera profesional, no solo para desarrollarla de manera efectiva, sino para poder permanecer en ella.

La finalidad es aprovechar estos métodos y automatizarlos por medio de este agente inteligente a realizar el análisis con base a cuestionarios, manuales y folletos ya establecidos por Instituciones pedagógicas como los de *Kuder*, ediciones pedagógicas latinoamericanas Ltda [16].

**Resultados**

Esta investigación nos permitió diseñar el prototipo de un Agente inteligente en Orientación Vocacional.

En la Figura 2 Se muestran la interfaz del agente inteligente en orientación vocacional.




Fig. 2.- Interfaz del agente inteligente en orientación vocacional.

## Conclusiones

Se puede concluir que con esta investigación se establecen los elementos necesarios para desarrollar un robot virtual el cual es una herramienta de apoyo para los usuarios y especialistas en orientación vocacional.

La ventaja de este proyecto es que la orientación vocacional ya no se llevará a cabo realizando extensos test, se ahorra tiempo al no tener que ir a una oficina de orientación vocacional, el joven podrá realizar este análisis desde la comodidad y privacidad de su casa, este proceso será más agradable ya que se establecerá un dialogo simulado y amigable con otra persona.

La aplicación de nuestro sistema impacta con mayor incidencia en los jóvenes, los cuales podrán desempeñarse con mayor eficiencia en su actividad profesional. Asimismo nuestro sistema repercutirá en una disminución de la deserción y reprobación escolar.

Para continuar con esta investigación a futuro recomendamos aplicar los métodos empleados a la instrumentación de otras aplicaciones similares como: tutores, psicólogos, pedagogos virtuales o como un simple amigo virtual con el cual conversar y lograr las metas establecidas para cada caso de estudio.

## Referencias


[1]. Gladys Eneyda Sánchez Verdugo (1996), Investigación Orientación Vocacional. Los Mochis. Sin.http://www.monografias.com/trabajos14/orienvocac/orienvocac.shtml#FACTO

[2].-Mirna Celene Jaramillo Aldama (2005), Trabajo de Investigación para la Orientación Vocacional, I. T. C.

[3].-Alan Turing, Prueba o Test (2007), http://tikal.cifn.unam.mx/~jsegura/academic/traductores/Cap3.htm

[4].-A.L.I.C.E. and AIML c hat robot, Tip(2008):if you want to comment on any of these news ítems, visit AliceBotBlogSpot.com, http://www.alicebot.org.

[5].- Russell, S. & Norvig, P. (1995). *Artificial Intelligence: A Modern Aproach*, Prentice Hall

[6]. University of Phoenix, http://members. fortunecity.com / jon_alava/ docs/eliza.htm.

[7].-RIVERA, M. A. (2003). ¿APRENDIZAJE DE D IFERENTE TIPO? EXTRAÍDO EL 4 DE ENERO DE 2006 DESDE.HTTP://WWW.EDUCARCHILE.CL/NTG/INVESTIGADOR/1560/ARTICLE-77061.HTML.

[8].- Barber, F. Botti, V. y Koehler, J. (2002). IA: pasado, presente y futuro. *Novática*, 159, 4-7.

[9].-Hayes-Roth (1995). Citado por: Samper, J. (2005). [enlínea] Disponible en: http://www.wikilearning.com/ntroducci%C3%B3_n_al_concepto_de_agente_i-wkccp-5095-3.htm. [21 de Octubre 2005].

[10].- GRACIA, Grupo de Agentes de Software: Ingeniería y Aplicaciones.(2005) *Estado del Arte* [en línea]. Disponible en: http://grasia.fdi.ucm.es/ingenias/Spain/estado/index.php [21 de Octubre 2005].

[11].- ALICE AI Foundation. (2006) *AIML: Artificial Intelligence Markup Language*. Extraído el 8 de enero de 2006 desde http://www.alicebot.org/AIML.html

[12].- Navarro, D. (1997). ELIZA y Computer Power and Human Reason. RED Científica. Obtenido el 8 de enero de 2006 desde http://www.redcientifica.com/doc/doc199903130010.html

[13].-Shah, H. (2005). A.L.I.C.E.-an ACE in Digitaland. Ph.D. Thesis. Harrow School of Computer Science, University of Westminster.

[14].- Minsky, M. (1984). *The Society of Mind*. Simon & Schuster.

[15 ].- Rafael Mejía Santana (2006), Investigación sobre Simulación en línea de una entrevista para la toma de requerimientos de software utilizando un agente inteligente. En la Universidad de Colima, Facultad de telemática.

[16].- Frederic Kuder (1998), Kuder Escalas de Preferencias Perosanl, Forma AH, Ediciones Pedagógicas Latinoamérica, Ltda.

[17].-ALICE AI Foundation. (2006) *AIML: Artificial Intelligence Markup Language*. Extraído el 8 de enero de 2006 desde http://www.alicebot.org/AIML.html

[18].- Álvarez et. al. (2001), http://alejandro313.blog.com.es/2007/01/12/aiquieren_conversar_con_un_chatbot~1548107

# REPORTE TÉCNICO: INSTALACIÓN Y USO DE UN CLUSTER DE COMPUTADORAS

[1]*Contreras Torres Victoria* , [2] *Gutiérrez Morfin Roberto Carlos,*
[3]*Martínez Núñez Alberto Isaac* , [4]*Verduzco Ramírez Jesús Alberto.*

[234]*Instituto Tecnológico de Colima. Av. Tecnológico 1, Villa de Álvarez, Colima. México.,*
[1]*Universidad de Colima, Facultad de Ingeniería Mecánica y Eléctrica, Km. 9 Carretera Colima-Coquimatlán, Colima, México.*
[1]*victoria.cont@gmail.com,* [2]*roca_18_04@hotmail.com* [4]*javrtesis@gmail.com*

## I. INTRODUCCIÓN

Objetivo: Instalar un Cluster y configurar los servicios que se necesitan para llevar a cabo cálculos de diversas áreas del conocimiento.

Un Cluster es un conjunto de computadoras iguales o no, que combinan el hardware de cada una de las computadoras conectadas en una red de alta velocidad, para la comunicación entre cada una de las computadoras haciendo que esta red de computadoras se sienta como una sola, aprovechando los recursos al máximo de cada uno de los elementos.

La descripción de la estructura de un Cluster de computadoras es la siguiente: El Cluster funciona en una estructura en red donde el nodo central (Front-end o servidor) es el que aloja el sistema operativo. En el nodo Front-end el usuario administrador podrá hacer las modificaciones que se deseen a todos los nodos del Cluster. En la Figura 1 se observa la estructura general de un Cluster.

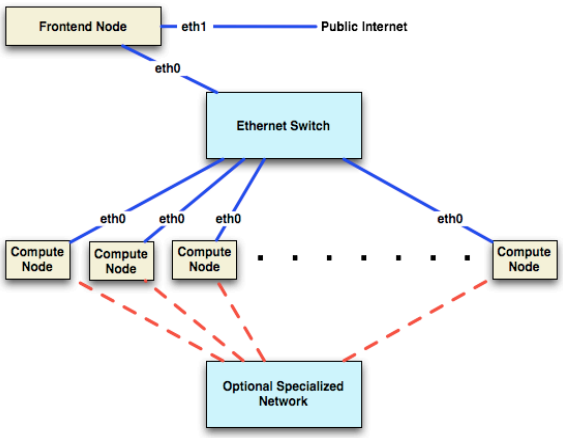



**Figura 1. Arquitectura General de un Cluster.**

El Front-end es el servidor o nodo maestro de nuestro Clúster, se encarga de la manipulación del Clúster, Los nodos son las computadoras esclavas que arman al Clúster, a los nodos se pueden acceder por medio del Front-end o individualmente. Es usual que los nodos no cuenten con una interfaz grafica donde se use el ratón, sino solamente el teclado. Eth1 y Eth0 son las interfaces de tarjetas de red del Front-end donde Eth1 debe ser la salida a internet para el Clúster, y Eth0 identifica a la red exclusiva del Clúster. Toda esta comunicación se realiza por medio de diferentes protocolos y servicios como lo son TCP/IP, SSH, NFS, TFTP, HTTP, DHCP, etc.

El sistema operativo se encarga de administrar los recursos distribuidos de los nodos y presentar ante el usuario y los programas una sola vista de todos los recursos y datos distribuidos.

La red local permite el tránsito de los mensajes de sincronización y los datos entre los nodos que integran el Cluster. Al terminar cada computadora su trabajo le envía los resultados a servidor por medio de la red local y el Fornt-end se encarga de reunirlos y presentarlos al usuario. Esto da la sensación de tener una supercomputadora trabajando, cuando en realidad son un conjunto de computadoras trabajando con fragmentos de un trabajo.

Las ventajas de utilizar Cluster en la ejecución de aplicaciones, consiste en que derivado de la agregación de recursos de cómputo se obtienen resultados en menor tiempo en comparación si solo fueran procesado por una computadora, el costo es mucho menor al de una supercomputadora, y además se dispone de la escalabilidad suficiente para poder

incrementar el número de computadoras utilizadas como nodos.

Las desventajas de un Cluster son entre otras, que la utilización de uno de estos sistemas para la resolución de problemas es una tarea difícil. Para el desarrollo de aplicaciones que sean capaces de aprovechar los recursos del Cluster, es necesario recurrir a diferentes librerías especializadas de programación como MPI y OPEN-MP. Un problema importante inherente a la distribución de los nodos es que la eficiencia global del Cluster depende de la red, ya que esta puede llegar a sufrir el efecto cuello de botella por las necesidades de comunicación de los procesos residentes en los nodos, que requieren comunicarse por medio de la red local.

Sin embargo, a pesar de las desventajas antes descritas los Cluster son una tecnología muy popular actualmente.

## II. PROCESO DE INSTALACIÓN

Después de obtener esta información, se requirió elegir qué sistema operativo utilizaríamos para manejar el Cluster. Luego de esto llegamos a la conclusión de que el sistema que se usaría seria Linux y que la aplicación dedicada a la instalación de servicios y de interfaces en un Clúster seria Rocks Clúster. ¿Por qué Linux?, porque existen muchas distribuciones que son capaces de soportar este tipo de interfaces, son robustas, tienen muy buena documentación, y se tiene la ventaja del software libre. ¿Por qué Rocks? Porque se cuenta con suficiente documentación relacionada con la instalación y administración, además de que es u n proyecto de software de libre distribución en constante desarrollo.

Teniendo esto en cuenta se utilizó la versión de Rocks Cluster 5.4 para 64 bits, después de instalar y realizar pruebas se encontró con un problema en dicha distribución, el problema fue que la versión 5.4 Maverick p ara 64 bits de Rocks Cluster no está completa y hacían falta algunas configuraciones que dichas distribución requería. P or tales motivos se decidió cambiar a una versión anterior, esta es la distribución 5.3 Tacos para 64 bits de Rocks Cluster.

En la parte de instalación del Clúster encontramos varias tareas que se deben de hacer antes de comenzar la instalación del Clúster, una de ellas es el saber cuál computadora será el servidor Front-end y equiparla con 2 tarjetas Ethernet, en nuestro caso utilizamos una tarjeta PCI Ethernet para completar este requerimiento que se solicita por defecto a la hora de la instalación. (nota: si no se le agrega una segunda tarjeta al Front-end antes de la instalación entonces algunos de los servicios no funcionaran de forma correcta como el DHCP necesario para la asignación de IPs de los nodos).

Para comenzar a instalar el servidor o Front-end del Clúster, es necesario descargar el software desde la página oficial y que el software contenga los siguientes paquetes o discos disponibles también en la página oficial localizada en la dirección:

http://www.Rocksclusters.org/wordpress/

Donde se puede encontrar además del software básico de Rocks, diferentes paquetes de software.

A estos paquetes de software y librerías se les llaman Roll. Los Roll contienen librerías o aplicaciones para diferentes áreas del conocimiento. (area51 seguridad, bio bioinformática, web server, y mas) los principales Rolls que debemos tener en cualquier instalación con Rocks son el Roll Base, Web server, Kernel, Core OS Roll disco 1 y 2.

Existe también un disco con todos los Rolls en un solo disco DVD llamado Jumbo, este ya incluye todos los Rolls para dicha distribución.

Se introducen el CD o DVD donde se incluye el Roll Kernel, este se en cargara del booteo en nuestra maquina. Se inicia la instalación y al inicio es necesario escribir (puede variar dependiendo de la versión ) ***build*** para que el instalador sepa que se quiere instalar un Front-end, después de ciertas pantallas se preguntara en una de ellas cual es la IP de la tarjeta Ethernet que se conectara a internet, dicha tarjeta suele ser identificarse con la etiqueta eth1 y puede ser como automática o estática , en el caso exclusivo de nuestro Clúster se seleccionó automática,

En el caso particular de la tarjeta eth1 en donde la IP es dinámica, no se tiene una dirección fija para ella, pero le asignamos a ETH1: 192.168.2.101 esta dirección porque al momento de instalar teníamos en nuestro cuarto de cómputo un router al cual configuramos con ese rango de direcciones. Así que si se va a configurar esta tarjeta procurar que sea con un rango de direcciones que sea común para la institución o lugar donde se colocará el Cluster y así pueda tener de alguna forma una dirección fija para el Front-end.

Luego sigue el paso de configuraciones de red:

- Gateway 192.168.1.10
- DNS servers: 192.168.1.254

Y para finalizar esta parte de configuración se pide el password para el sistema, solo se pide el password porque el usuario con el que se ingresa al Cluster como administrador es root así que así se inicia sesión. Después de todo esto sigue lo que se podría decir la tercera fase de la instalación

Esta parte es sobre cómo se repartirá los recursos de la máquina, en algunos tutoriales se recomienda poner algunos valores que son considerandos los valores mínimos y que pueden ser modificados a gusto del usuario administrador.

La creación física de una red local, en este caso se utilizo un switch con 8 puertos Ethernet, cables Ethernet y 6 computadoras Dell Vostro 430a, con microcontrolador de Intel Core i7 (8 núcleos y 2.8 GHz) , con 4 GB en RAM y 500 GB en disco duro.

Luego de la instalación fue necesario revisar que los servicios estuvieran funcionando de manera correcta, tales servicios como SSH, DHCP, TFTP, NIS, ARP, HTTP, MYSQL y otros.

En la instalación de Rocks Clúster se incluyeron varios paquetes o roles como ganglia, bio, hpc, webserver, cada uno de los cuales es útil para diferentes enfoques de trabajo para lo que se pueda utilizar un Clúster. En pocas palabras Ganglia (Figura 2) se encarga del monitoreo de los recursos del Clúster, memoria, procesadores, red, etc. Bio incluye librerías especiales para aplicaciones de bioinformatica. HPC son librerías que se n ecesitan para trabajar con paralelismo. WebServer son los paquetes comunes como HTTP, Apache, MSQL, y mas, todo para montar un servidor web y poner el Clúster en la red.

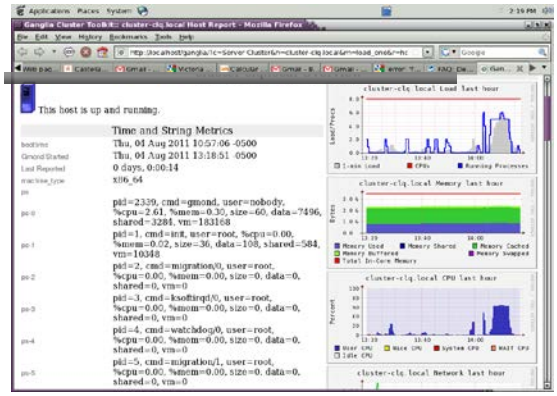



**Figura 2. Sistema de Monitoreo del Cluster Ganglia.**

Ya al contar con los servicios funcionando dentro del Front-end, ahora se procede a instalar los nodos esclavos. Para realizar esto se introduce un comando en la terminal:

```
# insert-ethers
```

Donde se realizará la detección de la maquina y su automática instalación por medio de la red. En la Figura 3 se observa el programa que realiza lo anterior, para ello se debe de elegir una opción del menú en recomendado es elegir la opción PC o Computer. Después se podrá proseguir con el proceso.

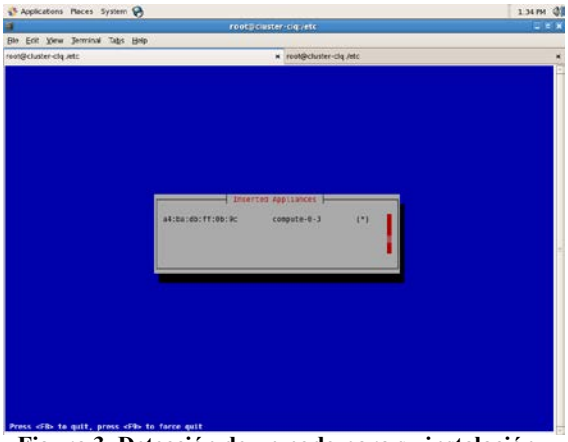


**Figura 3. Detección de un nodo para su instalación.**

Posterior a este proceso de instalación del Clúster sigue la parte de configuración, creación de usuarios, subir el Clúster a internet, y diferentes aspectos más que cada administrador del Clúster puede llevar a cabo para la personalización del sistema.

## III. FASE DE PRUEBAS

En esta fase se escribieron un conjunto de códigos en el lenguaje C usando la librería MPI, con la finalidad de comprobar el funcionamiento del Cluster. Los programas se ejecutaron modificando el número de nodos e incluso retirando los nodos físicamente del Cluster para analizar el efecto en el desempeño al modificar el número en seis, cinco y finalmente en cuatro nodos.

## IV. CONCLUSIONES

En este documento se describe el proceso de instalación y configuración de los servicios de un Cluster construido con cinco PCs de escritorio.

Se ha utilizado el software Rock Cluster como sistema operativo. Así mismo se ej ecutaron algunos códigos de programas escritos en lenguaje ANSI C y la librería MPI.

Finalmente, podemos decir que los objetivos marcados al inicio de este proyecto se al canzaron de manera satisfactoria, dejando valiosas experiencias a los participantes de este proyecto.

## VI. EVIDENCIAS

En las Figuras 4,5 y 6 se muestran imágenes del Cluster instalado.



**Figura 4. El Cluster con seis nodos.**



**Figura 5. El Cluster con cinco nodos.**



**Figura 6. El Cluster con cuatro nodos.**

# REPORTE TÉCNICO: INSTALACIÓN, CONFIGURACIÓN Y PUESTA A PUNTO DE UNA NUBE PRIVADA CON UNA CON UNA ESTRUCTURA (IaaS)

[1]Gutiérrez Morfín Roberto Carlos, [2]Verduzco Ramírez Jesús Alberto,
[3]Rocha Medrano Pedro, [4]Contreras Torres Victoria.

*[123] Instituto Tecnológico de Colima. Av. Tecnológico, Villa de Álvarez, Colima. México.,*
*[4]Universidad de Colima, Facultad de Ingeniería Mecánica y Eléctrica, Coquimalán, Colima.*
*[1]rocarmos@gmail.com, [2]averduzco@itcolima.edu.mx, [3]procha@itcolima.edu.mx, [4]victoria.cont@gmail.com.*

**Resumen**

Este artículo describe la instalación, configuración de servicios y puesta en marcha de una nube privada con la estructura IaaS. Los conceptos básicos del cómputo en la nube son revisados, además se describe el proceso de instalación y la configuración, así como la ejecución de instancias de máquinas virtuales con el software HybridFox.

**Palabras clave:** Computación distribuida, Virtualización, Cómputo en la nube.

## 1. Introducción

Habíamos llegado a pensar que los servicios que brinda una PC (infraestructura, plataforma y software) se iban a ofrecer por medio de Internet como si fuese un servicio de luz, agua, o servicios de ese tipo, pues en la actualidad se ha logrado hacerlo con la tecnología del Cómputo en la Nube (*Cloud Computing).* Los usuarios obtienen estos servicios bajo demanda esto quiere decir que solo lo que requieran es lo que pagarán, nada más.

Las grandes ventajas que trae consigo para las empresas, la reducción de gastos es lo primero, ya no tendrán que tener un equipo de expertos en el centro de cómputo porque la infraestructura ya no estará en las empresas estará en los proveedores de servicios de la nube.

### 1.1 ¿Qué es Cómputo en la Nube?

La computación en nube es el acceso a las computadoras y su funcionalidad a través de Internet o una red de área local. Los usuarios demandan una solicitud de acceso a la nube de un conjunto de servicios web que gestionan un conjunto de recursos informáticos, es decir, las máquinas, de red, almacenamiento, sistemas operativos, entornos de desarrollo de aplicaciones, programas de aplicación. Cuando se concede una fracción de los recursos en la nube, esta se dedica a atender los requerimientos del usuario hasta que se les libere. Se le llama "cómputo en la nube" porque el usuario no puede ver o especificar la ubicación física y la organización de los equipos de alojamiento de los recursos que en última instancia se les permite usar. Es decir, los recursos provienen de una *nube*. Una *nube* es un conjunto de máquinas y servicios web que implementan la computación en nube. (Eucalyptus Systems, 2011)

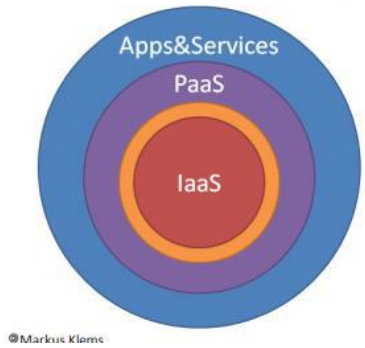



**Fig. 1 Estilos de Infraestructuras de Cómputo en la Nube**

- *(IaaS) Infraestructura como servicio:*

Son los cimientos físicos (hardware) que sostienen la computación en nube Se trata del entramado de una granja de servidores virtuales que ofrecen la posibilidad de seleccionar el sistema operativo y albergar el software que se desee, con almacenamiento "ilimitado" para bases de datos, backups, etc.

- *(PaaS) Plataforma como servicio:*

La plataforma como servicio se refiere a la posibilidad de poder desplegar todo tipo de aplicaciones sin tener que preocuparse de la instalación, gestión y administración de la infraestructura, es el intermediario entre las aplicaciones y el hardware sobre el que se ejecutan.

- *(SaaS) Software como servicio*

Es el software distribuido como servicio por el proveedor sin tener la necesidad de completar sus instalaciones y mantenimiento posterior que suelen ofrecer escalabilidad y balanceo de carga entre otros. De esta manera no es necesario el pago de la licencia por el software propietario, sino un alquiler por el uso del software en cuestión.

## 1.3 Tipos de infraestructuras

- *Nube pública:* La infraestructura está disponible para el público en general y es manejada por una organización encargada de vender servicios en la nube.
- *Nube comunitaria:* La infraestructura es compartida por varias organizaciones que tienen intereses en común.
- *Nube híbrida:* Mezcla de infraestructura pública, privada o comunitaria. Los usuarios son entidades únicas unidas por tecnología estandarizada.
- *Nube privada:* La infraestructura es usada por una sola organización y puede ser operada por terceros. También existen nubes privadas comerciales donde los clientes adquieren espacios dedicados y separados de la nube pública (Sancho, 2011).

## 1.4 Características

Una de las principales diferencias del Cómputo en la Nube, es que no hay necesidad de conocer la infraestructura, esta, pasa a ser una nube donde las aplicaciones y servicios pueden fácilmente escalar, funcionar rápido y casi nunca fallan.

Este tipo de servicio se paga según alguna métrica de consumo, no por el equipo usado en sí, sino por ejemplo, por el consumo de electricidad o por uso de CPU/hora. Entre otras características podemos mencionar:

- Auto Reparable: En caso de fallo, el último backup de la aplicación pasa a ser automáticamente la copia primaria.
- Escalable: Todo el sistema/arquitectura es predecible y eficiente. Si un servidor maneja 1000 transacciones, 2 servidores manejaran 2000 transacciones.
- Regidos por un Acuerdo de Nivel de Servicio (SLA): que define varias políticas como los tiempos esperados de rendimiento.
- Virtualización: las aplicaciones son independientes del hardware en el que corran, incluso varias aplicaciones pueden correr en una misma máquina o una aplicación puede usar varias máquinas simultáneamente.
- Multipropósito: El sistema está creado de tal forma que permite a diferentes clientes compartir la infraestructura sin preocuparse de ello y sin comprometer su seguridad y privacidad (Aroche, 2008).
- Seguridad: Los datos, cuando están en aplicaciones en nube, se alojan en centros de datos, es decir, empresas específicamente dedicadas a la custodia y salvaguarda de datos de empresas de todo tipo: bancos, entidades financieras, gobierno, multinacionales, pymes, etc. Son empresas que cuentan con todas las medidas de seguridad necesarias, tanto físicas como de software, de forma que no exista jamás una pérdida de información ni de integridad de los datos.
- Asignación de recursos en modo multiusuario. A diferencia de las aplicaciones de software tradicionales, en el cómputo en nube, el proveedor tiene una única aplicación que abre a todos los usuarios que desean utilizarla, estableciendo recursos de acceso y prestaciones distintos para cada usuario. Al ser aplicaciones multiusuario, puede haber miles de internautas utilizando la misma herramienta a la vez, cada uno con las mismas o distintas prestaciones (Alcocer, 2010).

## 1.5 ¿Cuál es la relación entre la virtualización y el Cómputo en la Nube?

La virtualización es la capacidad de ejecutar "máquinas virtuales" en la parte superior de un "*hipervisor*". Una máquina virtual (VM) es una implementación software de una máquina que ejecuta los programas como una máquina física. Cada VM incluye su propio *kernel*, sistema operativo, bibliotecas de apoyo y aplicaciones.
Un *hipervisor* proporciona una abstracción uniforme de la máquina física subyacente. Varias máquinas virtuales se pueden ejecutar simultáneamente en un único *hipervisor*. El desacoplamiento de la máquina virtual desde el hardware físico subyacente permite la VM mismo que se inició en distintos equipos físicos.

Así, la virtualización es vista como un facilitador para el cómputo en la nube, proveedor de la flexibilidad necesaria para mover y asignar los recursos informáticos solicitados por el usuario siempre que los recursos físicos estén disponibles (Eucalyptus Systems, 2011).

Ventajas y desventajas del Cómputo en la Nube

- Acceso a la información y los servicios desde cualquier lugar.
- Servicios gratuitos y de pago según las necesidades del usuario.
- Empresas con facilidad de escalabilidad
- Capacidad de procesamiento y almacenamiento sin instalar máquinas localmente.
- Acceso de toda la información a terceras empresas.
- Dependencia de los servicios en línea. (Aroche, 2008)

## 2.0 Desarrollo

En nuestro proyecto utilizamos la distribución de Ubuntu Enterprise Cloud. En la figura 2 se muestra la estructura que genera Ubuntu Enterprise Cloud que permite el despliegue de una Cloud privada, con el enfoque de infraestructura como servicio (IaaS).

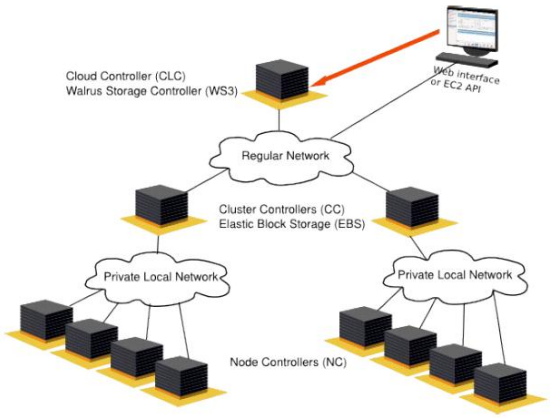



**Fig. 2 Estructura del cloud computing**

La tabla 1 describe los requerimientos físicos necesarios para un sistema que ejecutará uno o más de los siguientes servidores. Pueden residir todos en una sola máquina física que lleva como nombre Front-end.

- Controlador de nube (CLC)[1]
- Controlador de agrupamiento(CC)[2]
- Walrus (el servicio de almacenamiento tipo S3)[3]
- Controlador de almacenamiento (SC)[4]

**Tabla 1. Requerimientos del servidor**

| Hardware | Mínimo | Sugerido | Notas |
|---|---|---|---|
| CPU | 1 GHz | 2 x 2 GHz | Para una interfaz *todo en uno* ayuda tener un procesador de doble núcleo. |
| Memoria | 2 GB | 4 GB | La interfaz web de Java se beneficia de la gran cantidad de memoria disponible. |
| Disco | 5400 RPM IDE | 7200 RPM SATA | Los discos lentos funcionarán, pero se llevarán mucho tiempo en el arranque. |
| Espacio en disco | 40 GB | 200 GB | 40GB es sólo espacio suficiente para una única imagen, caché, etc. A Eucalyptus no le gusta quedarse sin espacio en disco. |
| Red | 100 Mbps | 1000 Mbps | Las imágenes de las máquinas ocupan cientos de megabytes y se deben copiar a los nodos a través de la red. |

Nodos

Uno de los elementos más importantes dentro de la estructura de una nube privada son los nodos, cuyos requerimientos se muestran en la tabla 2.

**Tabla 2. Requerimiento de los nodos**

| Hardware | Mínimo | Sugerido | Notas |
|---|---|---|---|
| CPU | Extensiones VT | VT, 64-bit, Multinúcleo | 64-bit puede correr ambas instancias i386 y amd64; de manera predeterminada, Eucalyptus sólo correrá 1 VM por núcleo CPU en un nodo. |
| Memoria | 1 GB | 4 GB | Más cantidad de memoria significará más y mayores huéspedes. |
| Disco | 5400 RPM IDE | 7200 RPM SATA o SCSI | Los nodos de Eucalyptus hacen un uso intensivo del disco; el cuello de botella en el rendimiento estará posiblemente en las esperas de E/S. |
| Espacio en disco | 40 GB | 100 GB | Las imágenes se almacenarán en caché local; a Eucalyptus no le gusta quedarse sin espacio en disco. |
| Red | 100 Mbps | 1000 Mbps | Las imágenes de las máquinas ocupan cientos de megabytes y se deben copiar a los nodos a través de la red. |

Instalación de Ubuntu Enterprise Cloud

Front-end

1. Descargar la imagen de Ubuntu Server 11.04 y grabarla en un CD.
2. Arrancar el equipo desde el CD y cuando le muestre las opciones de instalación selecciona "Instalar Ubuntu Enterprise Cloud"

[1] Provee la interface con la que el usuario de la Nube interactúa.

[2] Opera entre los Node Controllers y el Cloud Controller. Recibe las peticiones para asignar imágenes de máquinas del Cloud Controller y decide cual Node Controller correrá la instancia de máquina (MInst).

[3] Es usado para almacenar las imágenes de máquinas que pueden ser instanciadas por UEC y para acceder y almacenar datos.

[4] Permite crear dispositivos de bloques persistentes que pueden ser montados en máquinas en ejecución.

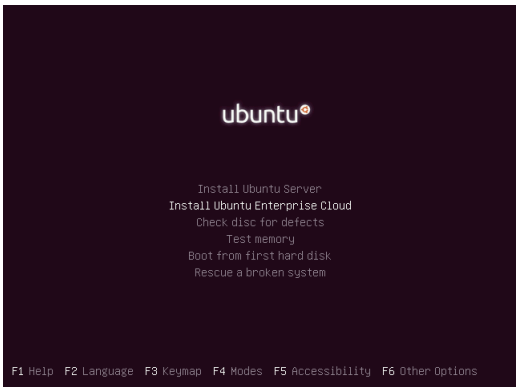



**Fig. 3 Selección del tipo de instalación**

3. Seleccionamos nuestra ubicación, configuración de teclado e idioma de teclado, cuando seleccionemos la opción deseada solo debemos aceptar.

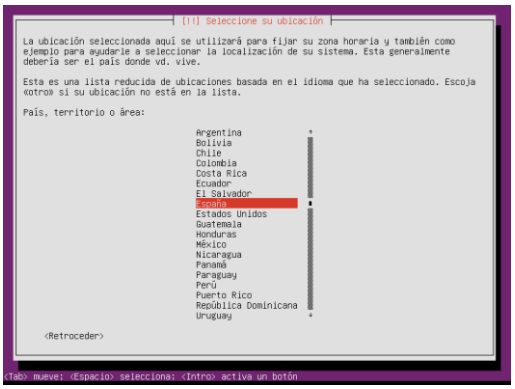



**Fig. 4 Selección del idioma**

4. En la siguiente pantalla introduciremos el nombre que le vamos a dar a nuestra máquina y pulsamos continuar.

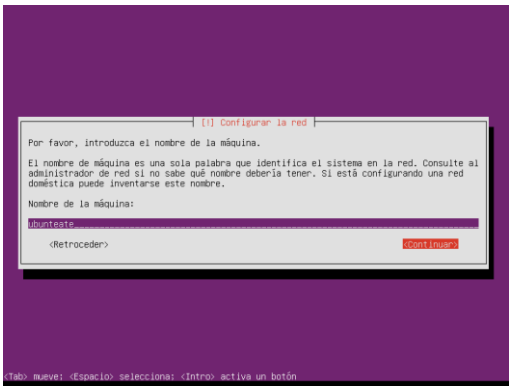



**Fig. 5 Nombre del servidor**

5. Ahora seleccionaremos el modo de instalación de la nube, si Eucalytus detecta una instalación funcionando no mostrará esta pantalla de lo contrario quiere decir que este será nuestro Front-end.

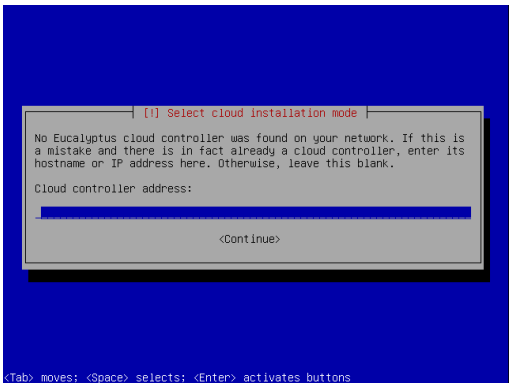



**Fig. 6 Modo de instalación**

6. A continuación, puede elegir los componentes a instalar, sobre la base de la topología elegida, y pulsamos continuar.

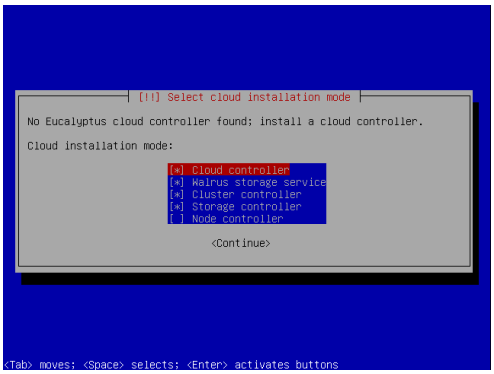



**Fig. 7 Selección de la topología**

7. Ahora seleccionaremos el modo de particionado del disco duro.
   a. Elegimos Guiado – utilizar todo el disco
   b. Nos advierte que vamos a perder todos los datos en el disco, pero que el borrado no se realizará hasta confirmarlo posteriormente.
   c. Pregunta que si deseamos escribir los cambios en los discos y presionamos que sí.
   d. Formateará las particiones.

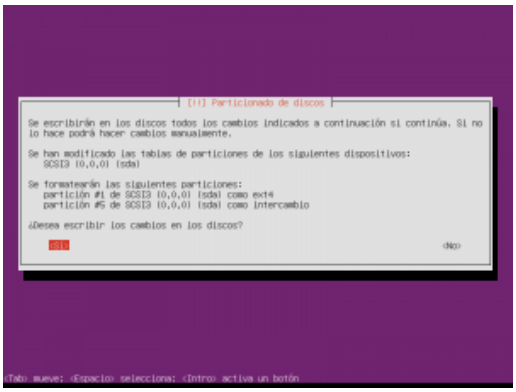



**Fig. 8 Particionado del disco**

8. Se instalará el sistema base y esperaremos a que termine.

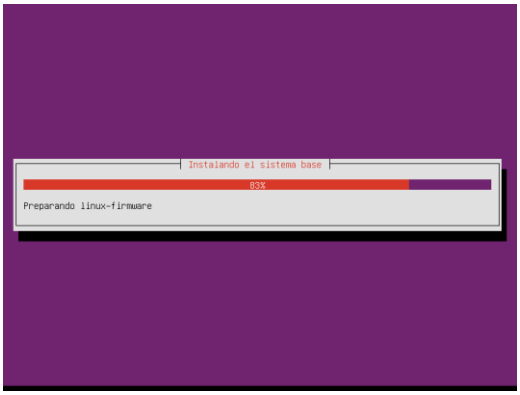



**Fig. 9 Instalación en proceso**

9. Configuración de usuarios y contraseñas:
   a. Nombre de usuario
   b. Escribimos el nombre completo para el nuevo usuario.
   c. Escribimos dos veces la contraseña que queramos ponerle al usuario elegido anteriormente.

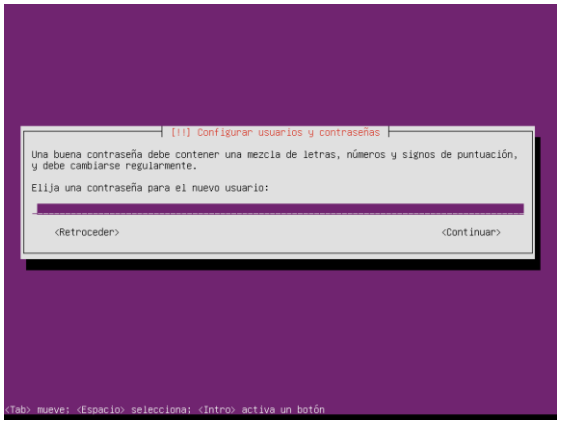



**Fig.** 10 **Configuración del usuario**




**Fig. 11 Cifrado de datos**

10. Después de auto configurarse, nos aparece la siguiente pantalla, no ponemos nada solo pulsamos continuar y esperamos a que termine de instalar los paquetes necesarios.




**Fig. 12 Gestor de paquetes**

11. Actualizaciones de seguridad.
    a. Marcamos Sin actualizaciones automáticas, para actualizar el equipo y se instalarán los paquetes.
    b. Aparece otra pantalla de la configuración del sistema de correo.
12. El sistema nos pide la configuración del Controlador de Cluster de Eucalytus, solo le ponemos el nombre y continuamos.

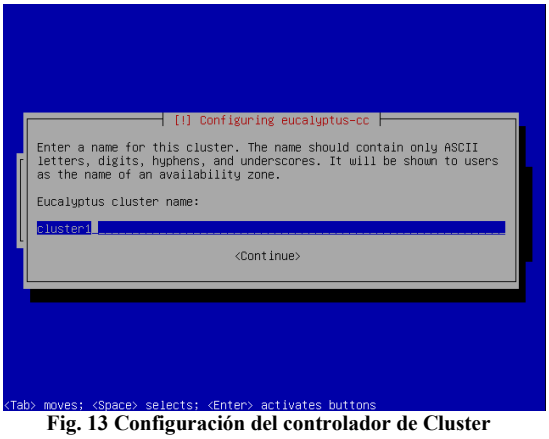



**Fig. 13 Configuración del controlador de Cluster**

13. Configuración del rango de direcciones IP públicas para las máquinas virtuales que se instancien en los nodos de la nube.

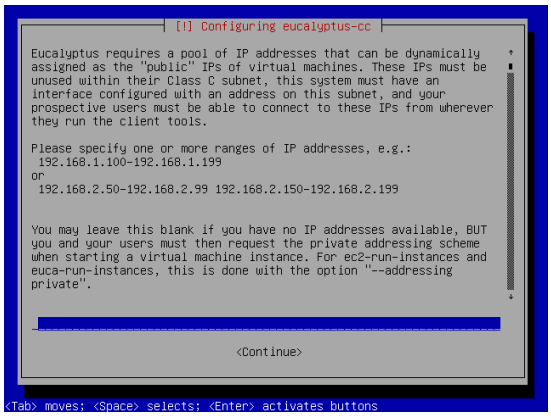



**Fig. 14 Configuración de IPs**

14. Configuración del GRUB
15. Fin de la instalación.

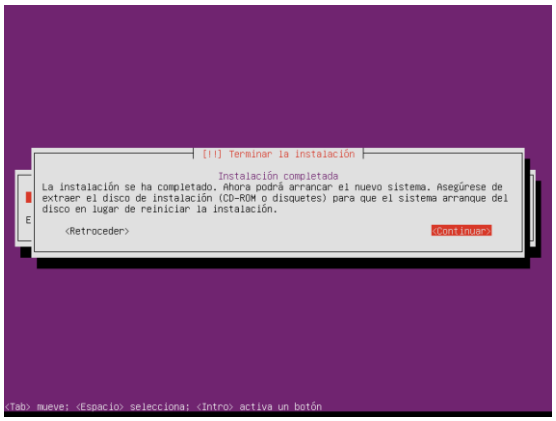



**Fig. 15 Fin de la instalación**

Nodos

1. La instalación del controlador de nodos es todavía más sencilla. Sólo asegúrese de estar conectado a la red en la que ya esté funcionando el controlador de clúster o nube.
2. Se siguen los primeros cuatro pasos que la instalación anterior, cuando llega al punto del modo de instalación de la nube detecta que ya hay instalado un controlador de cluster y solo muestra tres opciones: controlador de nodo, controlador de cluster y controlador de almacenamiento. En este caso seleccionamos Controlador de Nodo y continuamos.
3. Seguimos el mismo proceso de particionado que la instalación anterior.
4. El resto de la instalación debería seguir de manera ininterrumpida. Al terminar la instalación se reiniciará la máquina.

Instalar imágenes

1. Abrimos el navegador y tecleamos https://<ip>:8443 y nos abrirá la interfaz web del controlador de nube.

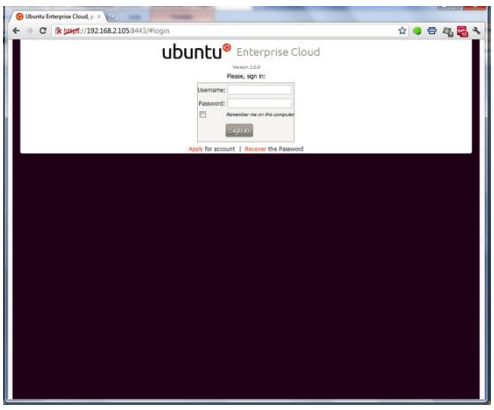



**Fig. 16 Interfaz del controlador de la nube**

2. Si vamos a usar la interfaz por primera vez el usuario y contraseña son *admin* después se puede cambiar si gustas, después de haber entrado nos vamos a la opción de store y le damos instalar a cualquier imagen.

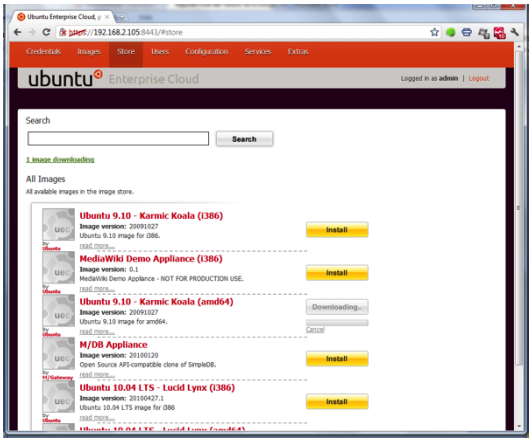



**Fig. 17 Imágenes disponibles**

3. Después de haber instalado la imagen, descargamos, instalamos y configuramos Hybridfox, un complemento de Mozilla Firefox Abrimos el archivo .xpi con el navegador web y dejamos que se instale.
4. En la Pestaña de Herramientas se encuentra el link del complemento, y le damos clic.

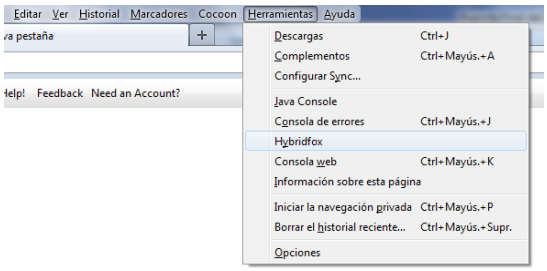



**Fig. 18 Abrir Hybridfox**

5. Configuraremos la región http://192.168.2.105:8773/services/Eucalyptus y tecleamos add y close.

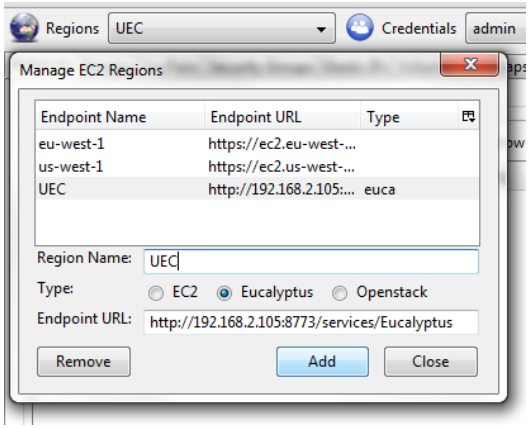



**Fig. 19 Configurar el Hybridfox**

6. Ahora configuramos las credenciales en la interfaz web nos vamos a la primera opción del menú que dice credenciales y hasta el final dice "Query interface credentials", pulsamos el botón que dice "hide keys" y nos mostrara la información que ocupamos y la colocamos como se muestra en la siguiente imagen.

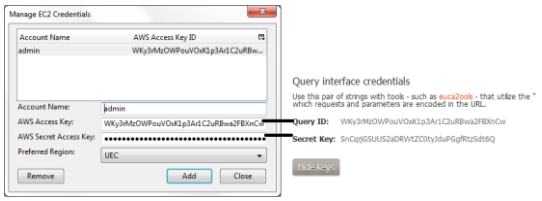



**Fig. 21 Configuración de credenciales**

7. Con esto queda listo nuestro Hybridfox para ejecutar las estancias de imágenes, Ahora creamos un par de llaves dentro de este en la pestaña de" KeysPars" le asignamos cualquier nombre y la guardamos en cualquier ubicación de nuestro PC.

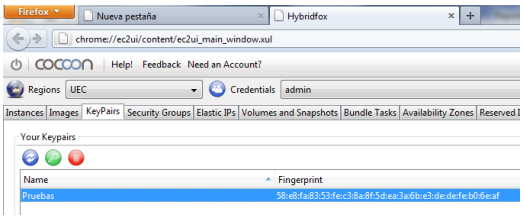



**Fig. 22 Elaborar la Keypair**

8. Con las llaves creadas es posible ejecutar y acceder a una máquina virtual, para lo cual se puede escoger de la pestaña imágenes el identificador emi de una de las imágenes disponibles en el sistema y dar clic en el botón "launch Instances de color verde".

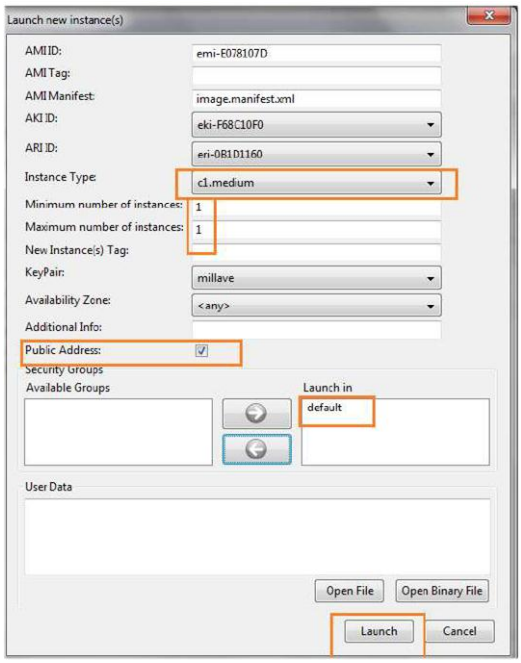



**Fig. 23 Configurar la imagen**

9. Cuando se corre una instancia muestra una leyenda de pendiente y cuando después pasa a una que dice corriendo, cuando ya está así eso quiere decir que ya puedes acceder con un cliente SSH, mientras no.

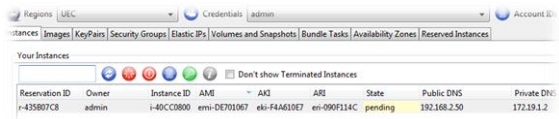



**Fig. 24 Imágenes activas**

## CONCLUSIONES

El cómputo en la nube como punto de vista es uno de los grandes pasos que ha dado las tecnologías de la información, y este gran avance es uno de los modelos que mas reduce los gastos de infraestructura dentro de las

grandes empresas, pero el inconveniente es la desconfianza por dejar información de la confiable en manos de terceros, pero como comentábamos anteriormente se puede hacer una nube hibrida donde la información más delicada se puede tener en una nube privada, que puede ser la mejor solución mas no la óptima en cuestión de gastos nos es conveniente.

La mayoría de las grandes compañías están desarrollando políticas que redirigen sus modelos de negocio hacia la Nube, lo que les está permitiendo aumentar su productividad en corto tiempo y con una inversión inicial de bajo costo, la implementación del comercio electrónico en la nube es una de las propuestas más novedosas en los negocios y una buena oportunidad para las empresas.

También para los usuarios particulares funciona, hemos usado los servicios de la nube pero ni cuenta nos damos en realidad un ejemplo de Cloud publica que más de alguna vez hemos utilizado es *Google Docs*, SkyDrive de Hotmail, Facebook, entre otros, y poco a poco vamos ir aumentando el uso de estos servicios que nos benefician mucho pero aquí la única desventaja es que dependemos del internet si estamos en un lugar donde no haya no podremos acceder a nuestra información.

*Cloud Computing* ofrece como servicio virtualización y optimización de recursos, tanto en capacidad de procesamiento como almacenamiento, esto significa que las aplicaciones y datos del usuario no se encuentran dentro de sus equipos físicos, ni dependen del sistema operativo que estén ejecutando, sino que dependen de los recursos que la Nube proporcione o del servicio que estén pagando, las nubes públicas que mencione anteriormente son gratuitas y nos proporcionan algunas aplicaciones y almacenamiento en línea.

*Ubuntu Enterprise Cloud* y *Eucalyptus* hicieron una buena complementación de sus servicios para poder lograr un sistema que proporcione una nube privada con una fácil instalación y con poco requerimiento de hardware. A partir de este sistema se puede escalar a capas superiores de *Cloud Computing* como Plataforma y Software, lo que implica que dicho sistema podría representar la base para la implementación de servicios en la Nube. Lo más interesante que tiene una estructura similar a la que utilizan los servicios de Amazon y con esto se puede implementar una Nube Privada.

## REFERENCIAS


Sancho, M. C. (2011-17-Abril). *El Financiero*. Retrieved 2011-12-Julio from El Financiero: http://www.elfinancierocr.com/ef_archivo/2011/abril/17/tecnologia2738842.html

Alcocer, A. (2010- 31-Marzo). *SocieTIC*. Retrieved 2011-19-Julio from SocieTIC: http://www.societic.com/2010/03/cloud-computing-caracteristicas-de-las-aplicaciones-en-cloud/

Aroche, S. F. (2008-12-Noviembre). *maestros del web*. Retrieved 2011-15-Julio from maestros del web: http://www.maestrosdelweb.com/editorial/cloud-computing-nueva-era-de-desarrollo/

Eucalyptus Systems, I. (2011). *eucalyptus*. Retrieved 2011-18-Julio from eucalyptus: http://open.eucalyptus.com/learn

Flynn, D. (2010). *Microsoft*. Retrieved 2011 йил 20-Julio from Microsoft: http://www.microsoft.com/spain/virtualizacion/private/overview/default.mspx

# EL ENTORNO URBANO COMO FACTOR EN EL COMPORTAMIENTO SOCIAL, CASO INFONAVIT - LA ESTANCIA

[1]Covarrubias Ruesga María Silvia del Rocío, [2]Leyva Palacios Alejandrina,[3]Flores Toscano Oscar Alejandro.
[1] Instituto Tecnológico de Colima, Avenida Tecnológico No. 1, C.P. 28976, Villa de Álvarez, Col. México.
*[1]rociocov@hotmail.com, [2]leyva_palacios@hotmail.com, [3]dan_alex@hotmail.com.*

*Resumen-* **El presente artículo es la continuación de la investigación realizada en la colonia INFONAVIT-La Estancia, iniciada en Julio de 2009. Esta investigación tiene el fin de demostrar que el mejoramiento de las condiciones del entorno urbano influye de manera positiva en el comportamiento social.**

**Durante la primera parte de la investigación, pudimos percatarnos, mediante visitas, encuestas y entrevistas, que en la colonia INFONAVIT-La Estancia existen problemáticas sociales así como deficiencias en la imagen y el entorno urbano, iniciándose en ese entonces obras de mejoramiento.**

**En esta segunda parte, después de finalizadas dichas obras de mejoramiento, se recabó de nuevo información de la colonia, aplicando la misma encuesta de hace dos años, para poder así comparar los resultados del antes y después de las adecuaciones realizadas en esta colonia por parte de las autoridades. Dicha comparativa nos servirá para verificar si se produjo un impacto en el comportamiento social.**

**Debido a que las mejoras realizadas en la Colonia INFONAVIT- La Estancia, no se llevaron a cabo en todas las zonas identificadas como problemáticas, se nos abre la posibilidad como investigadores el volver a realizar una comparativa, esta vez entre las áreas rehabilitadas y las no intervenidas.**

***Temas claves-*** **ciudad, imagen urbana, problemas sociales, entorno urbano.**

## I. INTRODUCCIÓN

Parece evidente, desde un punto de vista social, que hay diferentes zonas en las que es más probable encontrar altos niveles de delincuencia[1]. Hope y Hough (1988)[2] y Mayhew, Aye Maung y Mirrless-Black (1993)[3], por ejemplo, se relacionan los índices de delincuencia con tres tipos de zonas:

1. sub-zonas de alto nivel en las zonas céntricas deprimidas de las ciudades (incluiría las casas de los ricos y las zonas de edificios de propiedad privada en ocupación múltiple);
2. zonas multirraciales que se corresponden con viviendas privadas en alquiler; y
3. complejos urbanísticos de subvención municipal en alquileres más reducidos/pobres, ubicados en zonas céntricas deprimidas o en el anillo exterior.

Numerosos estudios señalan que las características de los barrios influyen en un mayor desarrollo de violencia tanto en adultos como en niños y por igual en ambos sexos (Farrintong, Sampson y Wikström, 1993[4]; Hawkins et al., 1999[5]; Kupersmidt et al., 1995[6]; Sampson y Lauritsen, 1994[7]; Sampson, Raudenbush y Earls, 1997[8]; Scott, 2004[9]).

De la misma forma, algunos autores ponen en evidencia que los barrios más desfavorecidos están asociados a u na mayor presencia de sucesos vitales estresantes y, a su vez, a una mayor presencia de conductas agresivas en los jóvenes.[10] Attar, Guerra y Tolan (1994)[11], confirman esto en sus investigaciones. En comparación con los jóvenes que vivían en otros barrios más favorecidos, éstos estaban expuestos a mayores sucesos angustiosos, lo que provocaba un aumento de comportamientos agresivos constatados por el profesor durante el periodo de un año. Asimismo, es muy posible también que las condiciones de una vida estresante derivada de vivir en un barrio desfavorecido, que provoca incomodidades para los niños y muchos problemas a los padres, les

dificulte la tarea de criar a sus hijos de un modo constructivo (Scott, 2004).[12]

"Pese a que la correlación de variables realizadas no da elementos contundentes para afirmar que los delitos se dan por determinados aspectos urbanos, si existen pistas referidas a que hay causantes como la presencia de graffiti, imagen descuidada de los componentes arquitectónicos, los remetimientos de las edificaciones y los usos de suelo sobre todo aquellos en donde hay poca vigilancia por parte de los habitantes hacia el exterior, pueden llegar a ser elementos generadores de delitos." [13]

En una investigación realizada (ver referencia), se encontró que en cuanto al "Deterioro de los espacios públicos: el 42.11% de los espacios con delincuencia esta muy deteriorada, el 31.05% esta en regular estado y el restante en buenas condiciones"[14], mientras que en la colonia INFONAVIT el **45.79**% está en malas condiciones, el **26.75**% en regulares y el **27.46%** (correspondiente a las áreas rehabilitadas) en buenas.

NOTA: Espacios públicos en malas condiciones: andadores que presentan inundación, la calle donde son comunes los accidentes automovilísticos, el jardín donde se reúnen pandillas, la casa de usos múltiples abandonada, andadores donde son comunes los robos, locales comerciales que están sin uso y una cancha que esta media escondida y donde se reúnen pandillas.

Regulares: los jardines que no han sido rehabilitados, pero que a pesar de eso no están en tan malas condiciones, además de que su ubicación es más favorable por estar cerca de vialidades más transitadas.

Buenas condiciones: los jardines remodelados, y el área lineal rehabilitada que se encuentra a la entrada de la colonia por Avenida Niños Héroes.

## II. DESARROLLO DE INVESTIGACIÓN

Nuestro objetivo es demostrar mediante las comparativas, en los resultados de las encuestas, que el mejoramiento de la imagen urbana, en colonias de alta densidad que tienen problemáticas sociales, trae beneficios a sus habitantes. Ya que estas colonias populares como INFONAVIT-La Estancia, cuentan con casas pequeñas habitadas por varias personas, por lo que requieren una compensación de espacio al exterior de las mismas; y cuando sucede que esos espacios comunes son mal utilizados por no contar con un diseño agradable que los atraiga hacia actividades recreativas y productivas adecuadas, el espacio público se convierte en un área problemática que promueve el comportamiento antisocial.




Fig. 1. Localización de la colonia INFONAVIT-La Estancia.

### Comparación de resultados

¿Se siente tranquilo de vivir aquí?. En esta pregunta realizada en el 2009 un 62% de los encuestados respondieron que si, y en esta segunda aplicación de encuestas en el 2011, la cantidad de personas que respondió que si se sentía tranquilo de vivir en esta colonia aumento en un 21% en relación con la primer encuesta. Es un buen avance ya que después de las remodelaciones realizadas en la colonia en el 2009 en menos de 2 años las personas han notado cambios positivos.

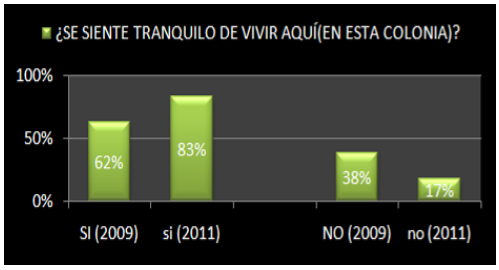



Fig. 2. Hay una mejoría en la percepción de la población, respecto a tranquilidad.

Y es más evidente el cambio cuando se comparan resultados entre las zonas rehabilitadas y las que no han sido intervenidas.

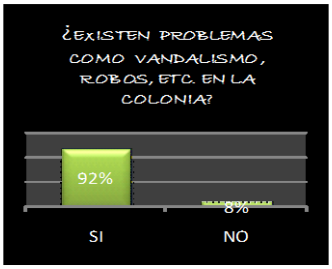



Fig. 3. Resultados de la encuesta del 2009.

Ante la misma pregunta: ¿Existen problemas como vandalismo, robos, etc. en la colonia?

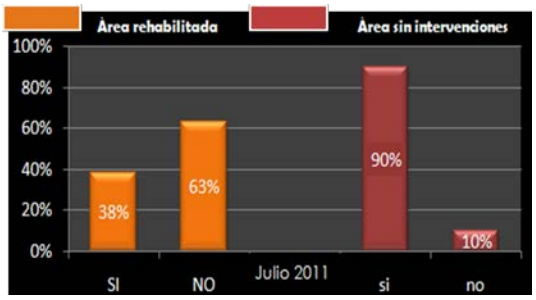



Fig. 4. Resultados de la encuesta del 2011, comparativa entre zonas.

Una de las preguntas realizadas en el 2009 era si existían áreas en donde se dieran reuniones de pandillas, a lo que respondió un 46% de los encuestados que si, y llegaban a identificar estas reuniones de pandillas en calles privadas y de acceso restringido, algunos parques en mal estado y avenidas. Pero en el levantamiento realizado a d os años de la remodelación de jardines en esta colonia, se puede apreciar que los puntos de reunión de pandillas se redujeron a tres puntos, en un sector de toda la colonia, aún sin intervención, en el que se dan conflictos y robos. Las características de esta zona son: poca iluminación en calles, locales comerciales abandonados, casas deterioradas, jardines descuidados, equipamiento urbano sin uso (casa de usos múltiples), andadores estrechos y traza urbana conflictiva. Se deben enfocar esfuerzos en transformar este sector para disminuir la delincuencia en la colonia, lo cual se podrá demostrar continuando esta investigación.




Fig. 6. En rojo las áreas donde se reúnen pandillas. En verde áreas rehabilitadas. En amarillo, el campo de futbol que solo fue empastado. (Encuesta 2011)



Fig. 7. En verde áreas recreativas. En circulo las zonas señaladas como reunión de pandillas. (Encuesta 2009)

Otra pregunta realizada en el 2009 en la colonia en cuestión fue para identificar esas zonas en donde los robos son mas usuales, ante ello la mitad de la población identificaba estos sitios con problemas, algunos de ellos siendo jardines, andadores y calles con acceso limitado. En el 2011 al realizar la misma pregunta, los habitantes en su mayoría si identifican los lugares, pero no porque existan mas sitios, sino porque coinciden en mencionar el mismo lugar; esto nos quiere decir que los sitios que anteriormente habían tenido problemáticas y donde los robos eran comunes se han convertido en sitios mas tranquilos y con mayor seguridad a partir de su rehabilitación, haciendo mas evidente los lugares que no lo son. Esto nos demuestra que al remodelar jardines, mobiliarios y pavimentos en zonas en donde existen conflictos sociales la tendencia de problemáticas tiende a disminuir, ya que dichos espacios empiezan a funcionar para lo que fueron creados en un inicio, es decir son utilizados por niños, jóvenes y familias en general, de esta forma los grupos de delincuentes buscan nuevos sitios para reunirse y cometer delitos. Es por ello que se tiene un cambio positivo en sus alrededores.




Fig. 8. En general aun existe la problemática, y es más fácilmente identificable. (2011)

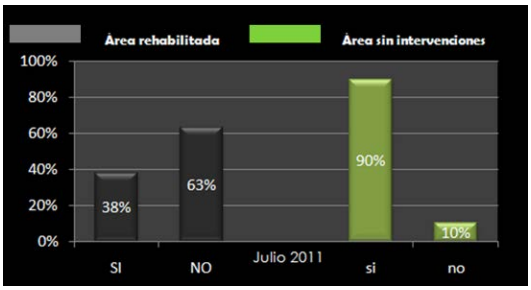



Fig. 9. En las zonas rehabilitadas, menos personas identifican lugares de robos. (2011)




Fig.10. En rojo las áreas donde hay robos frecuentes. En verde áreas rehabilitadas. En amarillo, el campo de futbol que solo fue empastado. (Encuesta 2011)

En la encuesta realizada en el 2009, se pudo apreciar que los jardines y áreas si tenían utilidad en la colonia y que los habitantes asistían con regularidad a estos sitios, sin embargo se presentaban ciertos problemas con las instalaciones y mantenimiento de los mismos y por ello el uso no era tan frecuente. Después de las remodelaciones realizadas a jardines, canchas y centros de reunión en el 2011, al recabar información sobre la misma utilización de estos espacios, la gente respondió que ahora van con mucha mas frecuencia a estos espacios, incluso que la cantidad de horas la cual dedica a estar en estos sitios aumento de 1 horas al día a 3 horas, ya sea para realizar actividades familiares o de deporte. Como se puede evidenciar en las gráficas, el aumento fue considerable de un 69% a un 94%, teniendo un incremento de 25% en utilidad de espacios y frecuencia de uso. Y el 100% de los habitantes de las zonas rehabilitadas manifiestan asistir a esos jardines.

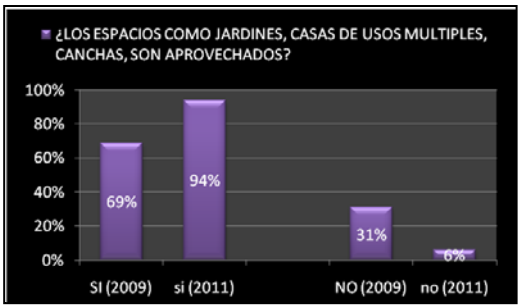



Fig. 11. Comparativo del aprovechamiento de jardines y canchas.

En la aplicación de la encuesta se omitió cuestionar sobre el uso de la casa de usos múltiples ya que presenta signos evidentes de abandono por parte de las autoridades.




Fig.12. Casa de usos múltiples, en abandono. (2011)



Fig.13. Casa de usos múltiples, en abandono. Vista hacia la unidad medica familiar 16. (Foto del 2011)

En la pregunta ¿les gusta asistir a los jardines de la colonia?, se pudo apreciar que se incrementó el número de personas con interés de visitar los jardines, esto quiere decir que las remodelaciones de los mismos no solo fueron apreciadas por quienes si los llegaban a visitar, sino que también por quienes no tenían la tendencias de visitarlos. Estos últimos pudieron apreciar un cambio atractivo por lo que se incrementó la asistencia a estos sitios. Es claro que el proponer mesas de trabajo, áreas de

recreación, internet inalámbrico, cursos de instrucción y otros, es un beneficio que incentiva el interés de las personas por visitar sitios al aire libre y cuidarlos, al utilizar positivamente el tiempo de ocio, causando un cambio benéfico en la sociedad.

Otra problemática que se presenta en la colonia son los accidentes viales, principalmente en la avenida 1º. de mayo, donde existe un aumento en el tránsito vehicular, faltando señalizaciones, lo que provoca confusiones y accidentes. (Este tema requerirá de otra investigación).

En la colonia existen inundaciones debido a la falta de mantenimiento a las bocas de tormenta, además de que la inclinación que presenta la colonia, las calles asfaltadas de la parte alta y la disposición que presentan las casas, favorecen este tipo de problemas, en el área entre los andadores Río Blanco y Cananea.



Fig.14. fotografía tomada en el 2009. Entre los andadores Rio Blanco y Cananea.



Fig.15. fotografía tomada en el 2011. Entre los andadores Rio Blanco y Cananea.

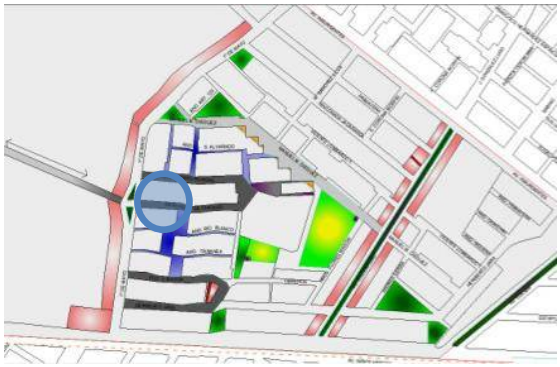


Fig.16. ubicación del área con problemas de inundación. Entre los andadores Rio Blanco y Cananea. En gris, las calles asfaltada.

La rehabilitación del jardín, ubicado a un lado de la cancha techada, trajo beneficios que se pudieron percibir rápidamente, fue (fig. 7, triangulo de la derecha).



Fig.17. fotografía 2009. En un área identificada como problemática, por ser lugar donde se reunían pandillas.



Fig.18. fotografía tomada en el 2010.

En el jardín ubicado en la calle Manuel M. Diéguez no se identificaban problemas de robos o reuniones de pandillas, aunque presentaba visible deterioro y acumulación de basura, pero al cambiar su fisonomía es aprovechado por varios pequeños, los cuales manifiestan estar contentos aunque dicen que: “faltaron porterías”.



Fig.19. fotografía tomada en el 2009. Jardín de la calle Manuel M. Diéguez.



Fig.20. fotografía actual (2011). Jardín de la calle Manuel M. Diéguez.



Fig.21. fotografía actual (2011). Jardín de la calle Manuel M. Diéguez.

Sin embargo aun existen áreas con mayores problemáticas, que se identificaban desde el inicio de la investigación en 2009. Dichas zonas no han presentado ningún tipo de mejora, y el problema ha ido en aumento, pues durante las visitas realizadas al lugar (alrededor de las 2.00pm y 4:00pm), se veían jóvenes con evidente problemas de drogadicción, en áreas que deberían ser para niños, pero que presentan gran deterioro y abandono, a pesar de encontrarse a un costado de la Unidad Médica Familiar del IMSS, de una escuela y del Jardín de Niños. (fig.6 y 10)(fig.22, 23)



Fig.22. fotografía actual (2011). A la salida del jardín de niños.



Fig.23. fotografía actual (2011). A un costado de la unidad medica familiar del IMSS

El área denominada como Plaza Solidaridad, sobre la avenida del mismo nombre (fig.6 y 7), también presenta problema de reunión de pandillas, no siendo aún atendido. (fig. 24)



Fig.24. fotografía actual (2011). En plaza solidaridad.

## CONCLUSIONES

Esta es ya la segunda parte de la investigación, en la cual, a partir de las mejoras que se realizaron en la colonia INFONAVIT- La Estancia por parte de las autoridades, y mediante la comparación de los resultados obtenidos del antes y el después de las mejoras al espacio urbano, podemos llegar a s ustentar que: aunque la problemática urbana no es el único tema que hay que atender, el mejoramiento de de la imagen urbana tiene implicaciones positivas en la sociedad, y que de esa forma tienden a disminuir los problemas sociales.

Aunque no se dio de forma demasiado evidente, debido a que aun falta mejorar las áreas más conflictivas, si se pudo percibir un cambio en el comportamiento de los habitantes, pues ya eran más las familias que visitaban los parques y jardines, y como mencionamos anteriormente esto evita que dichas áreas sean adoptadas por pandilleros.

Otros datos valiosos e interesantes que pudimos obtener son:

- A pesar de que hace dos años y ahora aun existen deficiencias urbanas que favorecen el comportamiento antisocial, los habitantes mencionan que les gusta vivir ahí. Esto nos muestra, que a pesar de tener necesidades y problemáticas y de percatarse de ellas, sienten amor hacia el lugar que habitan. Y esto es muy benéfico, pues con el debido apoyo gubernamental y un adecuado plan estratégico de rescate, se puede lograr que los habitantes sean miembros activos del mejoramiento urbano y social.
- Gracias a que se rehabilitan los espacios públicos, se disminuyen algunos de los problemas sociales predominaste en colonias de alta densidad como es el caso de INFONAVIT-La Estancia.

- Sabemos que por si solo, el mejoramiento de la imagen urbana no será el regenerador de la sociedad, pero también sabemos que forma parte importante de que nosotros como sociedad habitemos en un ambiente agradable que permita el desarrollarnos en armonía.

- Además de la rehabilitación de espacios públicos, se deben implementar campañas de educación ambiental, educación en la familia, y demás programas de apoyo en beneficio de la sociedad, para lograr un avance significativo e integral en el mejoramiento del comportamiento social.

- La planeación urbana, no tiene porque ser dirigida por intereses personales o de unos cuantos, pues por falsas soluciones a los problemas y necesidades de la sociedad, se crean mayores problemáticas sociales futuras.

- Es necesaria la aplicación de una planeación estratégica y olvidarnos de la planeación tradicional, pues la solución de los problemas actuales debe ser con una visión a largo plazo, donde las generaciones futuras no tengan que pagar las malas decisiones actuales.

## TRABAJO FUTURO

Se buscará el apoyo de instituciones gubernamentales, privadas y de los habitantes del lugar, para rehabilitar las zonas problemáticas identificadas que aun no han sido atendidas. Y de esa forma contribuir en el mejoramiento de nuestra sociedad.

## AGRADECIMIENTOS

Agradecemos el apoyo otorgado por el Instituto Tecnológico de Colima, que nos dio la oportunidad de participar en el evento de Formación de Jóvenes Investigadores, pudiendo de esta forma dar seguimiento al presente trabajo y muy en especial a la Dra. Arq. María Silvia del Rocío Covarrubias Ruesga quien volvió a confiar en nosotros, nos encaminó y asesoró la realización de este trabajo.

## REFERENCIAS

[1] Mª Elena de la Peña Fernández. Conducta antisocial en adolescentes: Factores de riesgo y de protección. Madrid 2010

[2] Hope, T. y Hough, M. (1988). Area, crime and incivilities: A profile from the British Crime Survey. En T. Hope y M. Shaw (eds.), Communities and crime reduction Londres: HMSO.

[3] Mayhew, P., Aye Maung, N. y Mirrless-Black, C. (1993). The 1992 British Crime Survey. Londres: HMSO.

[4] Farrington, D.P., Sampson, R.J. y Wikström, P-O.H. (1993). Integrating individual and ecological aspects of crime. Estocolmo, Consejo Nacional para la Prevención de la Delincuencia.

[5] Hawkins, J.D., Herrenkohl, T.I., Farrington, D.P. Brewer, D., Catalano, R.F., Harachi,(1999). A review of predictors of youth violence. In R. Loeber y D.P. Farrington (Eds), Serious and violent juvenile offenders: Risk factors and successful interventions. Thousand Oaks: Sage

[6] Kupersmidt, J.B., Griesler, P.C., De Rosier, M.E., Patterson, C.J. y Davis, P.W. (1995). Childhood aggression and peer relations in the context of family and neighborhood factors. Child Development,

[7] Sampson, R.J. y Lauritsen, J.L. (1994). Violent victimization and offending: individual, situational, and community-level risk factors. En A.J. Reiss y J.Roth (Eds)

Understanding and the preventing violence. Washington: National Academic Press.

[8] Sampson, R.J., Raudenbush, S.W. y Earls, F. (1997). Neighborhoods and violent crime: A multilevel study of collective efficacy.

[9] Scott, S. (2004). Agresividad infantil grave: ¿es realmente posible prevenirla?.VIII Reunión Intenacional sobre Biología y Sociología de la Violencia. Valencia: Centro Reina Sofia para el Estudio de la Violencia.

[10] Mª Elena de la Peña Fernández. Conducta antisocial en adolescentes: Factores de riesgo y de protección. Madrid 2010

[11] Attar, B., Guerra, N.G. y Tolan, P. H. (1994). Neighborhood disadvantage, stressful life events, and adjustment in elementary school children, Journal of Clinical Child Psychology.

[12] Scott, S. (2004). Agresividad infantil grave: ¿es realmente posible prevenirla?.VIII Reunión Internacional sobre Biología y Sociología de la Violencia. Valencia: Centro Reina Sofia para el Estudio de la Violencia.

[13] M.C. Reyna Valladares Anguiano/ Estructura urbana y delincuencia, el caso de Colima-Villa de Álvarez 1999-2002/Coquimatlán colima 2005.

[14] Proyecto de investigación elaborado por Ana Isabel Galvíz Mosqueda, surgido del trabajo Estructura urbana y delincuencia, el caso de Colima-Villa de Álvarez 1999-2002/Coquimatlán colima 2005.

# FACTIBILIDAD ECONÓMICA DE LA ELABORACIÓN DE VIVIENDAS A BASE DE CONTENEDORES METÁLICOS PARA EL ESTADO DE COLIMA

[1]Moreno Peña Ricardo, [2]Aviña Iglesias Ramón, [3]Guerrero García Isabel, [4]Tadeo Ruiz. Ana C.
[1234]Instituto Tecnológico de Colima, Av. Tecnológico No. 1, Col. Liberación, Villa de Álvarez, Colima.
[1]*rmorenor@infonavit.org.mx,* [2]*raai5@hotmail.com,*
[3]*isabel_itc@hotmail.com,* [4]*tadeo2005_2@hotmail.com*

***Resumen-*** **El presente proyecto muestra la posibilidad de brindar a los habitantes del Estado de Colima la opción de construir viviendas con pocos recursos económicos, aborda temas como sistemas constructivos tradicionales y sistemas constructivos a base de contenedores metálicos. El objetivo general es realizar una evaluación económica de la construcción de una vivienda hecha a base de contenedores metálicos.**

***Temas claves-*** **vivienda, mampostería, contenedores metálicos.**

## ABSTRACT

The current project shows the possibility of giving the people of the state of Colima, the option to build houses with little economic resources. It also contains issues like traditional constructive systems and constructive systems based in metallic containers. The General Objective is to perform an economic evaluation of the construction of house made out of metallic container.

.

## INTRODUCCIÓN

El puerto de Manzanillo ha crecido considerablemente, debido al comercio marítimo que llega al puerto; por lo tanto, el manejo de los contenedores metálicos ha aumentado de igual manera, cuando estos se dañan tienen que ser reciclados a un bajo precio.

Esta forma de construir es muy importante ya que sirve como apoyo para las familias de pocos recursos, esto no reduce las condiciones de habitabilidad, ya que cumplen con todas las normas de construcción. Este nuevo sistema ha servido mucho para apoyo en los países Latino Americanos, ya que con las tragedias, los contenedores has servido como viviendas para los damnificados, y dándoles una calidad de vida adecuada para la sociedad, ejemplo, el País de Haití. [1] Otro País que desarrolla oficinas hechas con contenedores marítimos es Argentina, en donde presenta la Empresa 4housing el proyecto a la venta [2].

También el Arquitecto Luis de Garrido autor del libro Sustainable Architecture Containers, un ejemplar cuyo objetivo es mostrar una metodología para hacer arquitectura sostenible utilizando contenedores. El tipo de vivienda que el Arquitecto nos muestra en su obra está considerada como un referente global en arquitectura sostenible. Luis de Garrido fue elegido "Architect of the year 2008" por la ISBA. El libro se completa con 150 ilustraciones y 60 esquemas que muestran el funcionamiento bioclimático de cada proyecto.

## FUNDAMENTO TEÓRICO

En este apartado presentamos los conceptos teóricos que respaldan la presente investigación.

**Vivienda**
Se define como un refugio temporal o permanente destinado a la habitación humana. [3].

**Mampostería**
Sistema tradicional de construcción que consiste en realizar muros y paramentos, para diversos fines, mediante la colocación manual de los elementos o los materiales que los componen (denominados mampuestos) que pueden ser, por ejemplo: Ladrillos, bloques de cemento prefabricados, piedras.

**Contenedores Marítimos**
Es una caja metálica, de la forma de un paralelepípedo, sirviendo para el transporte de mercancías y que puede utilizarse con varios modos de transporte. El comité ISO TC-104 definió el estándar detallado (dimensiones,

materiales, limitaciones, etc) aplicado a los contenedores. Es detallado en las normas ISO 668 (1995) e ISO 1496-1 (1990).

## COMPARACIÓN DE MODELOS DE CONSTRUCCIÓN

### Vivienda de Interés Social Medio

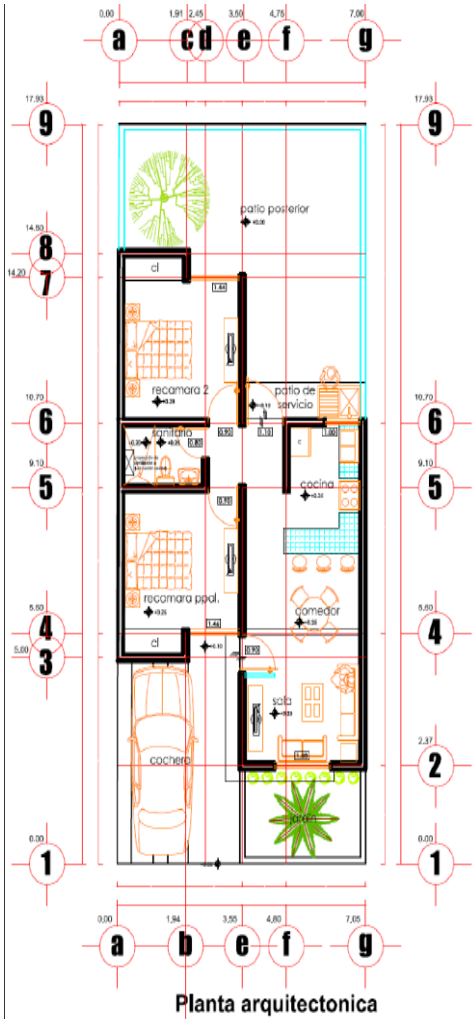



Fig. 1 Planta Arquitectónica de Vivienda de Interes Social Medio.

Para el desarrollo de la planta arquitectónica, se consideran los espacios básicos como lo son: una cochera, una sala, un comedor, una cocina, dos recamaras, un baño completo y un patio de servicio, respetando el reglamento de construcción de vivienda para el estado y cumpliendo con los porcentajes de ventilación e iluminación.

Las dimensiones de los contenedores más utilizados en la region son de dos tipos, los que miden 12m. X 2.5m. Y las de 6m. X 2.5m. Esto quiere decir que utilizamos 2 contenedores de 12 x 2.5, logramos un frente para una vivienda de 5m. Y una superficie de construcción de 60m². Sin embargo este frente es muy poco para ser comparado con las viviendas de tipo de interés social medio-bajo, por lo que se trataría de unir al centro de ambos contenedores láminas de un metro de ancho y así lograr frentes de 6m. o 7m.

A continuación un plano en cual achurado con líneas azules nos indica cuáles serán las áreas que cubrirán los contenedores, y de esta manera tener una idea clara de cómo se obtienen los generadores especialmente en cortes generados en las mismas laminas.

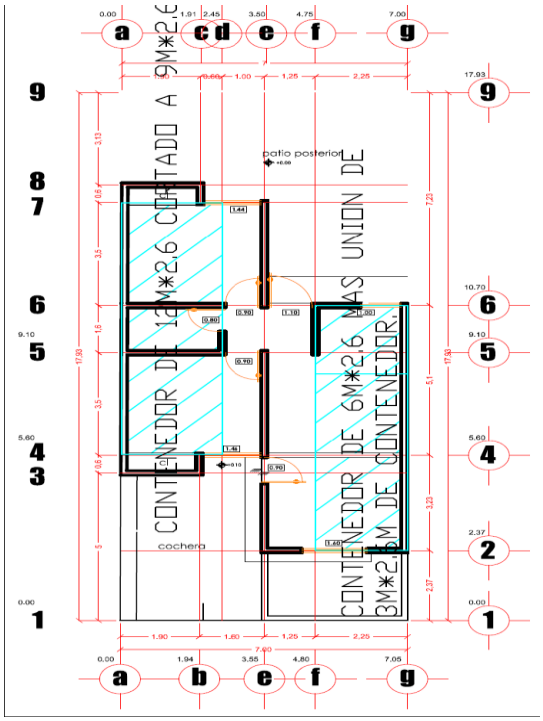



Fig. 2 Planta Arquitectónica de Vivienda A Base de Contenedores

## ANÁLISIS DE LOS RESULTADOS

### Sistema Constructivo Tradicional.

De acuerdo con el resultado obtenido por los generadores, el costo para la vivienda tradicional es $233,364.00 pesos mexicanos; en donde muros de mampostería es el valor más elevado, dentro de este sistema de constructivo.

### Sistema Constructivo a base de contenedores.

De acuerdo con el resultado obtenido por los generadores, el costo para la vivienda a base de contenedores es $248,876.00 pesos mexicanos; en donde soldadura, vigas y postes es el valor más elevado, dentro de este sistema de constructivo.

Por tal hecho no es económicamente factible construir para este sistema de construcción propuesto, el desarrollo de la vivienda a base de contenedores; esto por la gran cantidad de cortes que se tienen que hacer al contenedor para lograr espacios habitables y adecuados para la ocupación familiar.

## REFERENCIAS


[1] Shipping container housing*;* Seatle, Washington,EU.;2010.

[2] 4housing Módulos habitables; productos a la venta*;* Buenos Aires, Ar.;2012.

[3] "SISTEMAS DE AUTOPRODUCCION DE LA VIVIENDA"
Elementos básicos para su conceptualización e impulso
Enrique Ortiz, UAM Xochimilco.

# Colofón

Esta obra consta de 50 ejemplares y se elaboró
durante los primeros días
del mes de mayo de 2012, en los talleres del
Instituto Tecnológico de Colima
Av. Tecnológico No. 1
Villa de Álvarez, Col., México
C.P. 28976
Se utilizaron las fuentes Arial 8, 9 y 12,
así como Arial Balck 24




# VI Verano de Formación de Jóvenes Investigadores

**SEGUNDA EDICIÓN**
Noviembre 2011

**INSTITUTO TECNOLÓGICO DE COLIMA**
Av. Tecnológico No. 1, Villa de Álvarez, Colima C.P. 28976, Colonia Liberación
Teléfono: (0052)3123129920 Fax: (0052)3123140683
http://www.itcolima.edu.mx

**Derechos Reservados © 2011**
Prohibida la reproducción parcial o total de esta obra, por cualquier medio, sin autorización escrita del autor.

**Impreso en México**

ISBN: 978-607-95775-0-6

2011

Instituto Tecnológico de Colima

Memoria de Proyectos

ISBN: 978-607-95775-0-6