



**UNIVERSIDAD MICHOACANA DE
SAN NICOLÁS DE HIDALGO**

FACULTAD DE INGENIERÍA ELÉCTRICA

***APLICACIÓN DE VISIÓN CON LABVIEW EN EL
SEGUIMIENTO DE LÍNEA PARA UN MÓVIL***

TESIS

QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN ELECTRÓNICA

PRESENTA:
CARLOS ALBERTO TREJO SERÁFICO

ASESOR:
**MAESTRO EN INGENIERIA ELECTRICA
SALVADOR RAMÍREZ ZAVALA**

MORELIA, MICHOACÁN, DICIEMBRE 2013.





Universidad Michoacana de San Nicolás de Hidalgo
Facultad de Ingeniería Eléctrica



Morelia, Michoacán, a 11 de Octubre de 2013

Oficio No.1079/2013

C. Carlos Alberto Trejo Seráfico
Pasante de Ingeniería Electrónica.
PRESENTE.

En atención a su solicitud de fecha 11 de Octubre de 2013, me permito comunicarle que se aprueba el **Tema de Tesis** propuesto para presentar Examen Recepcional, de la **Carrera de Ingeniería Electrónica**.

El tema de tesis aprobado es: **“Aplicación de Visión con Labview en el seguimiento de línea para un Móvil.”** el cual deberá desarrollarse bajo el siguiente índice:

- Capítulo 1.- Introducción.
- Capítulo 2.- Procesamiento digital de imágenes y descripción del software IMAQ.
- Capítulo 3.- Aplicación y uso de IMAQ visión en la búsqueda de objetos.
- Capítulo 4.- Descripción, armado y configuración del prototipo para el seguimiento de línea.
- Capítulo 5.- Funcionamiento y pruebas del prototipo seguidor de línea.
- Capítulo 6.- Conclusiones.
 - Referencias.
 - Apéndices.

Para tales efectos fungirá como asesor de su tesis el M. I. Salvador Ramírez Zavala.

ATENTAMENTE

Dr. Juan Anzures Marín
Secretario Académico de la
Facultad de Ingeniería Eléctrica.



JAM/cmj

Dedicatoria

A mis padres:

Antonio Trejo García y Ángeles Serásico Martínez

Que con su vasto amor, apoyo, comprensión y paciencia se convirtieron en el pilar motivacional para levantarme después de una caída y seguir adelante hasta alcanzar mis metas.

A mis hermanos:

Ulises y Amwar

Que son la admiración y fortaleza que me han acompañado en la búsqueda de mis sueños.

A mis amigos:

Por ser la extensión de mi inspiración y empuje del logro que hoy llevo a cabo.

A mi facultad:

Por brindarme el conocimiento a través de la enseñanza de mis maestros y compañeros.

Resumen

El procesamiento digital de imágenes como herramienta para manipular imágenes del mundo real ha tomado un gran auge en los últimos años gracias al crecimiento y los avances de la tecnología computacional. Como consecuencia softwares computacionales comerciales que son capaces de procesar y analizar imágenes de manera más precisa y sofisticada tienen grandes aplicaciones en diferentes áreas como los son la Medicina, Robótica, Astronomía, Geografía, Arqueología, Biología, Aplicaciones Industriales entre otras. Este trabajo tiene como propósito diseñar e implementar un prototipo móvil controlado por una computadora a través del software computacional *IMAQ Vision* el cual posee como herramienta de enfoque para este proyecto el procesamiento digital de imágenes. El prototipo será capaz de detectar, rastrear y seguir un patrón de línea a través de la incorporación de una webcam que actuará como sensor detector de las propiedades cromáticas del patrón a localizar. Los datos recabados por el software *IMAQ Vision* serán enviados a través de comunicación serial hacia un microcontrolador encargado de ejecutar las instrucciones óptimas del hardware para el correcto desplazamiento del móvil y la búsqueda del patrón de línea. El entorno de desarrollo elegido para interactuar con el usuario se efectuará mediante el lenguaje de programación gráfica llamado *LabVIEW* el cual está destinado al desarrollo de aplicaciones que involucren sistemas de análisis, presentación y adquisición de datos.

Contenido

Dedicatoria	iii
Resumen	iv
Contenido	v
Lista de Figuras	viii
Capítulo 1. Introducción	1
1.1. Antecedentes	1
1.2. Objetivos	1
1.3. Justificación	2
1.4. Metodología	2
1.5. Contenido de la tesis	3
Capítulo 2. Procesamiento Digital de Imágenes y Descripción del Software IMAQ	4
2.1. Introducción	4
2.2. Visualización de una imagen	4
2.3. Transferencia de datos entre sensor y display	5
2.3.1. Principio de funcionalidad	5
2.4. Propiedades de los CCDs que influyen en la calidad, análisis y procesamiento de imágenes	7
2.4.1. Sensibilidad y resolución	7
2.4.2. Ruido y “pixeles calientes”	8
2.4.3. Efecto Blooming	9
2.5. Guía al uso de IMAQ en LabVIEW	9
2.6. LabVIEW 2012	10
2.7. Driver para el manejo de cámaras USB en IMAQ	12
2.8. Resumen	14

Capítulo 3. Aplicación y uso de IMAQ Vision en la Búsqueda de Objetos	15
3.1. Introducción	15
3.2. Software de la PC para reconocimiento, búsqueda y transmisión de datos ..	16
3.2.1. Diagrama de bloques para reconocimiento, búsqueda y transmisión de datos	16
3.2.2. Panel de control para reconocimiento, búsqueda y transmisión de datos	17
3.3. Estructuras del diagrama de bloques	18
3.4. Diagrama de flujo del software para reconocimiento	18
3.4.1. Selección de cámara, inicio de sesión y almacenamiento de video	19
3.4.2. Adquisición continua de imágenes	20
3.4.3. Selección y extracción de patrón	20
3.4.4. Aprender patrón	22
3.4.5. Búsqueda, arreglo y transmisión de datos del patrón	23
3.4.6. Cierre de sesión y reset de memoria	29
3.5. Resumen	29
Capítulo 4. Descripción, armado y configuración del prototipo para el seguimiento de línea	30
4.1. Introducción	30
4.2. Webcam	30
4.3. LabVIEW PC	31
4.4. Puerto serial y alimentación 1	32
4.5. Microcontrolador	32
4.5.1. PWM 1 y 2	33
4.6. Control del PWM	34
4.7. Puente H1 y H2	34
4.8. Servomotor 1 y 2	35
4.9. Alimentación 2	36
4.10. Software del microcontrolador	36
4.11. Armado del prototipo para el seguimiento de línea	40

4.12. Montaje y acoplamiento del hardware	41
4.13. Configuración de parámetros del software IMAQ Vision	45
4.14. Inicialización del software prototipo seguidor de línea	47
4.15. Resumen	50
Capítulo 5. Funcionamiento y pruebas del prototipo seguidor de línea	51
5.1. Funcionamiento del prototipo seguidor de línea	51
5.2. Pruebas de desplazamiento del prototipo	52
5.2.1. Desplazamiento de avance	52
5.2.2. Desplazamiento a la derecha	54
5.2.3. Desplazamiento a la izquierda	56
5.2.4. Desplazamiento de retroceso	57
5.3. Resumen	59
Capítulo 6. Conclusiones	60
6.1. Introducción	60
6.2. Conclusiones generales	60
6.3. Trabajos futuros	62
Apéndices	
A. Código del microcontrolador	63
B. Diseño del circuito impreso y montaje de componentes	65
C. Circuito esquemático	66
Referencias	67

Lista de Figuras

2.1.	Pixel de matriz rectangular	4
2.2.	Sensor fotodetector	5
2.3.	Mecanismo de transferencia del CMOS	6
2.4.	CTE en función de magnitud del pulso	6
2.5.	Visualización de la función de transferencia por modulación	7
2.6.	Imagen de pixeles calientes	8
2.7.	Blooming causado por puntero laser	9
2.8.	Panel de trabajo de LabVIEW	11
2.9.	Liga de herramientas de Vision and Motion de LabVIEW	12
2.10.	Sub herramientas de IMAQ USB	13
3.1.	Etapas para reconocimiento, búsqueda y seguimiento de patrón	15
3.2.	Diagrama de bloques	16
3.3.	Panel de control	17
3.4.	Ciclo While Loop	18
3.5.	Ciclo Case Structure	18
3.6.	Diagrama de flujo para reconocimiento	19
3.7a	Diagrama de bloques para selección	20
3.7b	Panel de control para selección	20
3.8a	Diagrama de bloques de secuencia de video	20
3.8b	Display de video	20
3.9.	Diagrama de selección y extracción de patrón	21
3.10.	Get Last Event	21
3.11.	Extracción del patrón	22
3.12.	Aprender patrón	23
3.13.	Diagrama de bloques de búsqueda y transmisión de datos	23
3.14.	Iconos de interruptores booleanos para búsqueda de patrón	24
3.15.	Controles de interruptores booleanos para búsqueda de patrón	24
3.16.	Controles ajustadores de parámetros de color	24
3.17.	Control de ajuste para validación de coincidencias	25
3.18.	Indicador de coincidencias	25
3.19.	Datos del indicador de coincidencias	26
3.20.	Arreglo para extraer posición de coordenadas X	26
3.21.	Cuadro de ayuda para cadena de datos	26
3.22.	Extracción de posición y coordenada	27
3.23.	Arreglo para ajuste de coordenadas en eje X	27
3.24.	Convertidor numérico a carácter	27
3.25.	Indicadores de cadena de caracteres	28
3.26.	Controles de configuración del puerto serial	28
3.27.	Diagrama de bloques para cierre y reset	29
4.1.	Diagrama de bloques del seguidor de línea	30
4.2.	Webcam USB	31
4.3.	Computadora Toshiba Satellite	31
4.4.	Puerto serie virtual	32

4.5.	Arquitectura del microcontrolador	33
4.6.	Empaquetado del microcontrolador	33
4.7.	Diagrama de bloques y tabla de verdad	34
4.8.	Puente H	35
4.9.	Servomotor	35
4.10.	Batería recargable	36
4.11.	Diagrama de flujo del programa principal	37
4.12a	Vista general de las conexiones del dispositivo	40
4.12b.	Conexiones de los puente H para M1 y M2 del dispositivo	41
4.13.	Dimensiones de la base para acoplamiento y montaje	41
4.14.	Vista inferior del acoplamiento de servomotores	42
4.15.	Vista lateral del montaje de la webcam	42
4.16.	Vista frontal del montaje de la webcam	42
4.17.	Vista superior del montaje de componentes	43
4.18.	Montaje y ubicación de webcam, circuito impreso y batería	44
4.19.	Montaje y acoplamiento completo del hardware	44
4.20.	Detección del puerto serie virtual	45
4.21.	Configuración de los controles del puerto serial	45
4.22.	Configuración de los controles de color	46
4.23.	Indicador de webcam disponibles	46
4.24.	Selector de webcam	46
4.25.	Patrón de línea a detectar	47
4.26.	Posicionamiento del prototipo seguidor de línea	47
4.27.	Visualización de la línea a detectar	48
4.28.	Elección del patrón a detectar	48
4.29.	Visualización del patrón a detectar	49
4.30.	Activación del control para aprender patrón	49
4.31.	Activación del control para búsqueda continua del patrón	49
4.32.	Indicador de coincidencias	50
5.1.	División de movimientos en el display de video	51
5.2.	Indicadores de posicionamiento en el eje X	52
5.3.	Rango y posición para desplazamiento de avance	53
5.4.	Activación de pines para desplazamiento de avance	53
5.5.	Secuencia de avance del prototipo seguidor de línea	54
5.6.	Rango y posición para desplazamiento a la derecha	54
5.7.	Activación de pines para desplazamiento a la derecha	55
5.8.	Secuencia de desplazamiento a la derecha del prototipo seguidor de línea	55
5.9.	Rango y posición para desplazamiento a la izquierda	56
5.10.	Activación de pines para desplazamiento a la izquierda	57
5.11.	Secuencia de desplazamiento a la izquierda del prototipo seguidor de línea	57
5.12.	Rango y posición para desplazamiento de retroceso	58
5.13.	Activación de pines para desplazamiento de retroceso	58
5.14.	Secuencia de retroceso del prototipo seguidor de línea	59

Capítulo 1

Introducción

1.1 Antecedentes

Durante milenios el hombre ha creado herramientas que con un largo proceso de perfeccionamiento se han ido modificando hasta obtener herramientas más cómodas y eficaces. Posteriormente ha creado máquinas que se encargan de realizar las duras tareas manuales. En el trabajo mecánico, el hombre ha pasado a trabajar como operario y a controlar el sistema, dejando a las máquinas las funciones de motor. En la actualidad ha creado sistemas automáticos. En el trabajo automático, el hombre se ha encargado de supervisar el sistema. El resto de tareas se realizan sin intervención humana mediante robots que trabajan con mayor eficiencia. Gran parte de las capacidades de un robot va de la mano con el tipo de sensor que este tenga ya que constituyen el sistema de percepción que facilita la información del mundo real para que los robots la interpreten. Dicho esto, el presente trabajo va encaminado al procesamiento de imágenes a través de una cámara que actuará como sensor del tipo lumínico ya que es muy útil para la identificación de objetos captando la intensidad luminosa de los mismos. Este tipo de sensores son parte de lo que se llama visión artificial y que en numerosas ocasiones están conformadas por cámaras tal como se propone en este proyecto; por medio de la computadora que tiene la versatilidad de reprogramar mediante un entorno gráfico amigable, junto con el software (*IMAQ LabVIEW*) instalado en la computadora.

1.2 Objetivos

Rastrear y ubicar patrones de color para reconocimiento y procesamiento de imágenes utilizando herramienta *IMAQ* de *LabVIEW* que servirá de apoyo en el tratamiento de la imagen a manera de resaltar sólo la intensidad cromática de los objetos o formas a detectar aplicada en la búsqueda, reconocimiento y seguimiento de línea que realizará un robot móvil a través de una webcam que actuará como sensor del campo de visión a tratar, además del control del movimiento dependiente del envío de datos del software con una

interfaz serial hacia un microcontrolador encargado de realizar las instrucciones necesarias para el control del hardware.

1.3 Justificación

El procesamiento de imágenes está revolucionando de manera rápida la tecnología hoy en día. El decremento en el precio de cámaras digitales, software de procesamiento de imágenes y el crecimiento en el área de aplicaciones de análisis hacen de este tipo de tecnología una de las más atractivas y redituables en diferentes áreas como lo son la industria, la computación, la robótica entre muchas otras y que solamente años atrás entraban en la categoría de inaccesibles para la mayoría de las personas que requerían un mejor análisis y que para ello utilizaban sensores que excedían el precio ajustable al bolsillo del consumidor. Entre las ventajas de usar una cámara como un “sensor universal” y el software apropiado para el análisis y procesamiento de imágenes se encuentran:

- Una mayor flexibilidad para reconfiguraciones de software
- Fácil obtención de un sensor más barato que uno de aplicación convencional
- Fácil programación con moderna y amigables herramientas de software

Es por ello que este trabajo se basa en el procesamiento de imágenes a través de una vía alterna en la capacidad sensitiva del dispositivo además del uso de un software con herramienta moderna con entornos gráficos que hacen de ella una forma más fácil de manejar.

1.4 Metodología

Para la realización de este trabajo primeramente se sustenta la base teórica de la investigación a través de fuentes informáticas que describirán las bases y definiciones principales del procesamiento digital de imágenes [1]. Posteriormente se analiza la herramienta proporcionada por el software a utilizar mediante el estudio de los manuales de referencia de *IMAQ Vision* con la cual se procede a rastrear, detectar y localizar un patrón de color determinado de una imagen [2][3]. A través de la aplicación de ciertos módulos de procesamiento digital de imágenes descritos en los manuales se desarrolla un programa

capaz de detectar un patrón de línea en el ambiente de programación gráfica de *LabVIEW*, una vez realizada la detección de patrón se establece el medio de comunicación por el cual se enviarán los datos recabados a un microcontrolador encargado de manipular y traducir dicha información en la activación y el control de giro de dos servomotores. Se armará y ensamblará un prototipo móvil que contendrá el dispositivo sensor de imagen, software y hardware para el óptimo desplazamiento del seguidor de línea. Se pondrá en práctica de campo experimental la búsqueda y detección del patrón de línea por parte del prototipo móvil bajo ciertas condiciones que resalten su funcionamiento y arrojen datos pertinentes para conclusiones del trabajo.

1.5 Contenido de la tesis

En el primer capítulo se presenta la introducción a este trabajo así como las ventajas del procesamiento digital de imágenes. Se describe el objetivo, justificación y la metodología de la investigación. En el capítulo dos se da una introducción al procesamiento digital de imágenes así como bases para el entendimiento de la transferencia de información entre el sensor y el display para la visualización de imágenes que servirán de apoyo en la búsqueda de patrón. Se presentan los pasos de instalación y guía de uso de la herramienta *IMAQ* de *LabVIEW* para la obtención y manipulación de imágenes. En el capítulo tres se presenta el modelo de trabajo en el que se aplica el software detector de patrones de color, etapas que lo conforman así como los diagrama de bloques de *LabVIEW* mediante el cual se captura y procesa la imagen para la detección de línea que seguirá el dispositivo móvil; se describe también el programa encargado del envío de datos mediante comunicación serial entre *LabVIEW* y el microcontrolador que ejecuta las operaciones mecánicas para búsqueda del patrón de línea; la función de cada dispositivo, partes y operación que constituyen el hardware del dispositivo móvil detector de línea. En el capítulo cuarto se describe el material para el armado del prototipo, se explican los pasos a seguir para su activación y configuración. Se establecen los parámetros óptimos para su inicialización y se describen los diferentes tipos de movimientos que dirigen al prototipo en su objetivo de localización y seguimiento de línea. En el quinto y último capítulo se establecen las conclusiones generales de esta tesis así como una propuesta de trabajos futuros con este software.

Capítulo 2

Procesamiento digital de imágenes y descripción del software IMAQ

2.1 Introducción

El campo de procesamiento digital de imágenes (PDI) se refiere a procesar o manipular las imágenes del mundo real de manera digital por medio de una computadora.

El interés por el PDI se basa principalmente en dos áreas de aplicación:

- Mejoramiento de la información pictórica para la interpretación humana.
- Procesamiento de datos de la imagen para su almacenamiento, transmisión y representación para manipulación autónoma de máquinas.

2.2 Visualización de una imagen

Naturalmente se está interesado en la definición de imagen que es esencial para el procesamiento de imágenes. Una forma de definir una imagen es una matriz rectangular (llamada imagen matriz) como se muestra en la Figura 2.1.

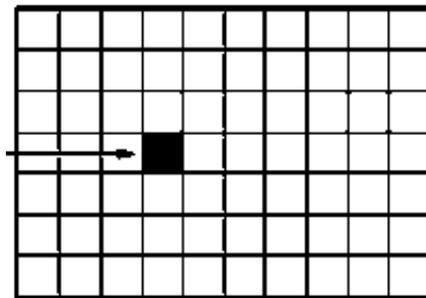


Figura 2.1 Pixel de matriz rectangular.

Esta se compone de filas y columnas de imagen. Un valor de línea junto con un valor de columna define una pequeña imagen llamada pixel a la cual le es asignado un valor representando el brillo del pixel.

2.3 Transferencia de datos entre sensor y display

Obviamente no es posible conectar cada pixel del sensor (cámara) a cada pixel del display en el mayor de los casos un monitor de computadora. Por ejemplo en el caso de un monitor de 320 x 240 pixeles, una conexión directa requeriría $320 \times 240 = 76,800$ cables entre el sensor y la PC. Para resolver este problema se utilizan los CCD (*charge coupled device*) que son sensores de imagen que consisten en un determinado número de foto-detectores, uno por cada pixel. Cada foto-detector capta una cierta cantidad de brillo correspondiente para cada pixel, este valor es transferido verticalmente a cada fila hacia abajo. Por supuesto el brillo no puede ser transferido de un foto-detector a otro y para ello se necesita otro componente, así que cada foto detector es conectado a un CMOS en combinación con una compuerta de transferencia como se observa en la Figura 2.2.

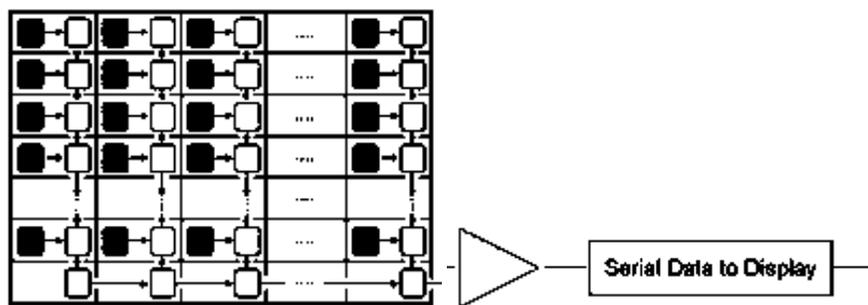


Figura 2.2 Sensor foto-detector.

2.3.1 Principio de funcionalidad

Los CMOS (Capacitor semiconductor de oxido Metálico) almacenan un cierto número de electrones que son previamente generados por su correspondiente foto detector por un cierto tiempo. La corriente generada está determinada por la ecuación:

$$I_{ph} = S \phi e \quad (1.1)$$

Donde:

I_{ph} Corriente del foto detector.

S Sensibilidad a la luz dependiendo de la longitud de onda de la luz.

ϕe Radiación de luz.

El mecanismo de transferencia de una celda CCD a otra se muestra en la Figura 2.3. Si el nivel de voltaje V_{G1} en la compuerta del CMOS que contiene la carga eléctrica Q es mayor que el nivel de voltaje V_{G2} en la compuerta del siguiente CMOS, la carga se mantiene en el condensador 1 (fase 1). Si V_{G1} cambia de mayor a menor y V_{G2} cambia de menor a mayor los electrones son transferidos.

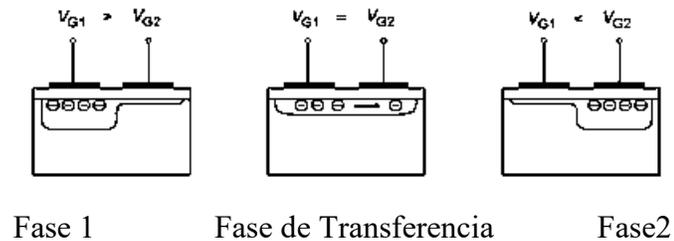


Figura 2.3 Mecanismo de transferencia del CMOS.

Lamentablemente algunos de los electrones se pierden durante el proceso de transferencia, este valor es definido por la eficiencia de transferencia de carga (CTE), este valor define el porcentaje de carga que se transmite en un solo proceso de transferencia, debido a la acumulación de este valor el CTE debe ser significativamente más alto que el 99.99%. La Figura 2.4 muestra la CTE para una gamma de CMOS con una compuerta de 4 mm de longitud, dependiendo de la duración de pulso de la señal de control, en este caso la duración de pulso debe ser de al menos 1.5 ns con el fin de garantizar un valor satisfactorio para el CTE.

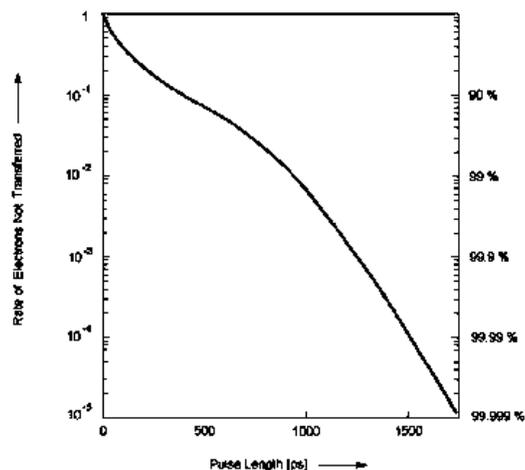


Figura 2.4 CTE en función de magnitud del pulso.

2.4 Propiedades de los CCDs que influyen en la calidad, análisis y procesamiento de imágenes

2.4.1 Sensibilidad y resolución

La *sensibilidad espectral* de un CCD es en general determinada por los foto-elementos, para obtener una máxima sensibilidad se pueden utilizar diversas capas de filtros en diferentes regiones espectrales; esta es la forma en que el color de los CCDs se realizan: por medio de tres foto elementos para cada píxel y filtros de color rojo, verde y azul. La *sensibilidad infrarroja* puede ser obtenida del foto-elemento *Schottky* que es usado en CCDs. Ellos pueden detectar formas de onda de hasta $5.7n\mu$ y diferencias de temperatura a 0.02°C . Cámaras de sensibilidad infrarroja pueden ser utilizadas por ejemplo en aplicaciones militares y médicas. La *capacidad de resolución* de un CCD puede ser descrita por la función de transferencia de modulación o MTF la cual se define como la modulación de la señal de salida en la que el sensor CCD registra una imagen que consta de rayas blancas y negras. Si d es la distancia entre dos píxeles del sensor y ns el número de bandas pares en blanco y negro, el producto $d ns$ proporciona una medida de la capacidad de resolución, si $ns d \ll 1$, la resolución de las rayas en blanco y negro es muy fácil. Para $d ns > 0,5$, no es posible distinguir entre dos líneas de vecinos. La función puede ser descrita matemáticamente por:

$$\text{MTF} = \frac{\sin(\pi ns d)}{\pi ns d} \quad (1.2)$$

En la Figura 2.5 es fácil ver que al menos dos sensores de píxeles son requeridos para detectar un par de bandas blanco y negro.

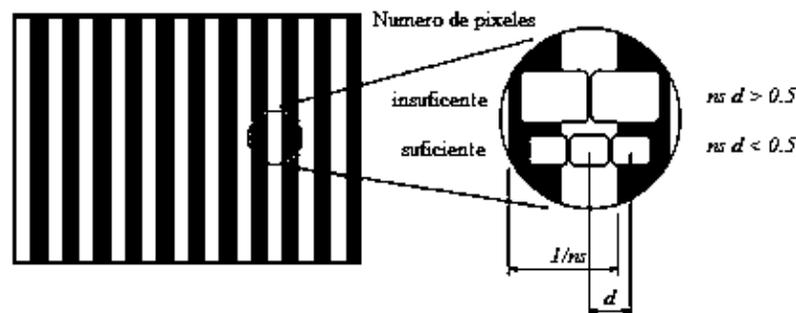


Figura 2.5 Visualización de la función de transferencia por modulación.

El índice 0 de (1.2) indica que esta función es teórica. El valor del MTF puede ser modificado por otros ciertos factores como lo son el CTE, la difracción o dispersión de luz.

2.4.2 Ruido y “pixeles calientes”

El ruido generado en un CCD es descrito por la radio señal-ruido (SNR).

$$SNR = \frac{n_{señal}}{nruido} \quad (1.3)$$

Donde:

$n_{señal}$ número de electrones que contribuyen a la señal de imagen.

$nruido$ número de “electrones de ruido”

El ruido estático que puede ser dividido en:

- Ruido fotón dado por el número de señal-electrón $n_{señal}$.
- Ruido CCD causado por los electrones generados en la celda CCD.
- Ruido amplificador el cual es generado en la salida del amplificador.

Un tipo de ruido especial que ocurre en la mayoría de las cámaras CCDs es causado por los *pixeles calientes*. Estos tienen sus orígenes en pequeñas contaminaciones o fallas en el área sensitiva de los CCDs que leen una alta pérdida de carga en la respectiva celda durante una cierta cantidad de tiempo. Esto a veces ocurre cuando se expone el lente de la cámara en un cuarto de total oscuridad y aparecen puntos blancos en la imagen como se observa en la Figura 2.6.



Figura. 2.6 Imagen de pixeles calientes.

2.4.3 Efecto blooming

El efecto del *blooming* ocurre especialmente en sensores de imagen de CCD y es causado por una sobrecarga de electrones en las celdas CCD. Para el observador el resultado es un incremento de la intensidad de luz en ciertas áreas de la imagen. La Figura 2.7 muestra este efecto usando un puntero laser sobre el sensor CCD.

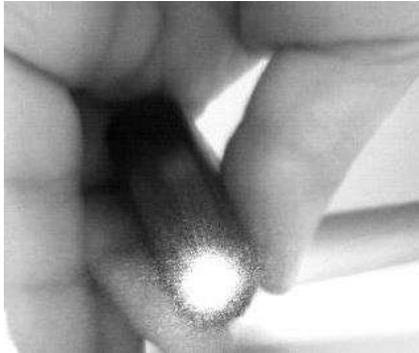


Figura 2.7 *Blooming* causado por puntero laser.

Modernos CCDs tienen un cierto número de mecanismos que ayudan a inhibir el efecto *blooming* entre los cuales se encuentran:

Compuertas Antiblooming que están localizadas horizontal o verticalmente al lado de las celdas CCD. Si ellas llevan el respectivo voltaje, los electrones no pueden ser transferidas a las celdas vecinas, pero si son capturadas por la compuerta *antiblooming*.

Reloj Antiblooming usa la recombinación natural de los electrones con “agujeros” en el semiconductor. Los agujeros son generados por una determinada estructura y frecuencia del reloj.

2.5 Guía al uso de IMAQ en LabVIEW

En la actualidad el reconocimiento de imágenes es una herramienta de gran utilidad en el área de control y automatización. Varias empresas usan el reconocimiento de imágenes en procesos de control de calidad donde intervienen defectos y selección de productos. Realizar un buen reconocimiento de imágenes requiere de un complejo sistema de procesamiento digital, que puede ser extremadamente difícil de implementar.

De acuerdo con el propósito de esta tesis se emplea el reconocimiento de imágenes para la determinación de la posición de objetos a través de programas como *LabVIEW* que presentan herramientas de procesamiento de imágenes que facilitan la creación de aplicaciones inteligentes de visión y reconocimiento. *IMAQ Vision* es una librería para *LabVIEW* que permite implementar aplicaciones inteligentes de imagen y visión, está diseñado para ser fácil de utilizar, construido específicamente para satisfacer las necesidades de los desarrolladores de aplicaciones de imágenes, también para reducir costos y tiempo, transparente gestión de memoria, llamados a VIs (virtual instruments) los cuales son programas o subrutinas que hacen de *IMAQ Vision* fácil de usar y aprender.

IMAQ Vision incluye un amplio conjunto de funciones:

- Optimización para escala de grises.
- Visualización de imágenes a color o binarias.
- Procesamiento de imágenes (estadísticas, filtrado, transformaciones geométricas).
- Calibración.
- Medición.
- Análisis.

El software necesario para la ejecución de la utilería de *IMAQ Vision* para el manejo de cámaras del tipo USB es el siguiente:

- LabVIEW 2012
- *Driver* para cámaras USB

2.6 LabVIEW 2012

LabVIEW es un lenguaje de programación de alto nivel, de tipo gráfico, y enfocado al uso en instrumentación. Pero como lenguaje de programación, debido a que cuenta con todas las estructuras, puede ser usado para elaborar cualquier algoritmo que se desee, en cualquier aplicación, como en análisis, telemática, juegos, manejo de textos, etc. Cada programa realizado en *LabVIEW* es llamado Instrumento Virtual (VI), el cual como cualquier otro ocupa espacio en la memoria del computador.

Requerimientos

Como la plataforma más usada en este medio son los PC, en términos de los mismos, lo mínimo para utilizar *LabVIEW*, es:

Un micro 386 con coprocesador. Como se requieren muchas operaciones de punto flotante, es indispensable el coprocesador. Los modelos a partir del 486Dx2 en adelante vienen con el coprocesador incluido en sí mismos.

Por uso de memoria, se recomienda usar 8 megas de RAM mínimo.

Si se usa un Demo con 2 megas en disco duro basta. Para el paquete completo es bueno disponer entre 40 y 50 megas de espacio en disco duro.

Como se aprecia el requerimiento es alto, pero hoy en día es posible conseguir un computador de este tipo a un precio mínimo, y en descenso día a día.

Después de haber instalado exitosamente y ejecutado *LabVIEW*, existe un grupo de iconos correspondientes en el panel de diagrama de bloques que aparece dando click derecho al mouse como se muestra en la Figura 2.8.

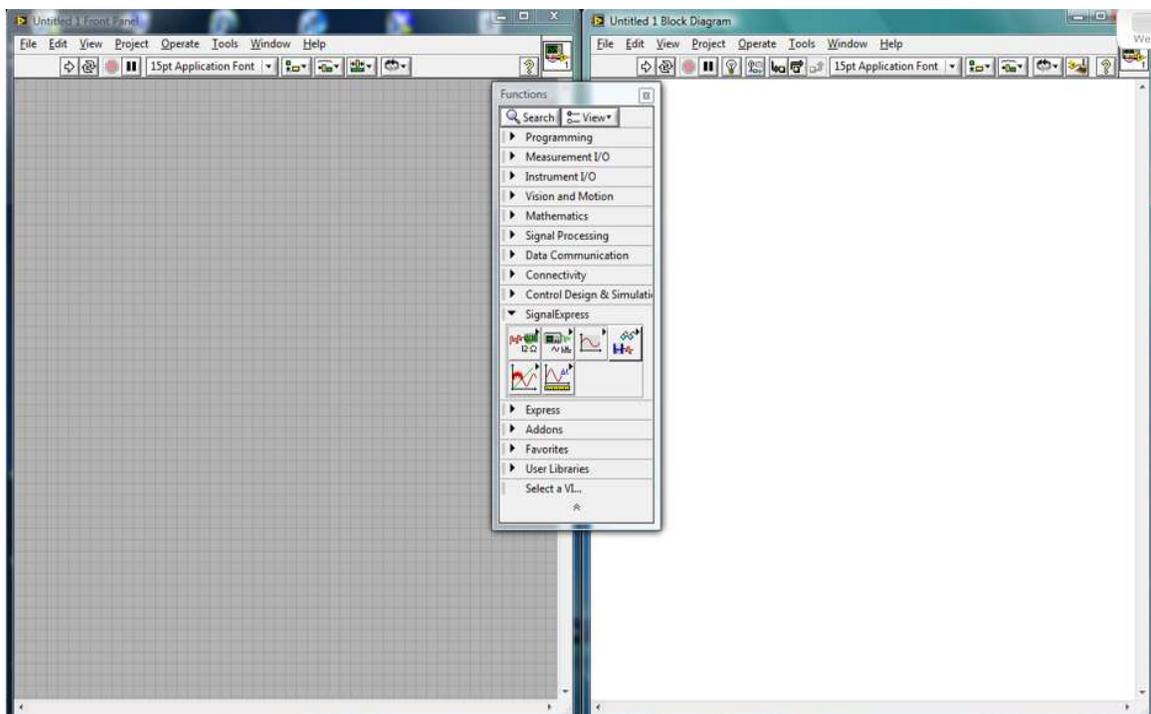


Figura 2.8 Panel de trabajo de *LabVIEW*.

Para el uso a la herramienta de *Vision* se requiere del uso de licencias contenidas en los discos de *LabVIEW 2012* así como sus respectivos drivers. Ya instalados aparece la liga de *Vision and Motion* mostrada en la Figura.2.9. De la cual sólo se hace uso de *Vision* que es la librería que contiene la herramienta necesaria para el procesamiento y análisis de imágenes.



Figura 2.9 Liga para herramienta de *Vision and Motion* de *LabVIEW*.

Los iconos mostrados en esta librería se explican con detalle en la siguiente sección, pero antes de utilizar la librería es necesario instalar el driver para USB que permite adquirir imágenes de una *Webcam*.

2.7 Driver para el manejo de cámaras USB en IMAQ

Requerimientos

- Windows Microsoft Windows Vista/Vista x64/XP/2000/Windows 7
- LabView 7.0 o mayor.
- Vision Assistant 8.0 o mayor, para USB Express VI support.

La liga disponible para descargar el driver del USB que permite conectar a los puertos USB del ordenador una *Webcam* se muestra en la página de *National Instruments* [4]. Este driver adquiere imágenes a través de la *Webcam* que actúa como sensor en la búsqueda de objetivos. Cabe resaltar que dicho driver no es directamente compatible con la plataforma de *Windows 7* al momento de instalarse (excepto *Windows XP* y *Windows Vista*), para ello se tienen que seguir los siguientes pasos descritos en la misma página de *National Instruments* para su correcta instalación.

Algunas de las herramientas con las que cuenta el *IMAQ USB* se tienen:



USB Snap Adquiere imágenes de la cámara USB como si se tomase una foto.



USB Grab Adquiere imágenes durante una adquisición continua.



USB Stop Detiene una adquisición continua.



USB Enumerate Crea una lista de cámaras USB disponibles.



USB Init Crea una sesión USB para una cámara específica.



USB Close Cierra la sesión de la cámara que fue abierta usando USB Init.

2.8 Resumen

En el capítulo se hizo referencia al campo del procesamiento digital de imágenes como punto de partida para sentar las bases del entendimiento del software IMAQ Vision, se analizó el principio de funcionalidad en la transferencia de datos entre el sensor y el display así como los factores que afectan la calidad y análisis del procesamiento digital de imágenes, posteriormente se dio una introducción a la guía de uso del software IMAQ Vision en donde se detallan los requerimientos de instalación del software y dispositivo de cámara USB, también se analizó el uso de las ventanas de configuración, manipulación de los controles y herramientas proporcionados por el software para procesamiento de imágenes que sirven de ayuda complementaria para el entendimiento del software.

Capítulo 3

Aplicación y uso de IMAQ Vision en la búsqueda de objetos

3.1 Introducción

Se muestra de manera general en la Figura 3.1 las etapas mediante las cuales se pretende adquirir una imagen continua de un determinado patrón de línea para su posterior reconocimiento, búsqueda y seguimiento a través de una webcam la cual es conectada a una computadora que contiene el software de procesamiento digital *IMAQ Vision*, este se encarga de procesar las detecciones de coincidencias para su envío por puerto serial a un microcontrolador previamente programado cuya función es activar las salidas correspondientes del sentido de giro de dos servomotores encargados del óptimo desplazamiento del prototipo.

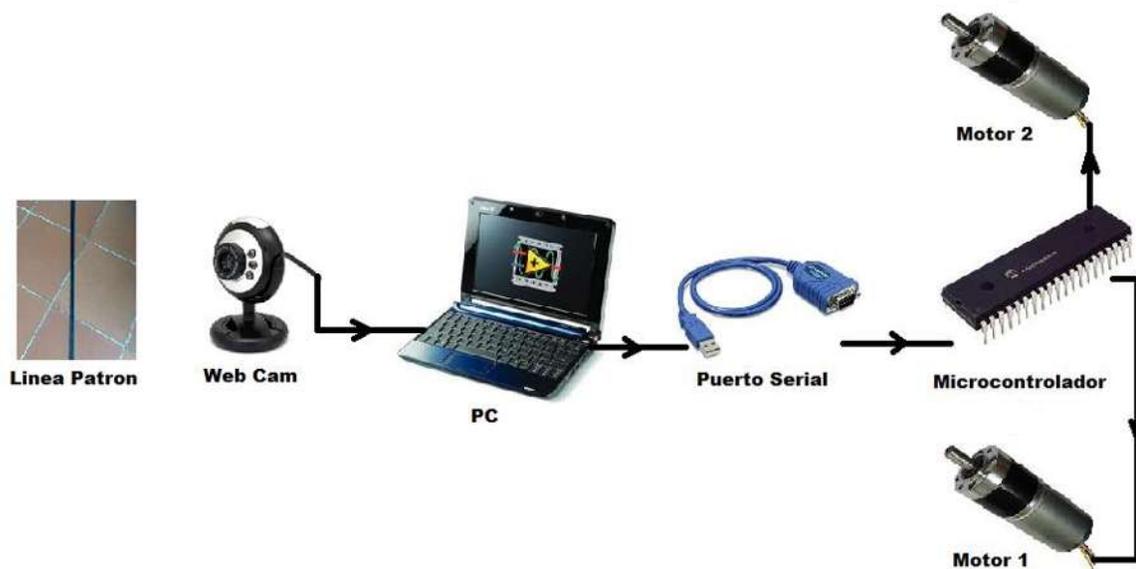


Figura.3.1 Etapas para reconocimiento, búsqueda y seguimiento de patrón.

3.2 Software de la PC para reconocimiento búsqueda y transmisión de datos

Se explica de manera detallada las partes que conforman el funcionamiento del software implementado para el reconocimiento búsqueda y transmisión de datos dentro del ambiente de programación de *LabVIEW* el cual se divide en dos diferentes áreas de trabajo, la primera es el diagrama de bloques donde se establecen las conexiones entre módulos y ciclos que conforman el programa y la segunda que es el panel de control en donde se visualizan, manipulan y configuran los controles e indicadores del programa implementado en el diagrama de bloques.

3.2.1 Diagrama de bloques para reconocimiento, búsqueda y transmisión de datos

El conjunto de ciclos y uniones de bloques que se muestra en la Figura 3.2 tiene la capacidad de reconocer y transmitir los datos necesarios para la búsqueda de objetos o formas en base al patrón de color por el cual este conformado. Se explicará con detalle en las siguientes secciones cada etapa en base a su correspondiente funcionalidad dentro del diagrama de bloques.

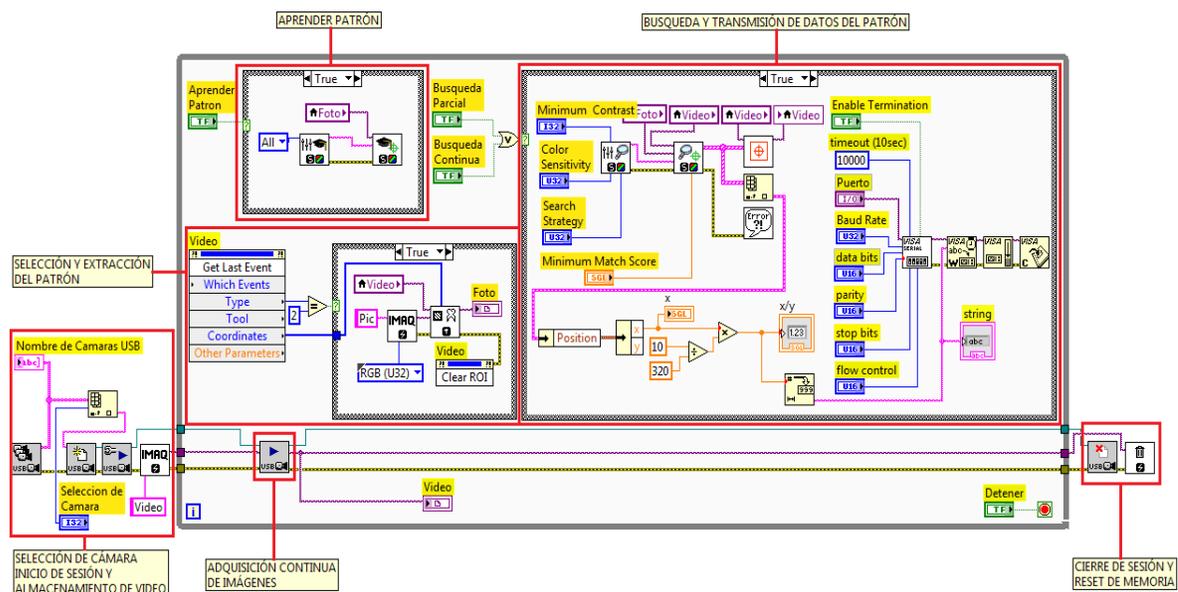


Figura.3.2 Diagrama de bloques.

3.2.2 Panel de control para reconocimiento, búsqueda y transmisión de datos

El panel de control diseñado que se muestra en la Figura 3.3 se puede observar el tipo de interfaz gráfica, sencilla y de fácil manejo para el usuario. El panel se encuentra conformado por los siguientes elementos:

1. Display de video.
2. Display de foto.
3. Display de cámaras USB.
4. Control selector de cámara USB.
5. Control de paro del programa.
6. Control para aprender patrón.
7. Control para búsqueda parcial.
8. Control para búsqueda continua.
9. Selector de puerto serie.
10. Control de bits de datos.
11. Control de terminación de carácter.
12. Selector del Baud Rate.
13. Control de bit de paridad.
14. Control de flujo.
15. Control del bit de paro.
16. Indicador de posición del patrón.
17. Indicador de carácter.
18. Indicador de posición arreglado.
19. Control para estrategia de búsqueda.
20. Control de valor mínimo de coincidencias.
21. Control de sensibilidad de color.
22. Control del contraste mínimo.

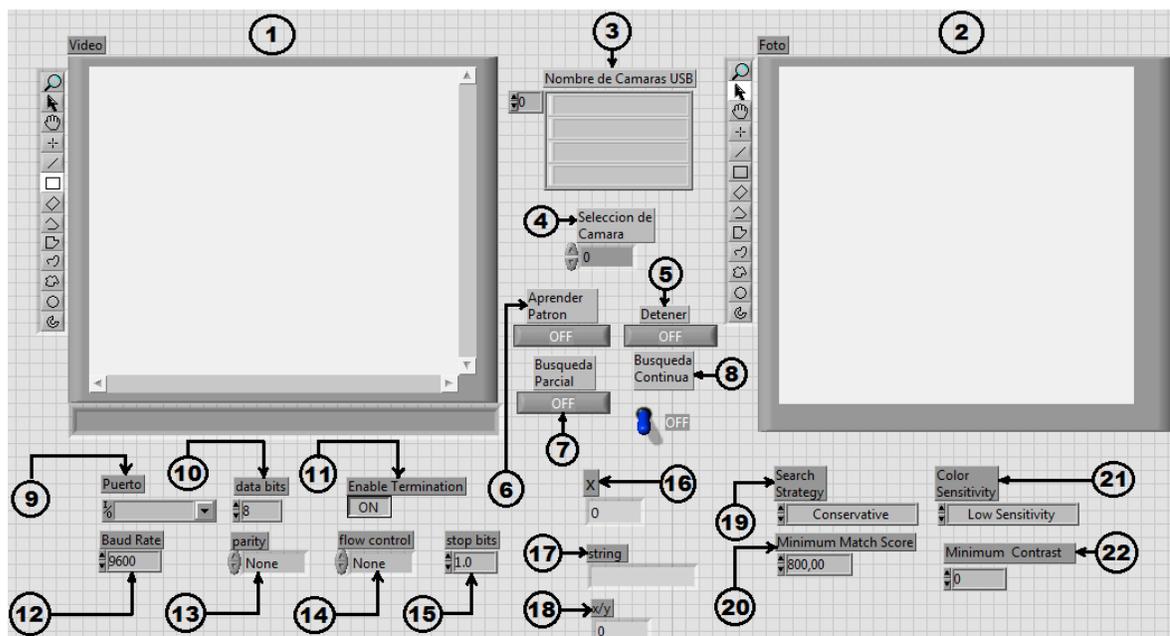


Figura 3.3 Panel de Control.

3.3 Estructuras del Diagrama de Bloques

El conjunto de etapas que se encuentran dentro de la estructura *While Loop* mostrada en la Figura 3.4, permite un corrimiento cíclico de cada etapa hasta que la terminal condicional de paro reciba un booleano que permita salir del ciclo.



Figura.3.4 Ciclo *While Loop*

La mayor parte de las etapas dentro del ciclo *While Loop* se encuentran a su vez dentro de otra estructura llamada *Case Structure* mostrada en la Figura 3.5, la cual se activa mediante la terminal booleana a la que se encuentre enlazada.



Figura.3.5 Ciclo *Case Structure*

3.4 Diagrama de Flujo del software para reconocimiento

En la Figura 3.6 se muestra el diagrama de bloques del software diseñado para el reconocimiento y procesamiento de imágenes en base a patrones de color utilizando IMAQ Vision. Se dará una explicación más detallada de cada bloque que integra este diagrama en las siguientes secciones del capítulo.

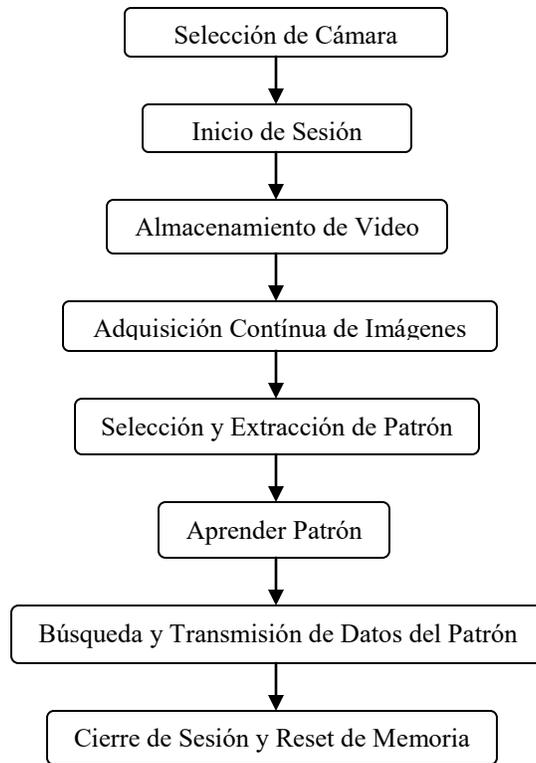


Figura.3.6 Diagrama de flujo para reconocimiento

3.4.1 Selección de cámara, inicio de sesión y almacenamiento de video

Esta estructura está conformada por diferentes tipos de bloques como se muestra en la Figura.3.7a, de los cuales algunos ya se mencionaron anteriormente, el primero crea una lista de las cámara USB disponibles  de las cuales solo una puede ser utilizada para la adquisición de imagen es por ello que se utiliza un bloque *Array*  el cual administra los caracteres de tipo cadena (color rosa) que lleva información sobre la cámara que se utilizara y son enviadas al siguiente bloque  el cual inicializa la sesión para la cámara USB que fue especificada anteriormente. El siguiente paso es configurar e inicializar una adquisición continua con el bloque . Después se procede a crear un espacio de memoria temporal para el video que se adquirirá, esto mediante los bloques *IMAQ Create* . En el entorno grafico de panel de control se visualiza el botón selector de cámara así como el display de las cámaras disponibles como se observa en la Figura 3.7b.

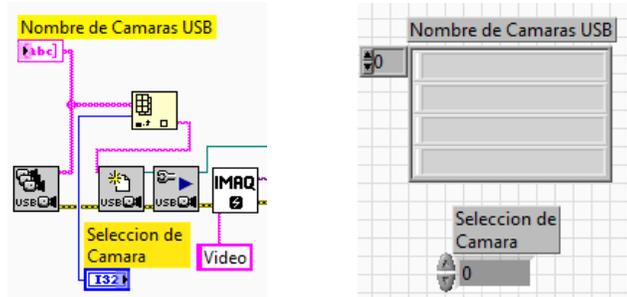


Figura 3.7 a) Diagrama de Bloques para selección, b) Panel de Control para selección

3.4.2 Adquisición Continua de Imágenes

Mediante USB grab  se puede realizar una adquisición continua de imágenes o dicho de otro modo se realiza una secuencia de video siempre y cuando esta se encuentre dentro del ciclo while; los datos de la secuencia de imágenes se envían al icono  para rastrear el patrón en tiempo real como se presenta en la Figura 3.8a. En el panel de control dicha adquisición es visualizada en el display llamado “video” mostrado en la Figura 3.8b el cual recibe los datos adquiridos del USB grab.

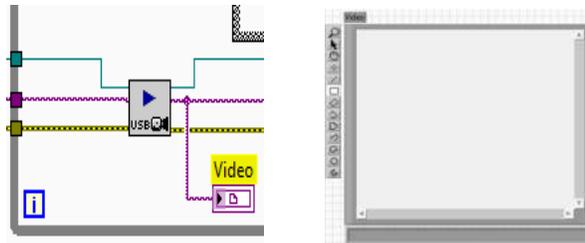


Figura.3.8 a) Diagrama de bloques de secuencia de video, b) Display de video.

3.4.3 Selección y extracción de Patrón

El propósito de esta etapa es señalar el área u objeto que queremos tomar como patrón para búsqueda. Como se muestra en la Figura 3.9 esto se realiza mediante un arreglo el cual tiene como función registrar la última acción realizada sobre el área del display de video, dicha acción será la de encerrar en una determinada área el objeto de búsqueda, extrayéndola y enviando la imagen al display de foto.

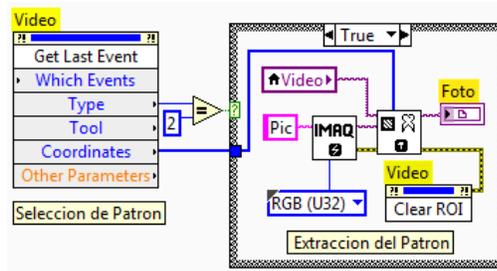


Figura.3.9 Diagrama de selección y extracción de patrón.

Selección de Patrón.

Mediante el menú de selección de video (*Get Last event*) mostrado en la Figura 3.10 se invoca un método que se encarga de enviar la información acerca del más reciente evento realizado sobre la imagen del display de video.

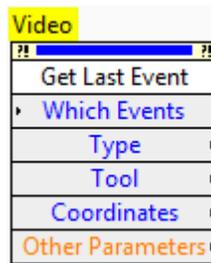


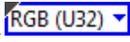
Figura.3.10 Get Last Event

Existen varios tipos de eventos que se pueden realizar sobre la imagen del display:

- 1 Click Event. El usuario hizo un click sobre la imagen del display.
- 2 Draw Event. El usuario hizo un dibujo sobre la imagen del display
- 4 Size Event. El usuario ha cambiado el tamaño de la imagen del display.
- 5 Scroll Event. El usuario ha movido la barra de desplazamiento de la imagen del display.
- 8 Double Click Event. El usuario ha hecho doble click sobre la imagen del display.

Para este trabajo se utiliza la opción (Draw Event) para señalar el área de interés en la que se encuentra el objeto a buscar y posteriormente enviarse las coordenadas de dicha área a la siguiente etapa. Como se muestra en la Figura. 3.9 se utiliza un comparador  que activa la etapa de extracción con la condición de que el indicador constante 2 sea igual al tipo de evento realizado sobre la imagen que para este caso debe ser únicamente el numero 2.

Extracción de Patrón.

Como se muestra en la Figura 3.11 la etapa de extracción consta de una locación temporal de memoria  donde se almacena la imagen del último evento realizado sobre la imagen, también se especifica el tipo de color RGB (*red,green,blue*) de 32 bits por pixel . La información se envía al modulo (IMAQ Extract)  encargado de extraer la imagen o patrón para ser visualizada en el display “foto” representada en el diagrama de bloques . Al realizarse la extracción se limpia la región de interés elegida por el usuario en el display de video  .La extracción de patrón depende de la condición booleana “true” del ciclo case que a su vez depende de que el usuario elija el área a comparar.

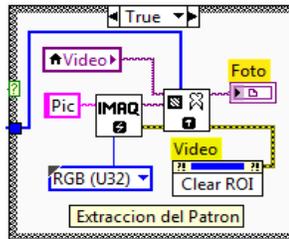


Figura 3.11 Extracción del Patrón.

3.4.4 Aprender Patrón

La etapa mostrada en la Figura 3.12 se encarga de establecer los parámetros necesarios para la comparación y reconocimiento de coincidencias en base a patrones de color entre la imagen visualizada en el display “foto” (elegida por el usuario) y la secuencia continua “video”. En el panel de control se encuentra el interruptor booleano  que permite inicializar esta etapa, cabe resaltar que antes de su inicialización es necesario que haya habido previamente una selección y extracción del área a detectar por el usuario la cual es mencionada en la sección 3.4.3. El modulo (setup learn color pattern)  establece los parámetros del patrón a través del modo de aprendizaje (learn mode)  donde se elige el modo (All) para extraer toda la información invariante tanto de desplazamiento como de rotación de las coincidencias en la imagen, los datos de los parámetros son enviados al

modulo (learn color pattern)  el cual crea una descripción de los colores de la imagen correspondiente al display “foto” 

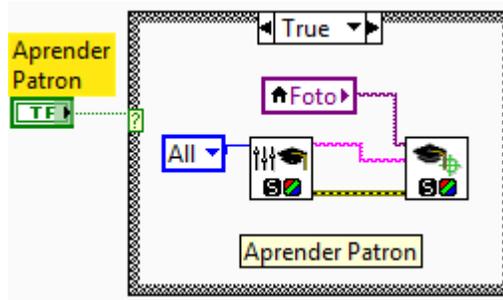


Figura 3.12 Aprender patrón.

3.4.5 Búsqueda, arreglo y transmisión de datos del patrón

Una de las etapas más importantes es la que se muestra en la Figura 3.13 donde se lleva la búsqueda del objeto a detectar, arreglo y envío de datos a través del puerto serie los cuales serán manipulados por un microcontrolador que activara los servomotores del prototipo móvil.

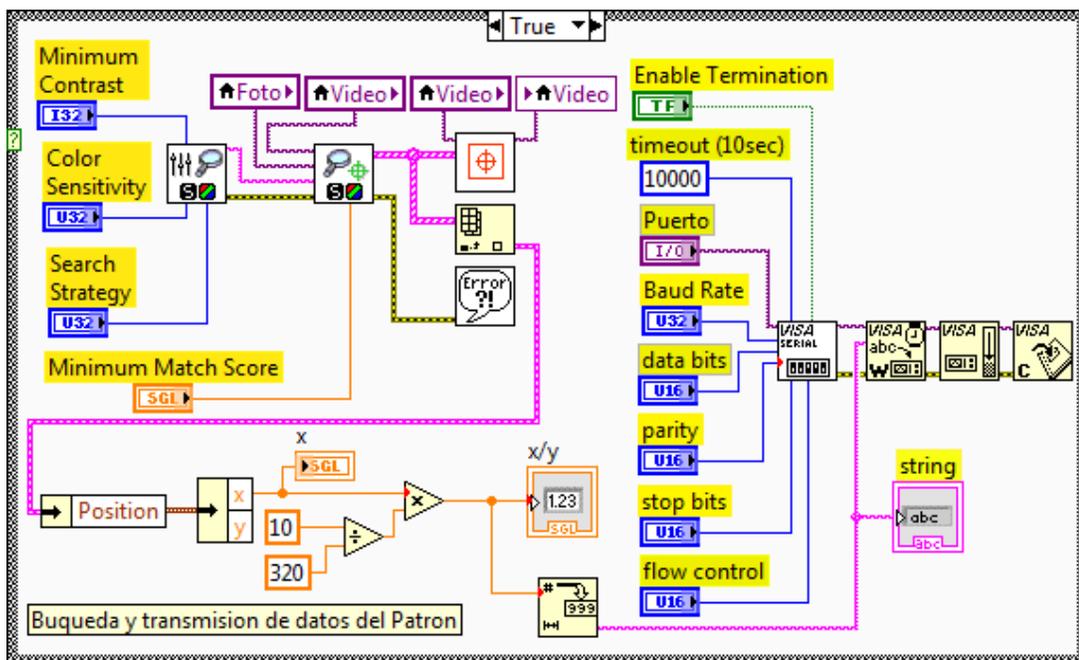


Figura 3.13 Diagrama de bloques de búsqueda y transmisión de datos.

El inicio de esta etapa se produce al momento de activar cualquiera de los interruptores booleanos mostrados en la Figura 3.14 que están conectados al comparador OR  con la diferencia de que el interruptor de búsqueda parcial solo reconoce el patrón a detectar en una sola imagen estática y el interruptor de búsqueda continua reconoce el patrón en forma real y continua a través de un video.

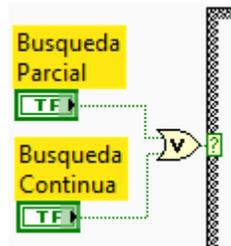


Figura 3.14 Iconos de interruptores booleanos para búsqueda de patrón.

Mismos interruptor y botón booleano mostrados en la Figura 3.15 se encuentran representados en el panel de control para un mejor manejo por parte del usuario en la búsqueda del objeto.



Figura 3.15 Controles de interruptores booleanos para búsqueda de patrón.

De manera similar a la etapa en la que se aprende la imagen también en esta es necesario hacer ajustes de parámetros para ello se utiliza el modulo ajustador de coincidencias de color  donde se puede ajustar el contraste mínimo, la sensibilidad de color y la estrategia de búsqueda los cuales también están en el panel de control mostrado en la Figura 3.16.

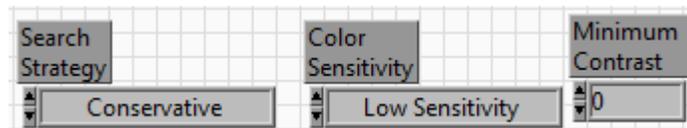


Figura 3.16 Controles ajustadores de parámetros de color.

El siguiente modulo  que fue previamente ajustado con su antecesor está encargado de buscar las coincidencias entre la imagen de patrón y la imagen de video, posee un control mínimo de coincidencias ajustable con un rango que va desde 0 hasta 1000. Para este trabajo el valor óptimo predeterminado de coincidencias como se muestra en la Figura 3.17 es de 800 ya que el ajuste máximo (1000) interpretaría demasiadas coincidencias dentro del rango deseado de búsqueda del patrón mientras el mínimo (0) no arrojaría coincidencias.

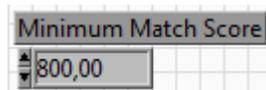


Figura 3.17 Control de ajuste para validación de coincidencias

Como se observa en la Figura 3.18 las coincidencias localizadas se reflejaran en un indicador visual  en el display de video. Como se observa en la barra de información se indican las dimensiones del display (320x240), el número de bits (1x32), el tipo de imagen (RGB) y las coordenadas XY del indicador de coincidencias en el área del display (209,174).



Figura 3.18 Indicador de coincidencias.

La cadena de coincidencias contiene diferentes tipos de datos entre los cuales se halla la posición, ángulo, escala, valor y delimitador del indicador de coincidencias en el display de video como se muestra en la Figura 3.19

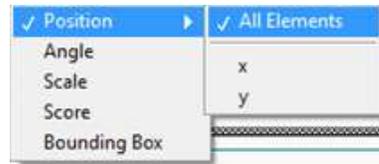


Figura 3.19 Datos del indicador de coincidencias.

Se hace uso y manipulación de las coordenadas en el eje X para ubicar y rastrear la posición del patrón a detectar, para ello se crea un arreglo mostrado en la Figura 3.20 en el cual se extrae solo la información de la posición contenida en la cadena de datos.

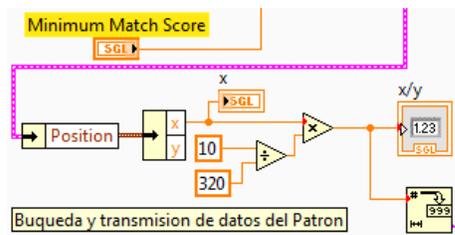


Figura 3.20 Arreglo para extraer posición de coordenadas X.

La información contenida en la cadena de datos puede ser visualizada en el cuadro de ayuda como se muestra en la Figura 3.21. Para acceder a ella se utilizan las teclas Ctrl+H.

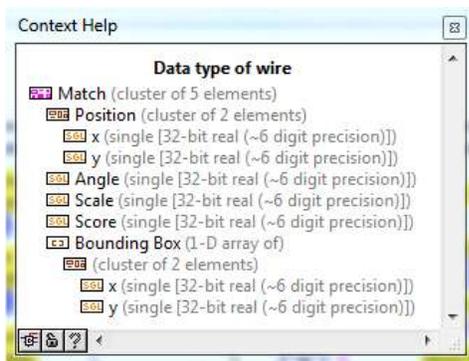


Figura 3.21 Cuadro de ayuda para cadena de datos.

Utilizando el elemento de la paleta de funciones llamado *unbundle by name* se desagrupa el contenido de la cadena especificando el nombre del elemento a manipular (posicion) de la cual a su vez se realizara otra extraccion para acceder al elemento de coordenadas X asi como se muestra en la Figura 3.22.

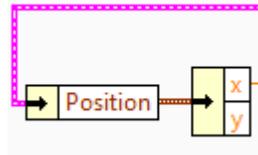


Figura 3.22 Extracción de posición y coordenada.

Como se menciono anteriormente, el display tiene una dimensión de 0 a 320 en el eje de las X, para su manipulación se hace un arreglo el cual es mostrado en la Figura 3.23, consiste en reducir sus dimensiones de 0 a 10 esto con el propósito de ajustar la información a la requerida por el modulo del puerto serial y también facilitar la manipulación de datos en la programación del microcontrolador que ejecutara las instrucciones para el hardware.

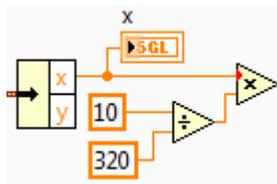


Figura 3.23 Arreglo para ajuste de coordenadas en eje x.

La cadena de datos que arroja la información de posición es del tipo numérico, esta debe ser convertida a una de cadena de caracteres la cual es requerida para la transmisión y recepción de la información a través del puerto serial. Para ello se hace uso de un convertidor numérico a cadena de carácter decimal. El convertidor a utilizar es mostrado en la Figura 3.24.

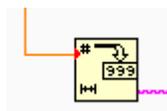


Figura 3.24. Convertidor numérico a carácter.

Los datos de la cadena de caracteres pueden ser visualizados en el panel de control debido a su respectivo indicador en el diagrama de bloques. Ambos son mostrados en la Figura 3.25.

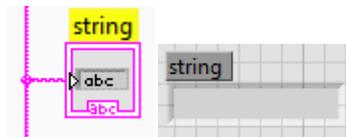


Figura 3.25. Indicadores de cadena de caracteres.

El modulo encargado del envío de datos a través del puerto serial es *Visa Write*  el cual tiene como función principal enviar una cadena de caracteres al microcontrolador en el momento en que se detecte una coincidencia de patrones de color o dicho de otro modo se active el indicador de coincidencias, los caracteres transmitidos a través del puerto serial llegaran al receptor del microcontrolador, donde son manipulados para su uso en el control y dirección de los servomotores utilizados en el desplazamiento del dispositivo móvil. El ajuste de parámetros para el correspondiente envío de datos es llevado a cabo por el modulo de configuración del puerto serial *Visa*  que es previo al *Visa Write*, este es conformado por terminales de ajuste tales como la selección del puerto al cual serán enviados los datos así como el relación de bits (baud rate) que determina la velocidad de transmisión de dichos datos, el numero de bits de datos que serán enviados así como el bit de paridad y el flujo de control, todos ellos representados en el panel de control para un mejor manejo de los mismos por el usuario como se muestra en la Figura 3.26.



Figura 3.26 Controles de configuración del puerto serial.

3.4.6 Cierre de sesión y reset de memoria

Esta estructura está conformada por dos tipos de bloques como se observa en la Figura.3.27; de los cuales algunos ya se mencionaron anteriormente, el primero cierra la sesión de la cámara USB seleccionada , después se borran los espacios de memoria ocupados por las imágenes adquiridas  tanto para la foto adquirida como para el patrón de la foto. Cabe resaltar que este parte solo se procederá si se cumple con la activación del paro del ciclo while que está ligado a la presencia de un error en las cadenas de bloque o la activación de la terminal booleana de paro por el usuario del software.

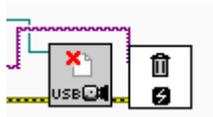


Figura 3.27 Diagrama de bloques para cierre y reset.

3.5 Resumen

Se explicaron a detalle las estructuras del programa encargado de la manipulación de datos a partir de la obtención de imágenes continuas por parte de la webcam y que son traducidas en información que es visualizada a través del entorno grafico del software y las etapas que lo conforman entre ellas las principales que son el reconocimiento, búsqueda y transmisión de datos, en conjunto todas las etapas mencionadas tienen la capacidad de reconocer el patrón de color indicado para ser después localizado mediante un display como indicador visual y finalmente transmitido a la etapa de control del hardware.

Capítulo 4

Descripción, armado y configuración del prototipo para el seguimiento de línea

4.1 Introducción

En el presente capítulo se explica cada etapa del dispositivo móvil representado en la Figura 4.1, donde el elemento maestro es el microcontrolador programable también se detalla el funcionamiento del programa encargado de controlar el sentido del giro y la velocidad de los servomotores en base a la información procesada por la PC en el software *IMAQ Vision* de *LabVIEW* que es enviada a través del puerto serial con el objetivo de corregir y redirigir el desplazamiento y la búsqueda del patrón a seguir por parte del seguidor de línea.

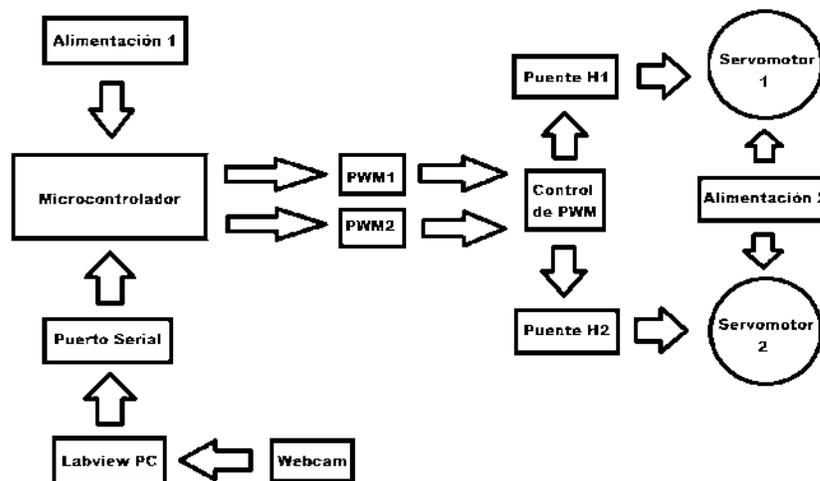


Figura 4.1 Diagrama de bloques del seguidor de línea.

4.2 Webcam.

La webcam es un dispositivo que se conecta a la computadora por medio del puerto USB y actúa como sensor para detectar una serie de imágenes del mundo exterior las cuales son manipuladas a través del software *IMAQ Vision* en la búsqueda de coincidencias en los

patrones de color. La webcam que se utiliza en el proyecto se muestra en la Figura 4.2, tiene una resolución de 3 mega pixeles, balance de blancos automático, compensación automática de color, interfaz USB 2.0, integrado a la webcam se encuentran seis leds blancos utilizados para iluminación nocturna.[5]



Figura 4.2 Webcam USB.

4.3 Labview PC.

La computadora portátil PC por sus siglas en ingles es el ordenador encargado de detectar y administrar los dispositivos conectados a los puertos con los que cuenta así como también redireccionar los datos obtenidos para su observación y manipulación a través del software ejecutado por la computadora e instalado en la memoria del disco duro. La PC a utilizar en este proyecto es de la marca *Toshiba* modelo *Satellite* mostrada en la Figura 4.3, consta de una memoria RAM de 2GB y un disco duro de 260GB del cual sólo 3 GB son utilizados para el software *IMAQ Vision*. Para uso del software en el seguidor de línea es necesario acoplar la computadora al dispositivo móvil ya que obligatoriamente el software requiere detectar una conexión de la webcam al ordenador por medio del puerto USB. También se hace uso de la fuente de energía eléctrica que proporciona la batería de la computadora que con una carga del 100% tiene una duración de dos horas en la alimentación de la webcam como también del puerto virtual del cual se habla más adelante.



Figura.4.3 Computadora Toshiba Satellite.

4.4 Puerto Serial y Alimentación 1

El puerto serie es el encargado de enviar los datos e instrucciones recabadas por el software a través del protocolo de comunicación RS232 para su recepción por parte del microcontrolador que se encarga de emitir instrucciones al hardware. Debido a que la computadora utilizada para este proyecto no cuenta con su propio puerto serial, se hace uso de un puerto serie virtual que es conectado a la computadora por USB Figura 4.4. Este puerto virtual cuenta con las terminales TX para transmisión de datos y RX para recepción además cuenta con las terminales de alimentación de 5v y GND que son utilizadas para energizar el microcontrolador y la etapa de control.[6]



Figura 4.4 Puerto Serie Virtual.

4.5 Microcontrolador.

El microcontrolador es un circuito programable capaz de realizar una tarea; encierra en un sólo chip CPU las memorias RAM y ROM, periféricos especiales y puertos de entrada y salida mostrados en la Figura 4.5. El microcontrolador usado en este proyecto es el PIC16F877A de la familia de microchip [7] cuyo empaquetado se muestra en la Figura 4.6. Se usa este microcontrolador por su facilidad de uso comodidad y rapidez en el desarrollo de aplicaciones, además de que existe abundante información, herramientas y software de libre disposición haciéndolo de los más populares entre los usuarios. Entre otras ventajas por las cuales se eligió usar el microcontrolador pic16F877A se mencionan las siguientes:

- Su fácil adquisición ya que se pueden conseguir en cualquier tienda de electrónica
- Por su bajo costo.
- Por su fácil aprendizaje ya que cuentan con el menor número de instrucciones.
- Por su set de instrucciones que no puede ser igualado por otro microcontrolador.
- Por la disponibilidad de herramientas en hardware y software.

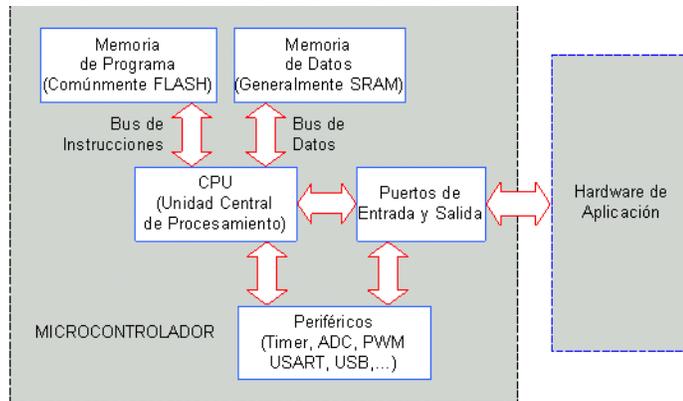


Figura 4.5 Arquitectura del microcontrolador.

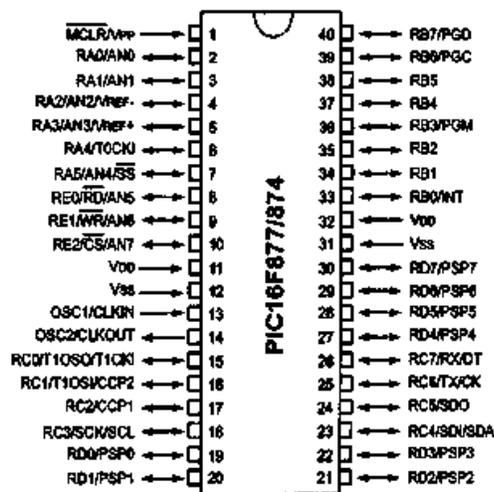


Figura 4.6 Empaquetado del microcontrolador.

4.5.1 PWM 1 y 2.

La modulación de ancho por pulso o PWM por sus siglas en ingles es la señal que se encarga de controlar la velocidad de los motores, el microcontrolador cuenta con dos módulos para generar este tipo de señal, los cuales son nombrados CCP1 y CCP2, cada modulo puede generar una onda PWM de hasta 10 bits de resolución con una frecuencia y ciclo de trabajo configurables, ambas ondas son semi independientes, esto es que pueden tener diferente ciclo de trabajo pero comparten la misma frecuencia, cada CCPx se puede habilitar o inhabilitar independientemente. La salida del modulo CCP1 es el pin RC2 y la del modulo CCP2 es el pin RC1 los cuales deben ser configurados como salidas.

4.6 Control de PWM

Esta etapa tiene como función principal la manipulación de las señales PWM que activan las respectivas entradas de los puentes H de cada motor y a su vez controlar el sentido del giro de cada uno, para ello se hace uso de un circuito integrado externo LM7408 el cual consta de un empaquetado de cuatro compuertas AND[8], cada compuerta se conecta a una de las cuatro salidas del puerto D del microcontrolador (RD0, RD1, RD2, RD3) dos de las compuertas a su vez se conectan a una misma señal PWM proveniente de los módulos CCP1 y CCP2. Por medio de programación se activan en alto o bajo los pines del puerto D con el fin de obtener a la salida de cada compuerta la señal PWM o la desactivación de la misma como se puede observar en el diagrama y la tabla de verdad de la Figura 4.7.

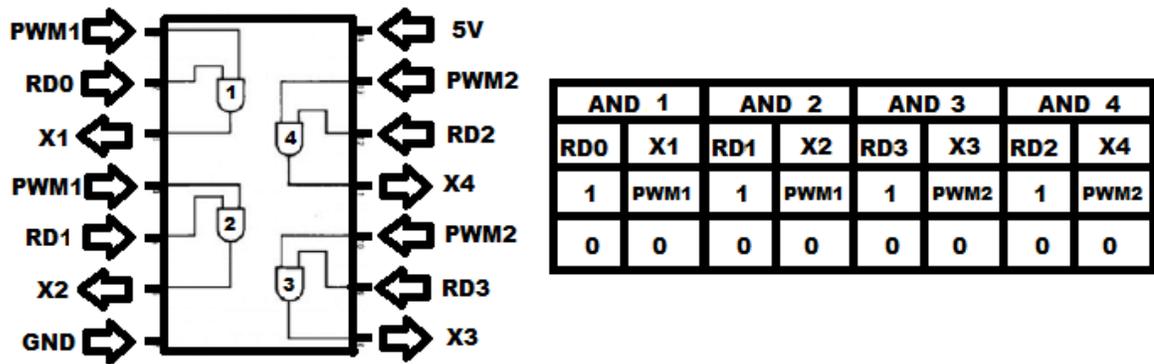


Figura 4.7 Diagrama de conexiones y tabla de verdad.

4.7 Puente H1 y H2

El puente H es un arreglo de componentes que en su mayoría se conforma por transistores de potencia como se muestra en la Figura 4.8, dicha distribución es una de las más utilizadas en la manipulación de motores de CC para la inversión del giro del motor. Para el proyecto se hace uso de dos de estas configuraciones una para cada motor a utilizar. Su funcionamiento radica en la aplicación de una señal positiva en la entrada marcada como avance la cual activa al transistor Q1 que a su vez activa el flujo de corriente a través de Q2 y Q5 haciendo que la terminal A del motor reciba voltaje y la terminal B la referencia; la misma analogía se presenta al activar la entrada marcada como retroceso con los transistores Q6, Q4 y Q3 con la diferencia que en este caso la terminal B recibe el voltaje y

la terminal A la referencia. Un punto importante que se debe tener en cuenta es que las señales de avance y retroceso jamás deben coincidir ya que si esto ocurre los transistores Q2, Q3, Q4 y Q5 cerrarían el circuito directamente entre el voltaje de la fuente y referencia sin pasar por el motor de modo que se excederá la capacidad de corriente emisor-colector de los transistores y se dañaran para siempre.

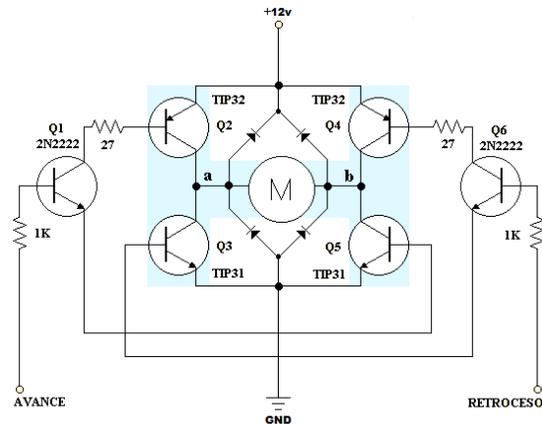


Figura 4.8 Puente H.

4.8 Servomotor 1 y 2

Un servomotor es un dispositivo similar a un motor de corriente directa que contiene un reductor de velocidad a base de engranes que se encargan de convertir gran parte de la velocidad del giro del motor en torque. Para el proyecto se hace uso de dos servomotores con un torque de 2kg/cm y engranaje de aluminio para desplazar el peso del dispositivo móvil a través de la trayectoria a seguir. Se acoplara al eje de cada servomotor una rueda de caucho la cual proporcionara agarre y adherencia al contacto con el piso para un mejor desplazamiento del dispositivo móvil. El servomotor a utilizar se muestra en la Figura 4.9.



Figura 4.9. Servomotor.

4.9 Alimentación 2

La alimentación encargada de energizar los servomotores la lleva a cabo un batería recargable marca Steren sellada de acido plomo de 12v 4Ah la cual tiene un peso de 1.4kg como se muestra en la Figura 4.10. La fuente es acoplada al dispositivo móvil y es conectada a las terminales de alimentación de puente H.



Figura 4.10 Batería Recargable.

4.10 Software del microcontrolador

Programa Principal.

El funcionamiento general del programa principal encargado de ejecutar los movimientos y direcciones del prototipo seguidor de línea se basa directamente en la información almacenada en la variable receptora de datos de la cual se realizan las instrucciones pertinentes para la manipulación del sentido del giro de los servomotores. Los movimientos coordinados e independientes de los servomotores dan como resultado cuatro tipos de movimientos que permiten al prototipo detectar y seguir un patrón de línea, estos movimientos son: avance, izquierda, derecha y retroceso. El diagrama de flujo de la Figura 4.11 muestra de manera general dichos movimientos respecto a la variable encargada de almacenar los datos. En conjunto todo el programa se presenta en el anexo 1.

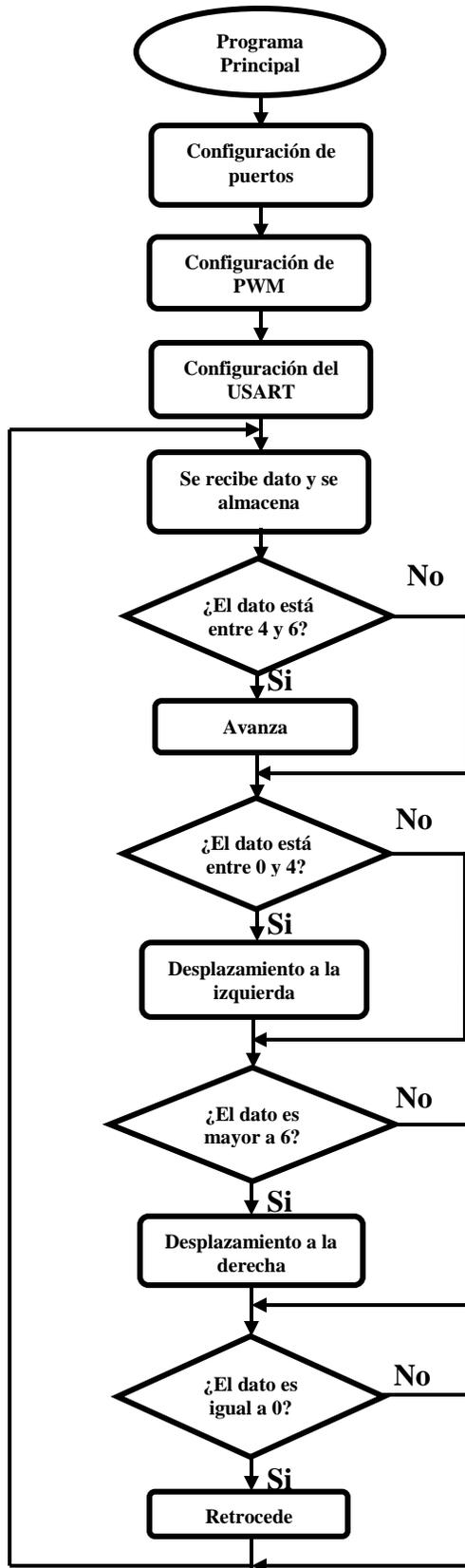


Figura 4.11 Diagrama de Flujo del programa principal.

Configuración de Puertos.

Se configura el puerto D como salida para su uso en el control de las señales PWM en el circuito externo LM7408. Así también se hacen las configuraciones de entrada del puerto C para la señal de recepción del pin RX del microcontrolador y las salidas correspondientes a las dos señales PWM que son administradas por el circuito externo para su activación en el puente H. La configuración como entrada o salida de los puertos C y D se realiza mediante los registros TRISC y TRISD respectivamente.

Configuración de PWM.

Se realiza la configuración de los registros CCP1CON y CCP2CON encargados de generar la señal PWM a las salidas de los pines 16 y 17 del microcontrolador así como también la configuración del registro T2CON en el que se establece un periodo de 20 μ s y una frecuencia de la señal PWM de 50KHz.

Configuración del USART.

Se establecen las configuraciones de los registros para la recepción de datos comenzando por la velocidad de transmisión a través del registro BRGH y SPBRG para el baud rate de 9600 que es el valor por default establecido tanto para recepción como transmisión de datos para la comunicación serial entre Labview y el microcontrolador. Otro registro que se configura es el RCSTA en donde se elige el modo de comunicación asíncrono así como la recepción de datos a 8 bits, la habilitación tanto del puerto serial como la de recepción de datos.

Recepción y almacenamiento de datos.

Después de las configuraciones a los registros se procede a ejecutar las instrucciones que controlaran el hardware del dispositivo por medio de los datos recabados por el software del programa a través de la recepción de datos del microcontrolador. El programa entra en un ciclo infinito en donde primeramente se llama la función de recepción de datos (getch) encargada de leer la información proveniente del puerto serial y almacenarla en una variable (c) para su manipulación en las condiciones del movimiento del prototipo.

Condición de Avance.

Se establece la condición en la cual si el valor almacenado en la variable c se encuentra dentro del rango de valores en código ASCII 4 y 6 se produzca una combinación de valores en alto y bajo de las salidas del puerto D (RD0, RD1, RD2, RD3) así como también la activación del ciclo de trabajo del PWM en los registros CCPR1L y CCPR2L. Para esta condición se establece un ciclo de trabajo de 52 (aproximadamente 80% de su capacidad máxima y óptima para una búsqueda en trayecto recto). Las salidas de los pines RD0 y RD1 son puestos en bajo (0) mientras RD2 y RD3 se ponen en alto (1) logrando con esto seguir una trayectoria recta a velocidad constante y torque necesario para desplazar el peso del dispositivo seguidor de línea.

Condición de desplazamiento a la izquierda.

En la segunda condición si el valor almacenado en la variable c se encuentra dentro del rango de valores en código ASCII 0 y 4 se produce una combinación de las salidas de los pines RD1 y RD2 en bajo (0), RD0 y RD3 en alto (1) logrando un desplazamiento a la izquierda a velocidad constante y torque necesario para desplazar el peso del dispositivo seguidor de línea. Para esta condición se establece un ciclo de trabajo de 49 (aproximadamente 75% de su capacidad máxima y óptima para una búsqueda en un desplazamiento a la izquierda).

Condición de desplazamiento a la derecha.

En la tercera condición se establece que si el valor almacenado en la variable c se encuentra dentro del rango de valores en código ASCII mayor a 6 se produzca una combinación de las salidas de los pines RD0 y RD3 en bajo (0), RD1 y RD1 en alto (1) logrando con esto un desplazamiento a la derecha a velocidad constante y torque necesario para desplazar el peso del dispositivo seguidor de línea. Para esta condición se establece un ciclo de trabajo de 49 (aproximadamente 75% de su capacidad máxima y óptima para una búsqueda en un desplazamiento a la derecha).

Condición de retroceso.

En la cuarta condición si el valor almacenado en la variable c en código ASCII es igual a 0 se producirá una combinación de valores en alto y bajo de las salidas de los pines RD1 y RD3 puestos en bajo (0), RD0 y RD2 en alto (1) logrando con esto un desplazamiento de retroceso a velocidad constante y torque necesario para desplazar el peso del dispositivo seguidor de línea. El programa descrito se presenta en el apéndice A. Para esta condición se establece un ciclo de trabajo de 39 (aproximadamente 65% de su capacidad máxima y óptima para una búsqueda en un desplazamiento de retroceso).

4.11 Armado del prototipo para el seguimiento de línea

Se presenta el funcionamiento del prototipo seguidor de línea en donde se enfatiza de una manera más detallada el uso de cada dispositivo del hardware en función del software, se explican los pasos a seguir en el uso del software IMAQ Vision para la selección del patrón a detectar, conexiones pertinentes del hardware así como su montaje y acoplamiento para obtener un desplazamiento correcto en el seguimiento de la trayectoria a detectar. Una vista general sobre las conexiones del hardware se presenta en la Figura 4.12a y 4.12b.

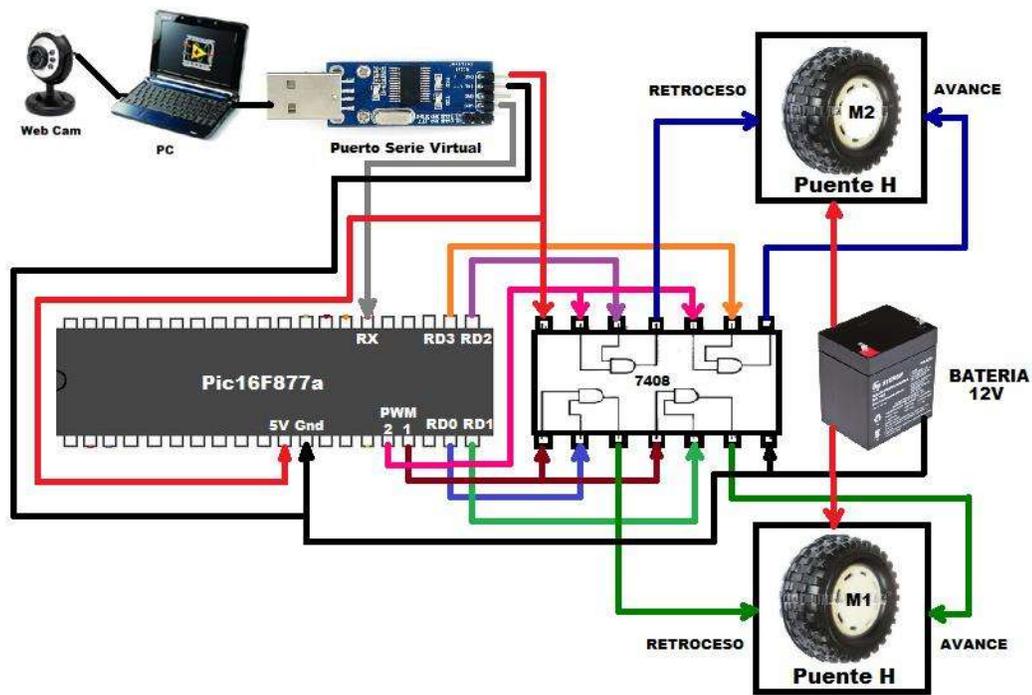


Figura 4.12a Vista general de las conexiones del dispositivo.

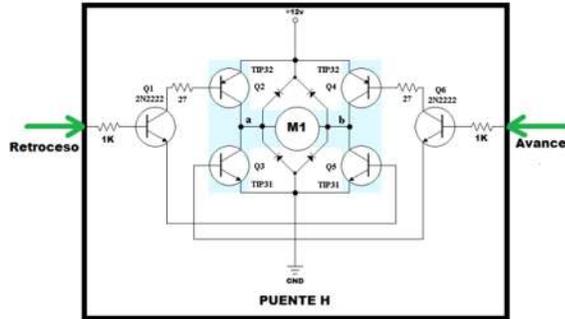


Figura 4.12b Conexiones de los puentes H para M1 y M2 del dispositivo.

4.12 Montaje y Acoplamiento del Hardware

Para el montaje de webcam, PC, microcontrolador, batería y ruedas que desplazaran el dispositivo se utiliza una base rígida de cerámica la cual puede ser moldeada y perforada para realizar conexiones y acoplamientos. Las dimensiones se presentan en la Figura 4.13.

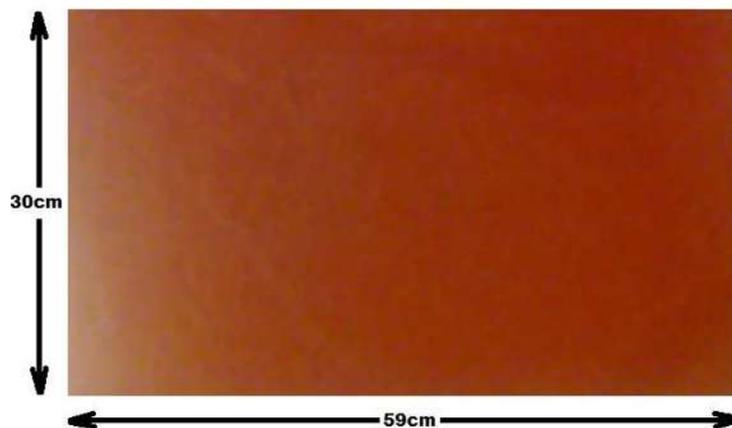


Figura 4.13 Dimensiones de la base para acoplamiento y montaje.

El primer acoplamiento es el de los dos servomotores con ruedas de caucho ubicadas en los ejes los cuales se ubican de manera simétrica a una distancia de 13 cm de las esquinas, cada rueda tiene un diámetro de 9cm. El apoyo central se conforma por una esfera ubicada a 8cm de la parte central, la esfera se ubica en un cilindro para tener un movimiento independiente y una fricción mínima con el piso. La vista inferior del dispositivo se muestra en la Figura 4.14.

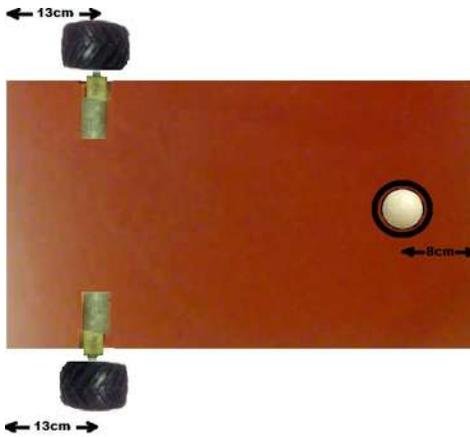


Figura 4.14 Vista inferior del acoplamiento de servomotores.

El montaje de la webcam se realiza en la parte superior del dispositivo móvil, es ubicada a una altura de 16 cm de la base y 24 cm desde el piso hasta el eje focal posicionado a un ángulo de 45 grados el cual se determino a través de pruebas con varios ángulos para una búsqueda óptima del trayecto sobre el piso como se muestra en la Figura 4.15.

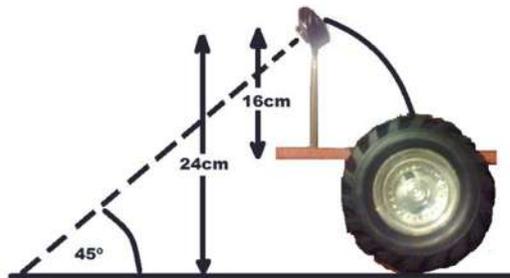


Figura 4.15 Vista lateral del montaje de la webcam.

La ubicación de la cámara se halla en la parte central del lado más cercano a los servomotores y las ruedas como se observa en la Figura 4.16.



Figura 4.16 Vista frontal del montaje de la webcam.

El siguiente montaje es el del circuito impreso que contiene las pistas, bornes y conexiones necesarias del microcontrolador con el puerto serie virtual, la etapa de control, los puentes H y las respectivas fuente de alimentación. El circuito impreso es diseñado con el software llamado *Eagle* que cuenta con las herramientas necesarias para realizar una impresión de conexiones definidas y estéticas pero con un alto desempeño en su función.

El material utilizado en el diseño y montaje de componente del circuito impreso es:

1 Microcontrolador PIC16f877A.	4 Bornes de 2 conexiones
1 Circuito Integrado LM7408.	1 Interruptor.
4 Transistores TIP30.	1 Cristal de 20MHz
4 Transistores TIP31C.	1 Resistencia 460Ω
4 Transistores 2N2222A.	1 Resistencia 22KΩ
8 Resistencias 1KΩ.	1 Condensador 0.47μF
1 Led	2 Condensadores 22μF

El diseño y montaje de superficie de los componentes en forma real se presenta en la Figura 4.17. El diseño impreso y el circuito esquemático se muestran en el anexo 2.

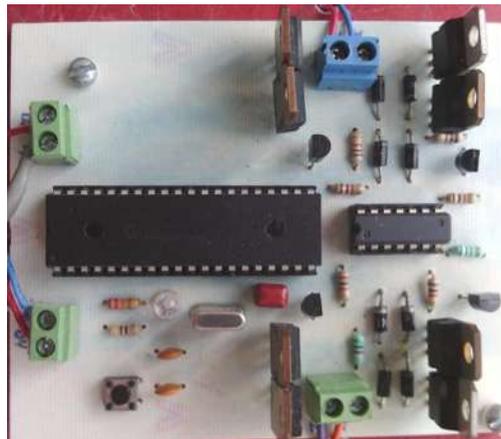


Figura 4.17 Vista Superior del montaje de componentes

El montaje del circuito impreso en la base cerámica es ubicado en la parte central tras la base de la webcam. Los bornes laterales ubicados entre los transistores de potencia son los conectores para las terminales de los servomotores, el borne inferior cercano al

microcontrolador conecta el circuito al suministro de alimentación 1 y su referencia mientras que el borne superior recibe la terminal TX del puerto serie virtual al receptor RX así como también el suministro de alimentación 2. Para el montaje de la batería se armo un cerco alrededor que contuviera el movimiento del mismo debido a su peso y el desplazamiento del dispositivo al momento de avanzar retroceder o dar un giro, su ubicación mostrada en la Figura 4.18 se dispuso en la parte media central de la base para una mejor distribución del peso.



Figura 4.18 Montaje y ubicación de webcam, circuito impreso y batería.

La computadora PC se ubica en la parte posterior y es sólo montada sin ser sujeta ya que su peso de 2.7 kg es suficiente para mantenerse estable. El montaje completo del hardware se muestra en la Figura 4.19.



Figura 4.19 Montaje y acoplamiento completo del hardware.

4.13 Configuración de parámetros del Software IMAQ Vision

Después de acoplar y poner en marcha el funcionamiento de la PC en la base del dispositivo se procede a iniciar el software de *Labview* para que aparezca en la pantalla el panel de control del software detector de línea. Enseguida se conectan a algunos de los puertos USB de la computadora la webcam y el puerto serie virtual ya que el software requiere obligatoriamente de la detección de estos dispositivos. Una vez conectados los dispositivos USB se hacen los ajustes necesarios a los controles del puerto serial virtual en la interfase de usuario en donde se encuentra la selección del puerto COM, el baud rate, los bits de datos, el bit de paridad y el flujo de control. La detección del puerto serie virtual puede ser visualizado como se muestra en la Figura 4.20 en el administrador de dispositivos de la PC donde es nombrado COM4.

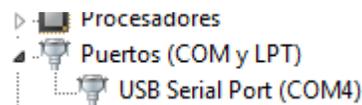


Figura 4.20 Detección del puerto serie virtual.

En el panel de control mostrado en la Figura 4.21 se elige de entre los puertos disponibles el puerto COM4 por el cual se transmitirán los datos, el baud rate por default es el de 9600 y el indicado para trabajar a la misma velocidad por el establecido en el microcontrolador. Los bits de datos están establecidos por default en 8 bits que son en número de bits de datos entrantes, se establece por default que no exista algún bit de paridad ni un flujo de control específico y que este activado el carácter de terminación el cual señala cuando se ha terminado de transmitir un dato al software.



Figura 4.21 Configuración de los controles del puerto serial.

El siguiente ajuste de parámetros mostrado en la Figura 4.22 es para las coincidencias de color donde se establece la estrategia de búsqueda, el valor mínimo de coincidencias, la sensibilidad al color y contraste mínimo. En el primero se establece una búsqueda conservativa por ser el método más fiable para detectar coincidencias. El valor mínimo de coincidencia valido para ser considerado una coincidencia tiene rango entre 0 y 1000, se elige un valor mínimo de 800. La sensibilidad al color se establece como alta para un mejor desempeño en la detección del patrón. El contraste mínimo por default es 0 como optimo para la búsqueda.



Figura 4.22 Configuración de los controles de color.

El siguiente ajuste es la elección de la webcam a utilizar, la PC cuenta con una cámara integrada de la marca Chicony sin embargo no se hace uso de esta por motivos de posición y resolución. Así para este caso se hace uso de la webcam conectada al puerto USB, las cámaras disponibles se visualizan en el indicador “nombre de cámara”, cada webcam es numerada desde cero en orden ascendente como se muestra en la Figura 4.23.

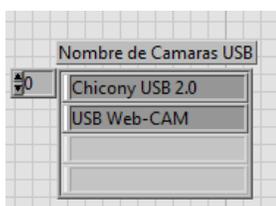


Figura 4.23 Indicador de webcam disponibles.

Para elegir la webcam se hace uso del control “selección de cámara” mostrado en la Figura 4.24 donde los cursores eligen el numero de la cámara visualizada en el indicador.

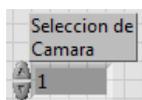


Figura 4.24 Selector de Webcam.

4.14 Inicialización del software prototipo seguidor de línea

Después de haber ajustado los parámetros del software ahora se va a determinar cuál es el patrón a detectar por el prototipo seguidor de línea, para este experimento se traza una línea negra de 2cm de ancho la cual tiene contraste con un piso de fondo blanco como se muestra en la Figura 4.25, mediante la línea se pretende realizar una trayectoria con diferentes tipos de ángulos que el dispositivo pueda seguir.



Figura 4.25 Patrón de línea a detectar.

Realizados los ajustes necesarios a los controles se procede a inicializar el prototipo seguidor de línea, como primer paso se posiciona el dispositivo de manera paralela sobre el patrón de color que se desea detectar (línea negra) a manera que la webcam posicionada a 45° la pueda visualizar como se observa en la Figura 4.26.



Figura 4.26 Posicionamiento del prototipo seguidor de línea.

Se procede a utilizar el panel frontal del software implementado mostrado en la Figura 4.27, la primera acción es ejecutar el programa dando click al botón con una flecha blanca  que se encuentra en la barra superior. Al momento de hacer esto se activa la captura de video por parte de la webcam permitiendo visualizar la línea a seguir.

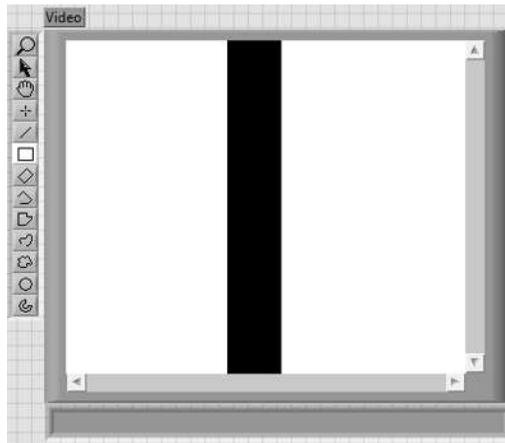


Figura 4.27 Visualización de la línea a detectar.

Se elige sobre la pantalla de video el patrón que se toma como comparativo a detectar entre la imagen y la secuencia de video. Esto se hace eligiendo la opción de región de interés ROI cuadrangular  que se encuentra al lado de la pantalla de video como se muestra en la Figura 4.28.

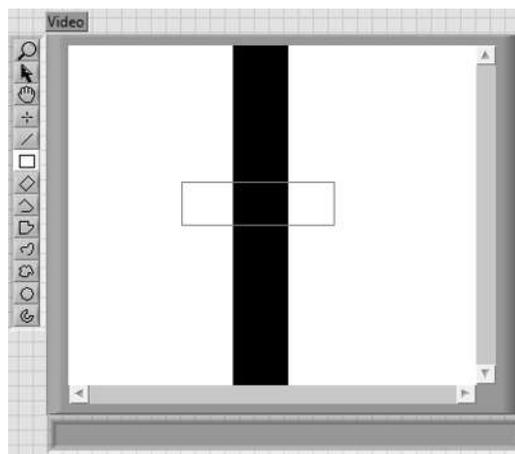


Figura 4.28 Elección del patrón a detectar.

El área elegida en la ventana de “video” se translada inmediatamente a la ventana “foto” en donde es visualizada como se muestra en la Figura 4.29.

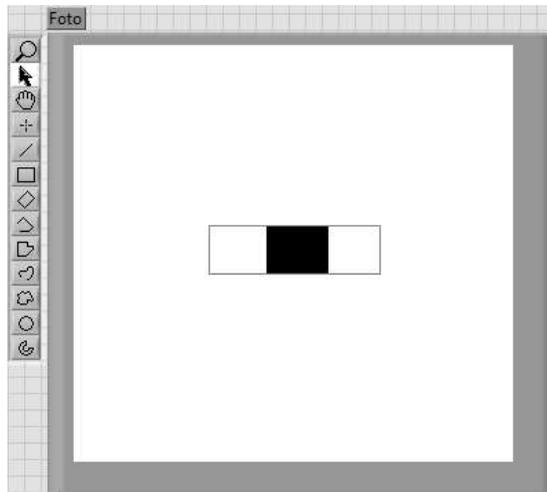


Figura 4.29 Visualización del patrón a detectar.

Una vez elegido y visualizado el patrón a detectar el siguiente paso es aprender el patrón, esto se hace haciendo un click sobre el control que lleva el mismo nombre mostrado en la Figura 4.30.



Figura 4.30 Activación del control para aprender patrón.

Ya aprendido el patrón a localizar el siguiente paso es localizar el patrón elegido en una búsqueda continua de video a través de los controles mostrados en la Figura 4.31, para ello se activa el control de búsqueda continua representado en un interruptor el cual cambia su estado de color azul (apagado) a verde (encendido).

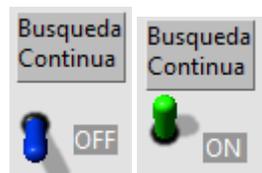


Figura 4.31 Activación del control para búsqueda continua del patrón.

Realizada la activación del control para búsqueda continua se observa reflejada en la imagen de video las coincidencias de color entre el patrón elegido y las halladas en la captura continua siendo señaladas por el indicador visual de coincidencias como se observa en la Figura 4.32.

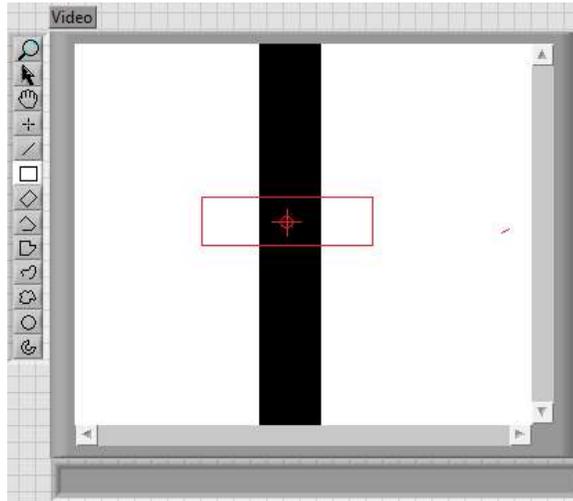


Figura 4.32 Indicador de coincidencias.

4.15 Resumen

Se analizó cada componente y dispositivo que conforma el hardware del prototipo seguidor de línea mostrando un esquema general de su estructura en un diagrama de bloques en donde se enfatizó el funcionamiento del microcontrolador el cual es el componente maestro que contiene el programa encargado de direccionar los datos provenientes del software para su manipulación en el movimiento del prototipo, se explicó la configuración de los registros utilizados en el programa para después entrar en el análisis del ciclo principal del programa que contiene las instrucciones principales para dirigir el movimiento del prototipo seguidor de línea. Se presentó el ensamblaje del prototipo seguidor de línea comenzando por el posicionamiento y acoplamiento de los dispositivos que la conforman como lo son webcam, servomotores, fuentes de alimentación, PC y circuito de control, se detallaron las configuraciones de parámetros necesarios en el software para el óptimo funcionamiento del prototipo de entre los cuales se destacan la velocidad de transmisión de datos por el puerto serial así como el parámetro de detección mínimo de coincidencias entre otras mencionadas que son de relevancia para el funcionamiento.

Capítulo 5

Funcionamiento y pruebas del prototipo seguidor de línea

5.1 Funcionamiento del prototipo seguidor de línea

La etapa de funcionamiento tiene como objetivo rastrear una imagen seleccionada y compararla con una adquisición de video en tiempo real, en la detección se involucra el control y corrección del movimiento del dispositivo en base a los datos recabados por el software IMAQ vision. Como se menciona en la sección 3.5.5 se crea un arreglo que extrae la información de la posición del indicador de coincidencias para su uso y manipulación de salidas del microcontrolador que concluye en la activación y control del giro de los servomotores. Mediante el arreglo realizado para la extracción de la posición en el eje X del indicador de coincidencias, se manipula el valor original de las dimensiones del display que va de 0 a 320 a un nuevo valor reducido que va de 0 a 10 esto para un mejor manejo de datos por parte del microcontrolador. Para el rastreo y seguimiento de línea se establecen tres tipos de movimientos los cuales corresponden a la posición del indicador de coincidencias dentro del área del display de video, un cuarto movimiento se activa al no hallar coincidencias correspondientes a una detección de patrón de color. De manera imaginaria el display de video se divide en tres partes de las cuales cada una es destinada a un movimiento específico así como se presenta en la Figura 5.1.

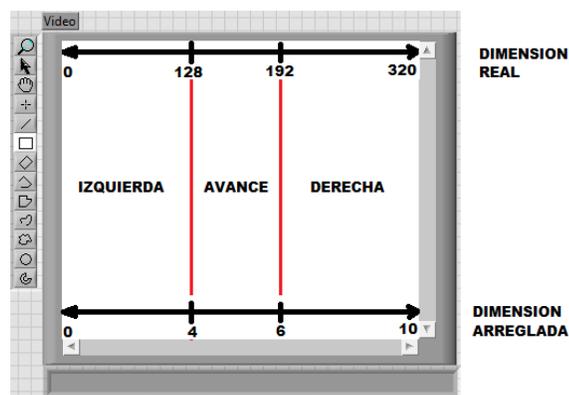


Figura 5.1 División de movimientos en el display de video.

El rango de valores en el eje de las X puede ser visualizado en los indicadores del panel de control de la Figura 5.2 siendo X el valor dimensional original dentro del rango de 0 a 320; X/Y es el valor dimensional reducido por el arreglo que va de 0 a 10 y por último el indicador *string* que es el dato enviado a través del puerto serial y requerido en cadena de caracteres por el microcontrolador.

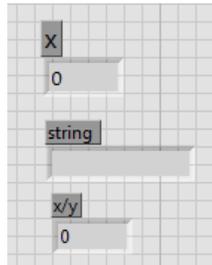


Figura 5.2 Indicadores de posicionamiento en el eje X.

5.2 Pruebas de desplazamiento del prototipo

Cada prueba de movimiento del prototipo se detalla y analiza en las siguientes secciones para un mejor entendimiento por parte del usuario sobre la estructura y relación de funcionamiento entre el hardware y el software que en conjunto integran el desplazamiento del prototipo seguidor de línea.

5.2.1 Desplazamiento de avance

El desplazamiento de avance tiene como objetivo movilizar el prototipo de manera recta, esto en consecuencia de la posición del indicador de coincidencias dentro del área central del display de video lo cual es traducido también como la detección del patrón de color dentro del eje focal central de la cámara. El rango de valores en el eje X para que el dispositivo mantenga una trayectoria recta de avance se encuentra entre 4 y 6. Como se ejemplifica en la Figura 5.3, tanto el indicador de coincidencias y consecuentemente la línea a seguir se encuentra en el umbral central, los indicadores señalan los valores en los que se encuentra dicha coincidencia para ser considerada dentro del rango de valores para un desplazamiento de avance.

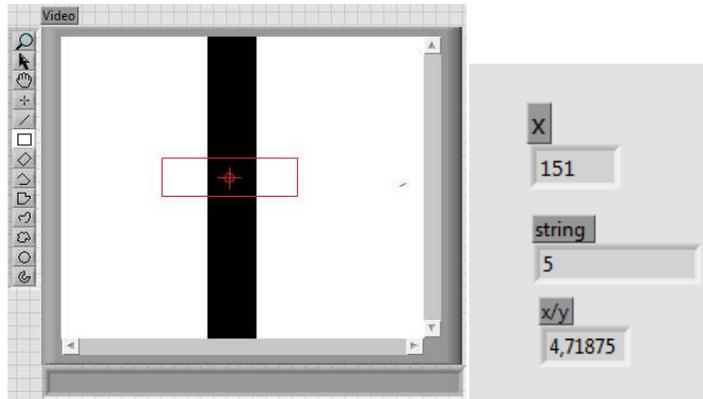


Figura 5.3 Rango y posición para desplazamiento de avance.

El dato *string* con valor de 5 es enviado a través del puerto serial al pin de recepción del microcontrolador para ser almacenado en la variable *c*, dentro del programa principal este valor invoca una condición en el rango de valores que activen la combinación en alto y bajo de las salidas del puerto D y que a su vez permitan al dispositivo activar el control de los servomotores en un mismo sentido. Los pines RD1 y RD3 del microcontrolador puestos en alto permitiendo la activación de las compuertas AND 2 y 3 del circuito integrado LM7408 para que consecuentemente estén en marcha el sentido de giro avance de cada puente H que controla el servomotor, mientras que los pines RD0 y RD2 puestos en bajo desactivan las salidas de las compuertas AND 1 y 4 para que no exista flujo de señal a través de las entradas de retroceso. Las conexiones activas son mostradas en la Figura 5.4.

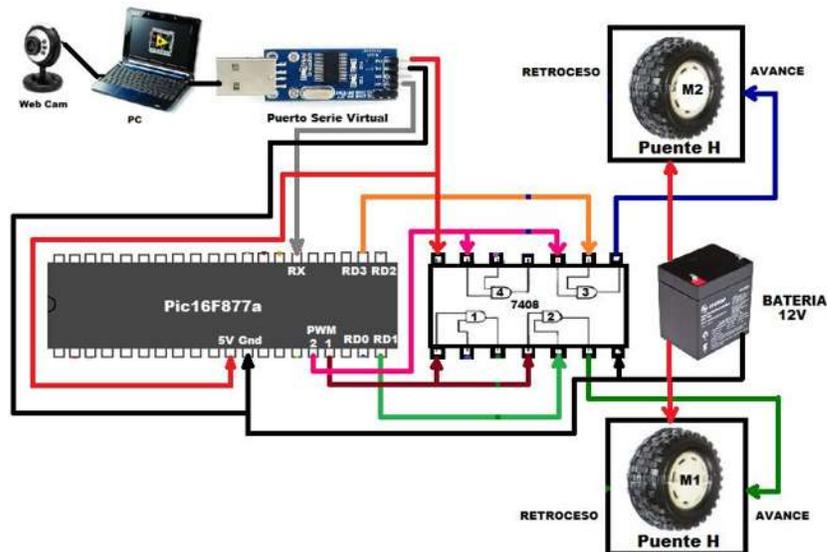


Figura 5.4 Activación de pines para desplazamiento de avance.

El movimiento de avance del prototipo seguidor de línea se muestra en la siguiente secuencia de la Figura 5.5.

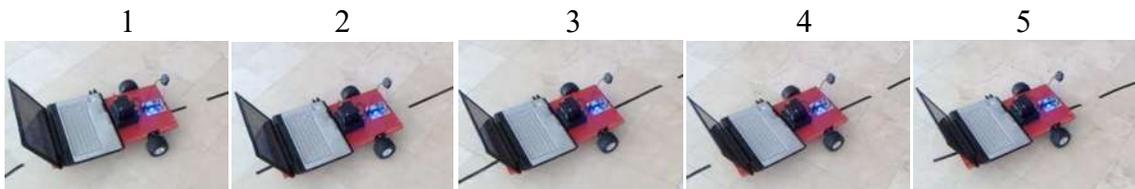


Figura 5.5 Secuencia de avance del prototipo seguidor de línea.

5.2.2 Desplazamiento a la derecha

Este desplazamiento moviliza el prototipo a la derecha, esto en consecuencia de la posición del indicador de coincidencias dentro del área lateral derecha del display de video lo cual es traducido también como la detección del patrón de color fuera del eje focal central de la cámara con un desvío hacia la derecha. El rango de valores en el eje X para que el dispositivo inicie este desplazamiento es establecido a los que sean mayores pero no iguales a 6. Como se ejemplifica en la Figura 5.6, tanto el indicador de coincidencias y consecuentemente la línea a seguir se encuentra fuera del umbral central, los indicadores señalan los valores en los que se encuentra dicha coincidencia para ser considerada dentro del rango de valores para un desplazamiento a la derecha.

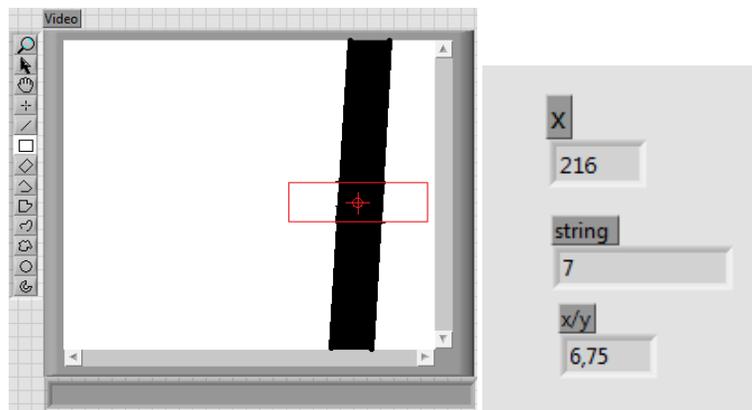


Figura 5.6 Rango y posición para desplazamiento a la derecha.

El dato *string* con valor de 7 es enviado a través del puerto serial al pin de recepción del microcontrolador para ser almacenado en la variable *c*, dentro del programa principal este

valor invoca una condición en el rango de valores que activen la combinación en alto y bajo de las salidas del puerto D y que a su vez permitan al dispositivo activar el control de los servomotores en un diferente sentido. Los pines RD1 y RD2 del microcontrolador son colocados en alto permitiendo la activación de las compuertas AND 2 y 4 del circuito integrado LM7408 para que consecuentemente estén en marcha el sentido de giro avance del motor 1 y el sentido de giro en retroceso del motor 2 que en conjunto generan el movimiento a la derecha, mientras que los pines RD0 y RD3 puestos en bajo desactivan las salidas de las compuertas AND 1 y 3 para que no exista flujo de señal a través de las entradas opuestas a las ya activadas. Las conexiones activas se muestran en la Figura 5.7.

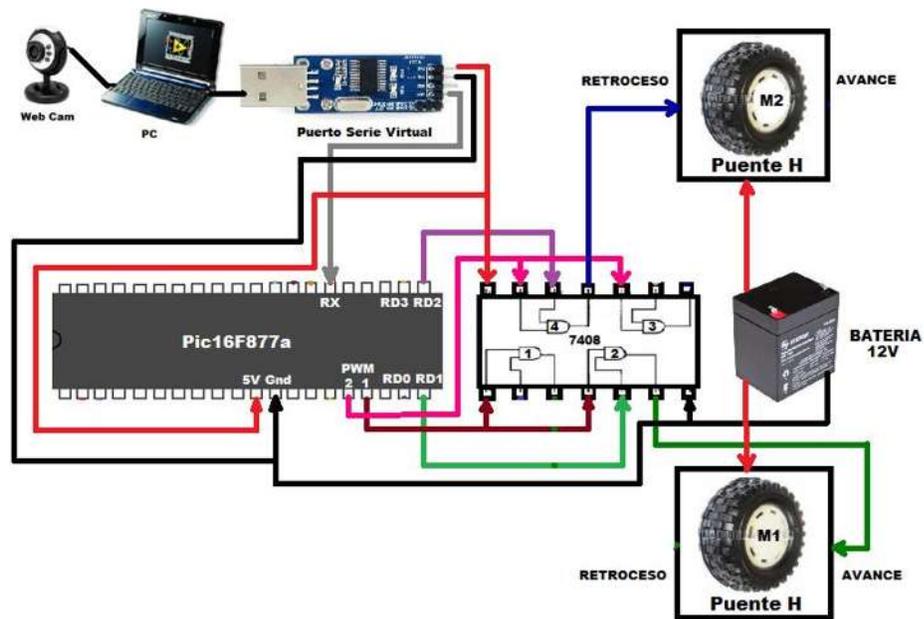


Figura 5.7 Activación de pines para desplazamiento a la derecha.

El movimiento a la derecha del prototipo seguidor de línea se muestra en la siguiente secuencia de la Figura 5.8.



Figura 5.8 Secuencia de desplazamiento a la derecha del prototipo seguidor de línea.

5.2.3 Desplazamiento a la izquierda

Este desplazamiento moviliza el prototipo a la izquierda, esto en consecuencia de la posición del indicador de coincidencias dentro del área lateral izquierda del display de video. El rango de valores en el eje X para que el dispositivo inicie este desplazamiento es establecido a los que sean mayores pero no iguales a 0 y menores pero no iguales a 4. Como se ejemplifica en la Figura 5.9, tanto el indicador de coincidencias y consecuentemente la línea a seguir se encuentra fuera del umbral central, los indicadores señalan los valores en los que se encuentra dicha coincidencia para ser considerada dentro del rango de valores para un desplazamiento a la izquierda.

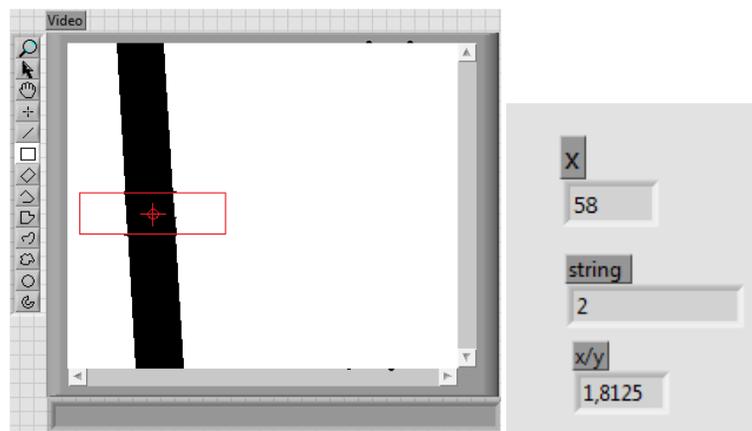


Figura 5.9 Rango y posición para desplazamiento a la izquierda.

El dato *string* con valor de 2 es enviado a través del puerto serial al pin de recepción del microcontrolador para ser almacenado en la variable *c*, dentro del programa principal este valor invoca una condición en el rango de valores que activen la combinación en alto y bajo de las salidas del puerto D y que a su vez permitan al dispositivo activar el control de los servomotores en un sentido contrario. Los pines RD0 y RD3 del microcontrolador son colocados en alto permitiendo la activación de las compuertas AND 1 y 3 del circuito integrado LM7408 para que consecuentemente estén en marcha el sentido de giro avance del motor 2 y el sentido de giro en retroceso del motor 1 que en conjunto generan el movimiento a la izquierda, mientras que los pines RD1 y RD2 puestos en bajo desactivan las salidas de las compuertas AND 2 y 4 para que no exista flujo de señal a través de las entradas opuestas a las ya activadas. Las conexiones activas se muestran en la Figura 5.10.

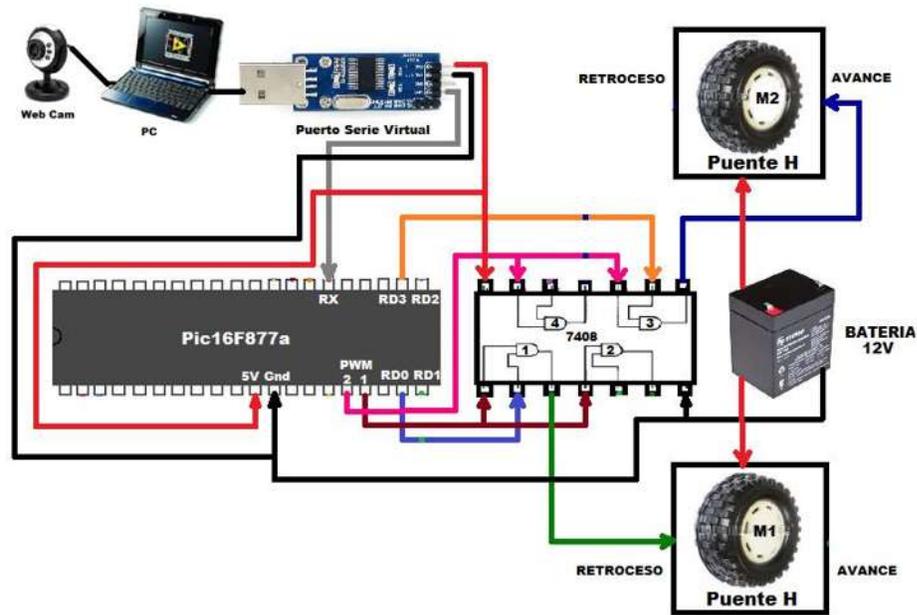


Figura 5.10 Activación de pines para desplazamiento a la izquierda.

El movimiento a la izquierda del prototipo seguidor de línea se muestra en la siguiente secuencia de la Figura 5.11.



Figura 5.11 Secuencia de desplazamiento a la izquierda del prototipo seguidor de línea.

5.2.4 Desplazamiento de retroceso

Este desplazamiento moviliza el prototipo hacia atrás si se pierde en algún momento la detección de la coincidencia dentro del área del display de video hasta que vuelva a detectar de nuevo el patrón. El rango de valores en el eje X para que el dispositivo inicie este desplazamiento es nulo e igual a 0 ya que se traduce en una no detección del patrón. Como se ejemplifica en la Figura 5.12, el indicador de coincidencias no aparece y consecuentemente la línea a seguir se encuentra fuera del campo óptico visual de la cámara o no existen coincidencias de color con el patrón a comparar.

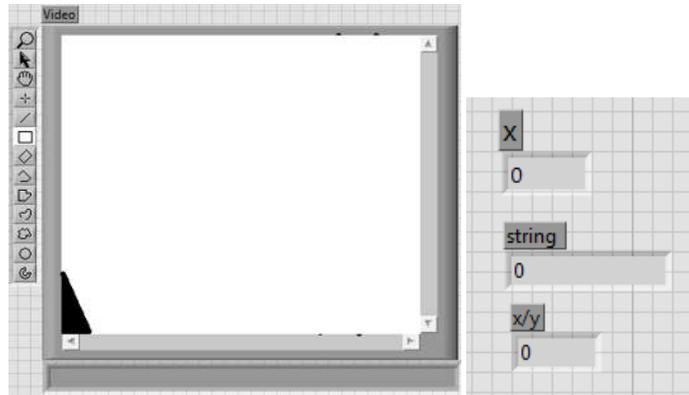


Figura 5.12 Rango y posición para desplazamiento de retroceso.

El dato *string* con valor de 0 es enviado a través del puerto serial al pin de recepción del microcontrolador para ser almacenado en la variable *c*, dentro del programa principal este valor invoca una condición en el rango de valores que activen la combinación en alto y bajo de las salidas del puerto D y que a su vez permitan al dispositivo activar el control de los servomotores en un mismo sentido. Los pines RD0 y RD2 del microcontrolador se colocan en alto permitiendo la activación de las compuertas AND 1 y 4 del circuito integrado LM7408 para que consecuentemente estén en marcha el sentido de giro en retroceso del motor 1 y 2 que en conjunto generan el movimiento hacia atrás, mientras que los pines RD1 y RD3 puestos en bajo desactivan las salidas de las compuertas AND 2 y 3 para que no exista flujo de señal a través de las entradas opuestas a las ya activadas. Las conexiones activas se muestran en la Figura 5.13.

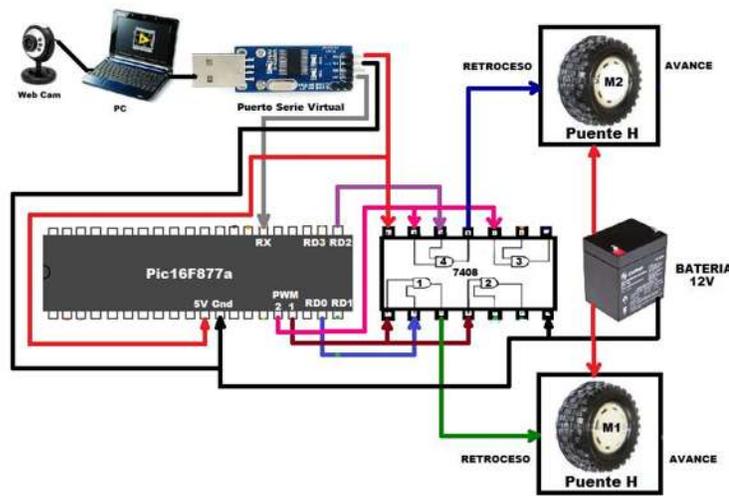


Figura 5.13 Activación de pines para desplazamiento de retroceso.

El movimiento de retroceso del prototipo seguidor de línea se muestra en la siguiente secuencia de la Figura 5.14.



Figura 5.14 Secuencia de retroceso del prototipo seguidor de línea.

5.3 Resumen

Se explicó el funcionamiento del prototipo seguidor de línea así como las pruebas de desplazamiento de los cuatro tipos de movimientos que presenta en la búsqueda del patrón a seguir señalando el umbral óptico necesario en el display de video para ser reconocido y manipulado por el microcontrolador en la activación de los sentidos de giro de cada servomotor y en consecuencia el desplazamiento de todo el dispositivo.

Capítulo 6

Conclusiones

6.1 Introducción

El procesamiento digital de imágenes ha tomado gran auge en los últimos años como una herramienta ampliamente usada en los diferentes campos de la ciencia entre ellos uno muy importante como lo es la robótica, IMAQ vision de LabVIEW ofrece una amplia gama de herramientas que adentran al usuario a los principios básicos del procesamiento de imágenes gracias al ambiente de programación gráfico con el que cuenta, no solamente para su uso estudiantil sino para aplicaciones industriales que requieren de la supervisión de eventos a nivel de inspección detallada que un humano no puede realizar a simple vista. Este trabajo tuvo como función aplicar algunas de las amplias herramientas que el software ofrece en un prototipo encargado de rastrear un patrón de línea en base a sus características cromáticas las cuales son manipuladas en el software mencionado.

6.2 Conclusiones Generales

Entre los beneficios del uso del paquete computacional IMAQ vision para el procesamiento digital de imágenes se mencionan los siguientes:

- Permite al usuario adentrarse en un ambiente de programación gráfica en el entendimiento y aplicación del procesamiento digital de imágenes a través de herramientas accesibles para la economía del usuario y que solo estaba destinada en el pasado al uso industrial de las empresas que contaban con los recursos para su alcance.
- Existe una amplia gama de aplicaciones con el uso de la herramienta IMAQ Vision en diferentes campos de la ciencia además de la robótica en donde es aplicado este trabajo, dependiendo sólo de la imaginación del usuario para explotar dichas herramientas.
- En el presente trabajo al utilizar el software IMAQ Vision en un prototipo seguidor de línea se alcanzaron los objetivos de localización y búsqueda del patrón de línea

para uso y demostración del potencial del software en principios básicos de programación y procesamiento de imágenes.

- Mediante un bajo coste en el uso de un sensor óptico indispensable como lo es una webcam que está al alcance del usuario en cualquier tienda es posible acceder al amplio mundo de ideas y desarrollo de proyectos que el procesamiento digital de imágenes ofrece.
- Entre las aportaciones de este trabajo se mencionan los pasos a seguir para la correcta instalación del software en conjunto con el driver de la cámara USB en la plataforma de Windows 7 así como el uso de los comandos de captura, aprendizaje y búsqueda de un patrón de color determinado.

La aplicación del software IMAQ Vision en el prototipo seguidor de línea presento limitantes las cuales se mencionan a continuación:

- El software IMAQ vision requiere estrictamente de la detección de una conexión webcam USB a sus sistema lo cual recae en el armado de un prototipo de dimensiones más amplias para el montaje de la PC.
- Además del espacio la PC aporta un mayor peso al prototipo que consecuentemente requería de una mayor cantidad de fuerza para su movilidad. Esto fue resuelto acoplando al prototipo servomotores que proveen un torque adecuado para desplazar el prototipo además de la aplicación de señales PWM que controlan el ancho de pulso y así la velocidad del giro del servomotor.
- La velocidad del dispositivo se limitaba al peso del prototipo y a la óptima detección del patrón de color a detectar. Para trayectos en línea recta la velocidad máxima del dispositivo se estableció a 5m/min; en trayectos con giro a la izquierda y a la derecha la velocidad se redujo a 4m/min debido a que una velocidad mayor originaba una perdida inmediata de la localización del patrón en el margen de vision de la cámara; en trayectos de retroceso la velocidad mínima establecida fue de 2.5 m/min para volver a detectar el patrón en caso de perderlo ya que una velocidad menor a esta causa el detenimiento del móvil debido a que no es suficiente el torque de los motores para desplazar el peso del prototipo.

- Para la detección de patrones de línea con ángulos mayores a 45 grados el prototipo presenta limitantes en la búsqueda debido a la brusca pérdida del patrón dentro del rango de visión de la cámara.

6.3 Trabajos futuros

- En base al prototipo realizado, implementar un software con comunicación inalámbrica que permita un control y manipulación de imágenes a distancia y a su vez elimine las limitantes de velocidad y giro del dispositivo con respecto a su peso y dimensiones.
- Agregar modificaciones al programa en el que se acoplen nuevas herramientas de la gran variedad que ofrece el software de IMAQ vision para un mejor procesamiento y detección del patrón dentro de la imagen para un mayor rango de vision y localización.
- Ampliar el margen de proyectos en el que se haga uso de esta poderosa herramienta de procesamiento digital de imágenes que ofrece el software IMAQ Vision.
- Entre algunas aplicaciones en las que se podría desempeñar el seguidor de línea recae en el transporte de carga industrial a mayor escala y peso través de una trayectoria indicada, así también como el transporte de material peligroso para el hombre como lo son cargas explosivas, material radioactivo o viral; se puede considerar a su vez el transporte de material frágil de un punto a otro y que requiera determinada estabilidad de movimientos por parte del móvil sin afectar la estructura de dicho material. Otra aplicación del trabajo con ciertas modificaciones se podría presentar en el traslado de personas con cierta incapacidad a través de trayectorias indicadas en un hospital, institución u hogar.

Apéndice A

Código del microcontrolador

```
/* =====
   Descripción Inicial

   Se recogerá una muestra de la señal del receptor serial al
   microcontrolador el cual activará 2 servomotores
   ===== */

/* =====
   Archivos cabecera con las definiciones básicas del PIC
   ===== */

#include<pic.h>

/* =====
   Variable Global para recepción de datos
   ===== */

unsigned char c;

/* =====
   Declaración inicial de las funciones para recepción.
   ===== */

unsigned char getch();

/* =====
   DEFINICION DE FUNCIONES
   ===== */

/* =====
   Función de Recepción de carácter por el puerto Serie
   unsigned char getch();
   ===== */

unsigned char getch()
{
    while(!RCIF); //Cuando exista dato en el registro de Recepción
    return(RCREG); //Se devuelve a la Función Principal
}

/* =====
   FUNCION PRINCIPAL
   ===== */

void main(void)
{
    TRISD=0b00000000; // Puerto D configurado como salida.
    TRISC7=1;         // RC7 configurado como entrada para RX de USART
    TRISC2=0;         // RC2 configurado como salida para PWM
    TRISC1=0;         // RC1 configurado como salida para PWM
}
```

```

CCP1CON=0b00001100;// Registro CCP1 configurado en modo PWM
CCP2CON=0b00001100;// Registro CCP2 configurado en modo PWM
T2CON=0b00000100; // Prescalador = 1 Timer 2 encendido
PR2=99;           // Valor para T PWM = 20us => Freq PWM = 50KHz
CCPR1L=0;        // Registro de control del duty cycle inicial = 0.
CCPR2L=0;        // Registro de control del duty cycle inicial = 0.

BRGH=1; //Modo de Alta Velocidad
SPBRG=129; //Transmision de 9600 Baudios
SYNC=0; //Se Configura como Modo Asincrono
SPEN=1; //Se Habilita el Puerto Serie
RX9=0; //Recepcion de 8 Bits
CREN=0; //Se Resetea la Recepcion
CREN=1; //Se habilita la Recepcion

while(1)
{
    c=getch(); // se recibe el dato y se almacena en la variable

    if(c>='4'&& c<='6') // si el dato se encuentra entre 4 y 6
                        // la direccion es hacia adelante

    {
        PORTD=0b00001010;//avance
        CCPR1L=52;
        CCPR2L=52;
    }

    if(c>'0'&& c<'4') // si el dato se encuentra entre 0 y 4
                    // la direccion es hacia la izquierda

    {
        PORTD=0b00001001;//izquierda
        CCPR1L=49;
        CCPR2L=49;
    }

    if(c>'6') // si el dato es mayor a 6
            //la direccion es hacia la derecha

    {
        PORTD=0b00000110;//derecha
        CCPR1L=49;
        CCPR2L=49;
    }

    if(c=='0') // si el dato es igual a 0
            // la direccion es hacia atras

    {
        PORTD=0b00000101;//retroceso
        CCPR1L=39;
        CCPR2L=39;
    }

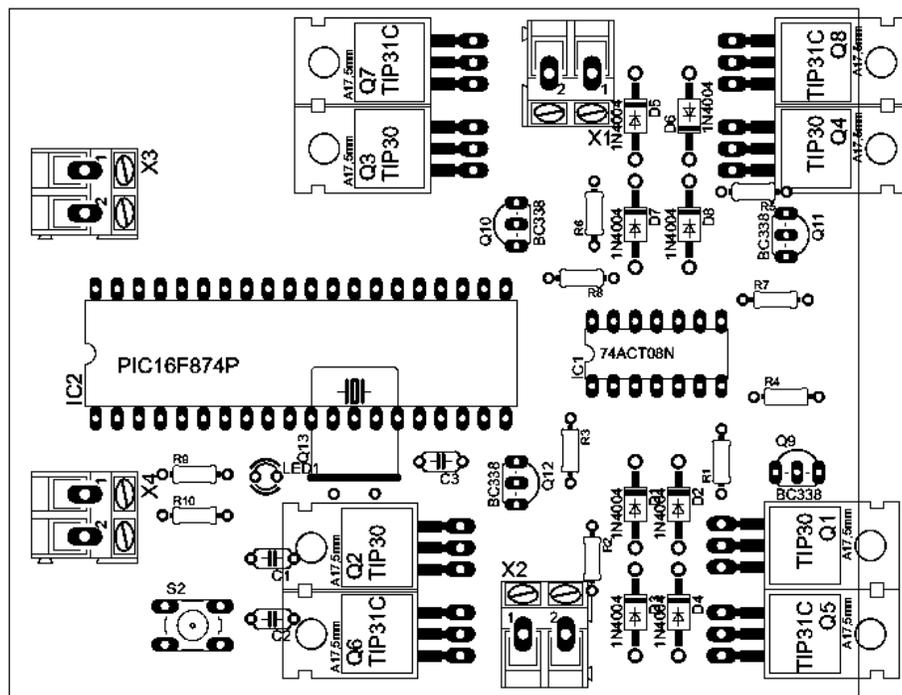
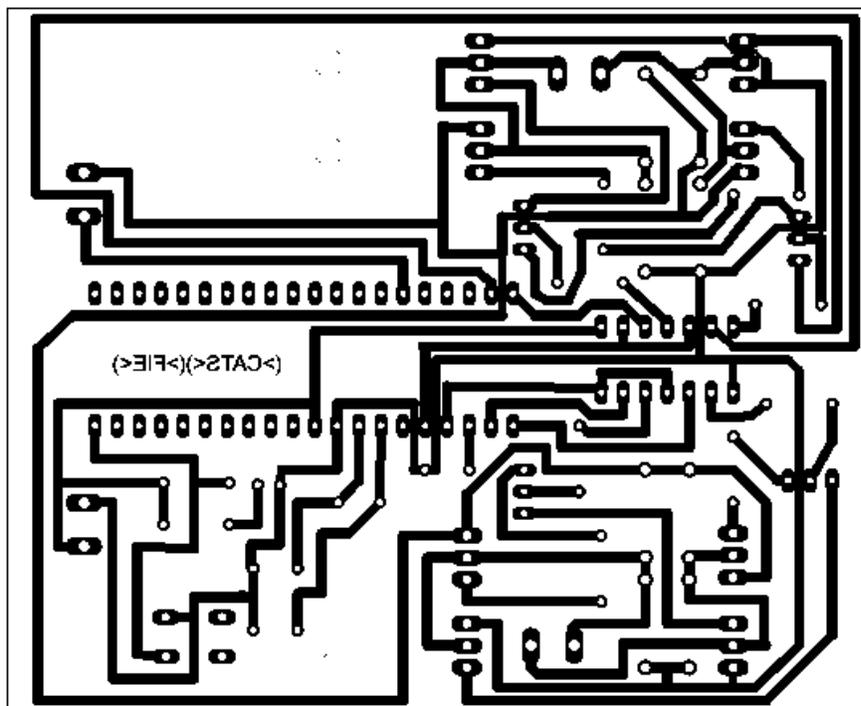
} // fin del while

} // fin del main

```

Apéndice B

Diseño del circuito impreso y montaje de componentes



Referencias

- [1] [R. Rodríguez 2011]
R. Rodríguez, Procesamiento y análisis digital de imágenes, Madrid: Ra-Ma, 2011.
- [2] [National Instruments 2013]
National Instruments. IMAQ Vision for G Reference Manual. 12 de abril del 2013.
<http://www.ni.com>
- [3] [National Instruments 2013]
National Instruments. IMAQ Vision for LabView User Manual. 12 de abril del 2013.
<http://www.ni.com>
- [4] [National Instruments 2013]
National Instruments. Driver para cámaras USB. 25 de abril del 2013.
<http://zone.ni.com/devzone/cda/epd/p/id/5030>.
- [5] [Mercado Libre 2013]
Mercado Libre. Webcam USB. 13 de Junio del 2013.
<http://articulo.mercadolibre.com.mx/MLM-427794354-mini-webcam-usb-con-micrófono-luz-led-vision-nocturna-hd-JM>
- [6] [Mercado Libre 2013]
Mercado Libre. Puerto Serie Virtual. 13 de Junio del 2013.
<http://articulo.mercadolibre.com.mx/MLM-429124471-usb-a-rs232-ttl-pl2303hx-modulo-arduino-sp0-JM>.
- [7] [Microchip 2013]
Microchip. Hoja de datos del PIC16F877A. 14 de Junio del 2013.
<http://www.microchip.com/downloads/en/devicedoc/39582b.pdf>

[8] [Datasheets 2013]

Datasheets. Hoja de datos del circuito LM7408. 19 de Agosto del 2013.

http://www.datasheetcatalog.com/datasheets_pdf/7/4/0/8/7408.shtml

[9] [Stereon 2013]

Stereon. Batería Stereon 12v. 4 de Septiembre del 2013.

<http://www.stereon.com.mx/catalogo/prod.asp?f=&sf=&c=&p=1203&desc=bateria-recargable-sellada--de-acido-plomo--de-12-volts--4-ah&result=bateria 12v>