PROYECTO DE APLICACION

BRAZO ROBOTICO CON BANDA TRANSPORTADORA

INTRODUCCION

- Implementar el siguiente proyecto utilizamos Lego Mindstorms NXT.
- Con Matlab desarrollamos la programación del Brazo Robótico.
- El manejo de sensores, servomotores y microcontroladores complementamos el proyecto.

DESCRIPCION DEL PROYECTO

- El proyecto consiste en la construcción de una banda transportadora con brazo robótico.
- En la banda circularán recipientes llenados a diferentes niveles de producto.
- Estos niveles serán medidos con el sensor de ultrasonido.
- Se utilizan dos sensores infrarrojos en la banda.

DESCRIPCION DEL PROYECTO

- El brazo robótico clasificará los recipientes según el nivel medido.
- El brazo tendrá 2 grados de libertad, utilizando 2 servomotores para mover la base de brazo y un motor que controlará el gipper o garra.

DESCRIPCION DEL PROYECTO

- Contará con un sensor de luz para detectar la presencia de los recipientes.

- La banda será controlada por un microcontrolador, el que tendrá como dispositivo de entrada dos sensores de luz infrarrojo y como dispositivo de salida un servomotor.

APLICACIONES:

- Mejorar la calidad de una línea de producción en una industria envasadora de bebidas.
- Recipientes que no cumplen con el nivel son retornados para que cumplan los requerimientos.

Fundamento Teórico

- Para el desarrollo del Proyecto se usaron varias herramientas de Hardware y Software.

- El Kit de Lego Mindstorms NXT tiene su software de programación, que funciona en base a LabView.

- En nuestro caso usaremos herramientas de programación alternativas, de esta forma trabajaremos un en entorno de diseño e ingeniería.

Herramientas de Hardware utilizadas.

- En esta sección se detallan los equipos utilizados para la construcción del brazo robótico y la banda transportadora .

- Para el primero se uso Lego Mindstorms .

- Para la segunda parte se usó materiales y componentes apropiados para la implementación de prototipos electrónicos.

Lego Mindstroms

- Lego Mindstroms es un juego de robótica fabricado por la empresa Lego, el cual posee elementos básicos de las teorías robóticas.

- Lego Mindstorms puede ser usado para construir un modelo de sistema integrado con partes electromecánicas controladas por computador.

Lego Mindstroms

NXT

- El bloque NXT es una versión mejorada a partir de Lego Mindstorms RCX .
- Debido a la comercialización de los bloques programables, Lego vendió la generación NXT en dos versiones: Versión Educativa y Comercial.

Lego Mindstroms

Microcontrolador

- El microcontrolador que posee es un ARM7 de 32 bits, que incluye 256 Kb de memoria Flash y 64 Kb de RAM externa.
- Posee mayores capacidades de ejecución de programas, evitando que los procesos inherentes de varios paquetes de datos colisionen y produzcan errores en la ejecución del software.

Lego Mindstroms

Entradas y Salidas

- En el bloque de NXT existen cuatro entradas para los sensores, y tres entradas para los servo motores.
- Las salidas de energía aún son tres localizadas en la parte posterior del bloque, haciendo que la conexión para los motores y partes móviles sean de más fácil acceso.

Lego Mindstroms

Comunicaciones

- El bloque de NXT puede comunicarse con el computador mediante la interfaz de USB que posee.
- Conectividad con *Bluetooth* no tan sólo permite conectarse con otros bloques, sino también con computadores, palms, teléfonos móviles, y otros aparatos con esta interfaz de comunicación.

Lego Mindstroms

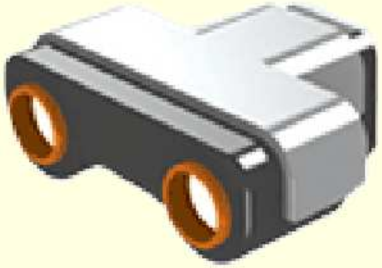
Sensor de Luz de Lego Mindstorms

- El sensor de luz es sin duda uno de los más útiles e interesantes de todo el kit del Lego Mindstorms NXT.
- Este sensor le permite a nuestro robot distinguir entre luz y oscuridad.

Lego Mindstroms

Sensor de Ultrasonido

- Los ultrasonidos son antes que nada sonido, exactamente igual que los que oímos normalmente, salvo que tienen una frecuencia mayor que la máxima audible por el oído humano.



Lego Mindstroms

Sensor de Tacto

- El sensor de contacto permite detectar si el bloque que lo posee ha colisionado o no con algún objeto que se encuentre en su trayectoria inmediata.

Lego Mindstroms

Sensor de Sonido

- El sensor solo detecta la “cantidad” de sonido.
- Este sensor lee el sonido ambiental y nos regresa una medida de 0 a 100%.
- Estos valores corresponden mas o menos a:
    - **4-5%** Una casa silenciosa.
    - **5-10%** Alguien hablando lejos.
    - **10-30%** Es una conversación cerca del sensor o música en un volumen normal.
    - **30-100%** Gente gritando o música a volumen alto.

Lego Mindstroms

Servomotor de Lego Mindstorms NXT

- Los motores de la serie Lego Robotics han sido de tres tipos, los cuales son independientes al bloque.
- Entrega movilidad al sistema dinámico según las necesidades de construcción.
- Los servos además de incluir un motor eléctrico convencional también incluyen un sensor de posición.

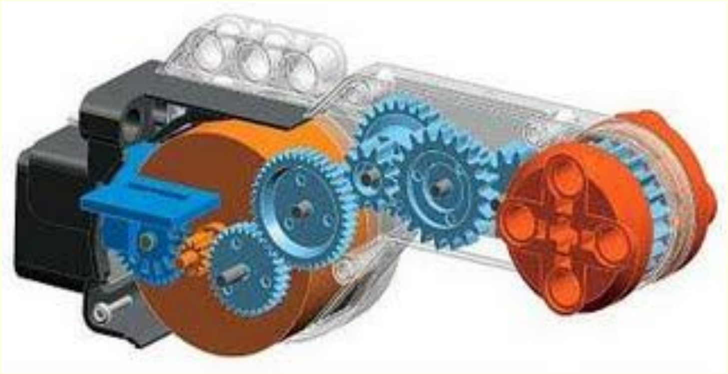
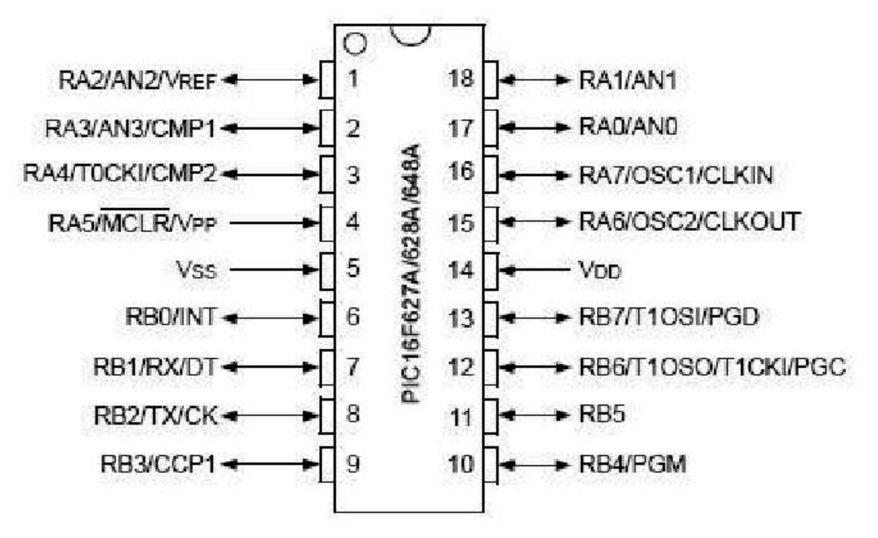


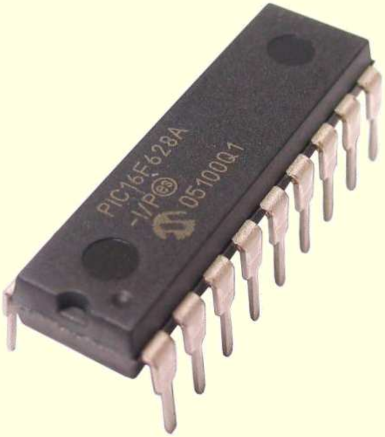
Fig. 2.1.7 Diseño interno del servomotor de lego NXT

Equipos adicionales utilizados

Pic16F628A

- Es un microcontrolador CMOS FLASH de 8 bits
- Capaz de operar con frecuencias de reloj hasta de 20 MHz .
- Posee internamente un oscilador de 4 MHz.





Equipos adicionales utilizados

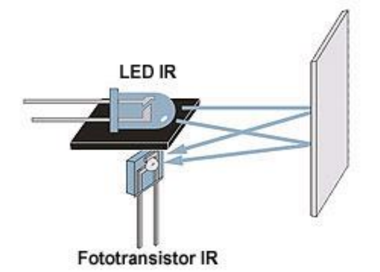
Servomotor Hitec HS311

- Los servos son un tipo especial de motor de D.C. que se caracterizan por su capacidad para posicionarse de forma inmediata en cualquier posición dentro de su intervalo de operación.
- El servomotor espera un tren de pulsos que se corresponde con el movimiento a realizar.

Equipos adicionales utilizados

Sensor de luz Infrarrojo

- Están diseñados especialmente para la detección, clasificación y posicionado de objetos; la detección de formas, colores diferencias de superficie, incluso bajo condiciones ambientales extremas.
- En su interior contiene dos elementos, uno es un diodo emisor de luz infrarroja y el otro un fototransistor, este último mide la cantidad de luz infrarroja en algún objeto, esta medición depende de la claridad del objeto a sensar.



Herramientas de Software

Programación de Lego Mindstorms NXT

- La programación del Lego Mindstorms se realiza mediante el software que se adjunta en el empaque original, el cual trae el firmware del robot y un programa que emula un árbol de decisiones.

- Una de las principales características de este software de programación, es su entorno visual, el cual emula la construcción por bloques, dando la posibilidad a cualquier usuario aprendiz acostumbrarse rápidamente a la programación de bloque.

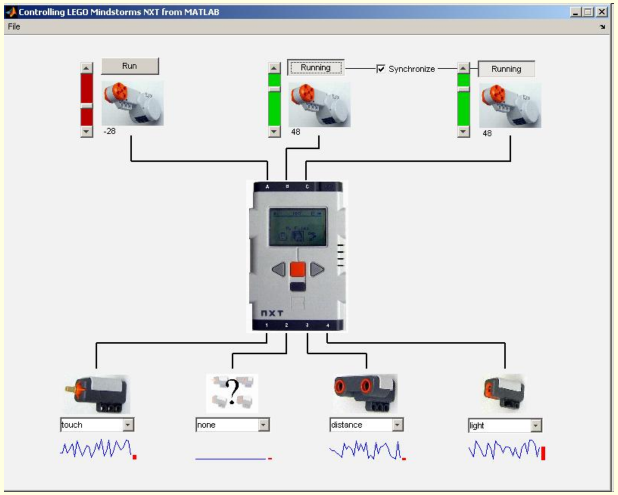
Herramientas de Software

Matlab y Simulink

- Es un ambiente de cómputo, de alta ejecución numérica y de visualización.

- MATLAB integra el análisis numérico, calculo de matrices, procesamiento de señales, diseño de sistemas de potencia, Mapeo y tratamiento de imágenes, instrumentación y adquisición de datos.

- Simulink es una herramienta para el modelaje, análisis y simulación de una amplia variedad de sistemas físicos y matemáticos.



Herramientas de Software

Cygwin

Nos servirá para ejecutar instrucciones en un entorno similar LINUX, es básicamente un simulador de UNIX que ejecuta los programas GNU ARM y Nexttool.

GNU ARM

Es un compilador que convierte los archivos creados desde matlab (archivos *.m) o cualquier otro programador que use lenguaje C o C++.

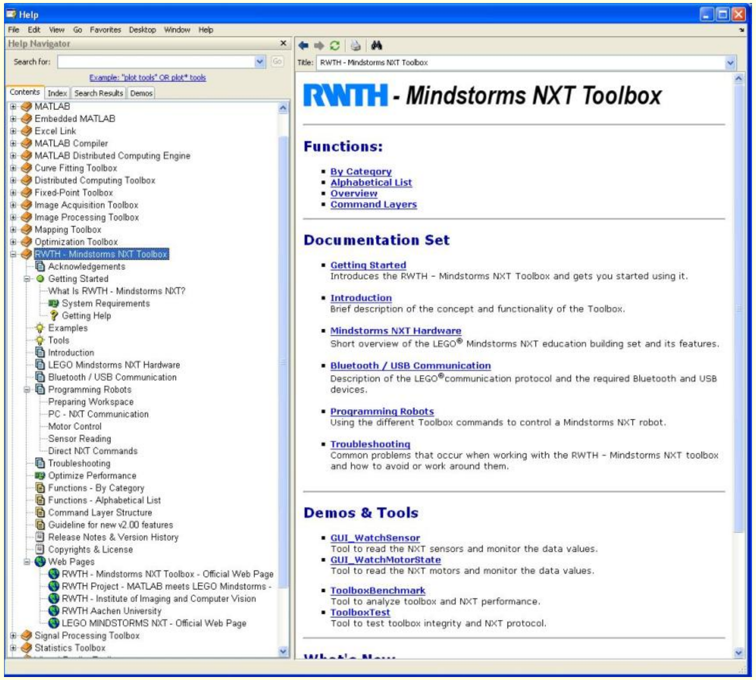
NXTOSEK

Es una plataforma de código abierto para LEGO MINDSTORMS NXT.

Herramientas de Software

RWTHMindstormNXT y Embedded Coder Robot NXT

Es un toolbox que ha sido desarrollado para el control de LEGO MINDSTORMS robots NXT con MATLAB a través de una conexión inalámbrica Bluetooth o vía USB. Embedded Coder Robot NXT es la librería que proporciona los bloques de Mindstorm NXT para desarrollar un modelo en Simulink.



RWTH - *Mindstorms NXT Toolbox*

Functions:

- By Category
- Alphabetical List
- Overview
- Command Layers

Documentation Set

- Getting Started
  Introduces the RWTH - Mindstorms NXT Toolbox and gets you started using it.
- Introduction
  Brief description of the concept and functionality of the Toolbox.
- Mindstorms NXT Hardware
  Short overview of the LEGO® Mindstorms NXT education building set and its features.
- Bluetooth / USB Communication
  Description of the LEGO®communication protocol and the required Bluetooth and USB devices.
- Programming Robots
  Using the different Toolbox commands to control a Mindstorms NXT robot.
- Troubleshooting
  Common problems that occur when working with the RWTH - Mindstorms NXT toolbox and how to avoid or work around them.

Demos & Tools

- GUI_WatchSensor
  Tool to read the NXT sensors and monitor the data values.
- GUI_WatchMotorState
  Tool to read the NXT motors and monitor the data values.
- ToolboxBenchmark
  Tool to analyze toolbox and NXT performance.
- ToolboxTest
  Tool to test toolbox integrity and NXT protocol.

Herramientas de Software

MikroBasic

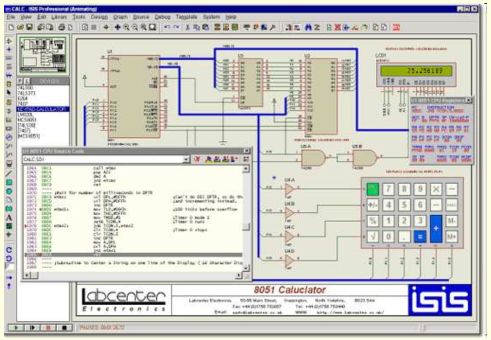
- Es una herramienta de desarrollo que se permite realizar proyectos para microcontroladores PIC.
- Proporciona una solución fácil para aplicaciones, sin comprometer el rendimiento o el control, desarrolla rápidamente y desplegar aplicaciones complejas.

Herramientas de Software

Proteus

- Es un paquete de software para el diseño de circuitos electrónicos que incluye captura de los esquemas, simulación analógica y digital combinadas y diseño de circuitos impresos.

- El paquete está compuesto por dos programas: ISIS, para la captura y simulación de circuitos; y ARES, para el diseño de PCB's.

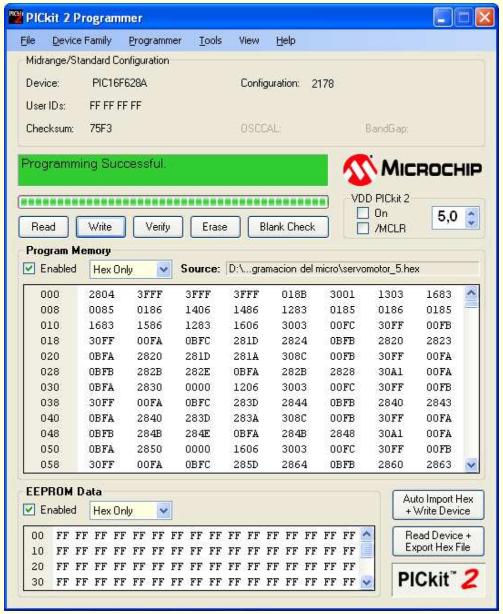


Herramientas de Software

Programador de Microcontroladores PIC Kit 2

- El PICkit 2 es un programador fabricado por Microchip para programar toda su línea de microcontroladores.

- El programador fue diseñado para programar los microcontroladores en circuito (ICSP) lo que significa que puede programar los microcontroladores montados directamente en tu aplicación y/o protoboard.



Midrange/Standard Configuration

Device: PIC16F628A

Configuration: 2178

User IDs: FF FF FF FF

Checksum: 75F3

OSCCAL:

BandGap:

Programming Successful.

VDD PICkit 2

☐ On

☐ /MCLR

5,0

Program Memory

☑ Enabled Hex Only

Source: D:\...gramacion del micro\servomotor_5.hex

```
000   2804  3FFF  3FFF  3FFF  018B  3001  1303  1683
008   0085  0186  1406  1486  1283  0185  0186  0185
010   1683  1586  1283  1606  3003  00FC  30FF  00FB
018   30FF  00FA  0BFC  281D  2824  0BFB  2820  2823
020   0BFA  2820  281D  281A  308C  00FB  30FF  00FA
028   0BFB  282B  282E  0BFA  282B  2828  30A1  00FA
030   0BFA  2830  0000  1206  3003  00FC  30FF  00FB
038   30FF  00FA  0BFC  283D  2844  0BFB  2840  2843
040   0BFA  2840  283D  283A  308C  00FB  30FF  00FA
048   0BFB  284B  284E  0BFA  284B  2848  30A1  00FA
050   0BFA  2850  0000  1606  3003  00FC  30FF  00FB
058   30FF  00FA  0BFC  285D  2864  0BFB  2860  2863
```

EEPROM Data

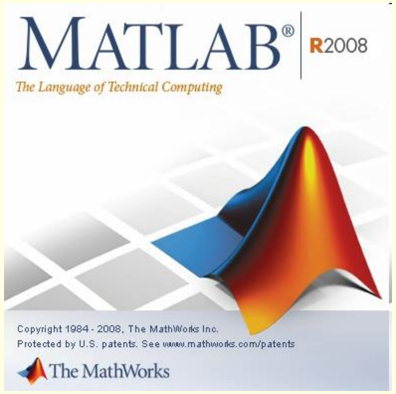
☑ Enabled Hex Only

```
00  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
10  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
20  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
30  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
```

Implementación del proyecto

- El desarrollo del proyecto consistió en 2 etapas, una de ellas el la construcción de un prototipo utilizando las herramientas de Lego Mindstorms,

- La segunda etapa consiste en la mejora del prototipo, pero esta vez utilizando las herramientas de software de Matlab y Simulink, con el fin de obtener mediciones en tiempo real de los sensores, visualizar estos valores en una escala de tiempo y tomar decisiones desde el entorno de Matlab.



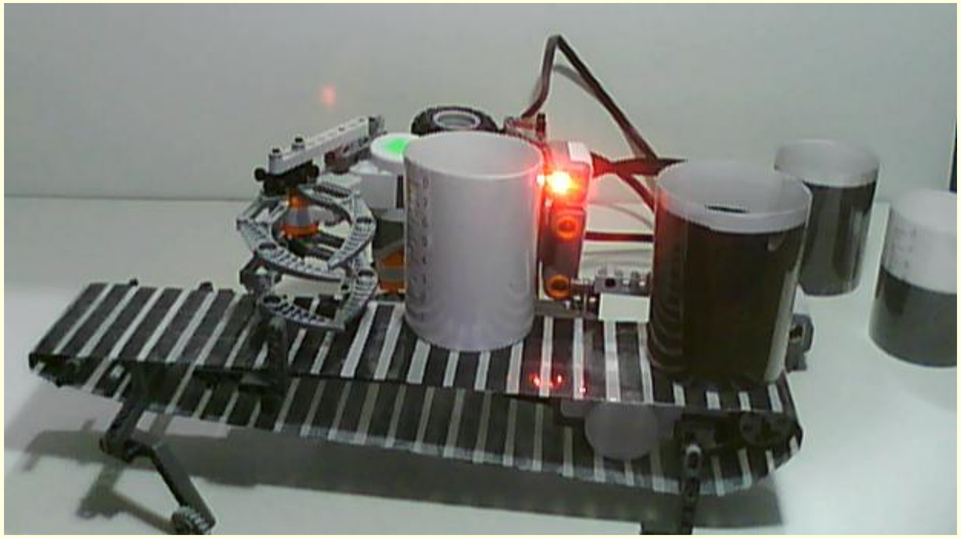
Prototipo Inicial

Utilizando el principio del llenado de botellas de cola, diseñamos un robot capaz de seleccionar botellas completamente llenas de otras que no contengan el nivel exacto de líquido, esto puede ser utilizado en el control de calidad para un proceso de llenado de botellas.

Prototipo Inicial

Construcción

La primera versión de la banda consistía en un brazo selector de envases con diferentes niveles de líquido, estos niveles eran representados franjas de colores en unos recipientes transparentes, es decir el sensor de luz se lo utilizó para medir estos dos colores (banco y negro).

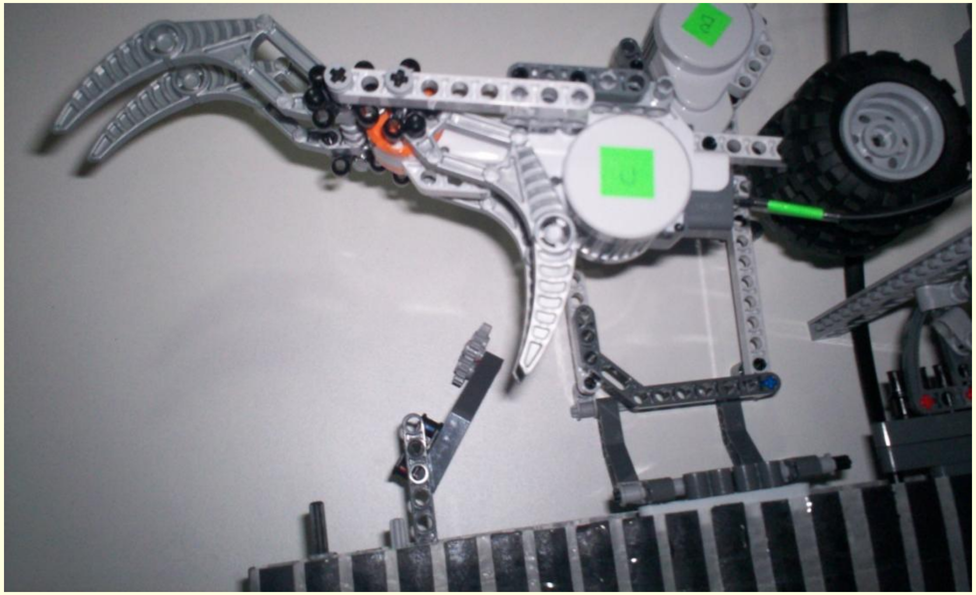


- Fig. 3.1.1 Pruebas de lectura utilizando sensor infrarrojo

Prototipo Inicial

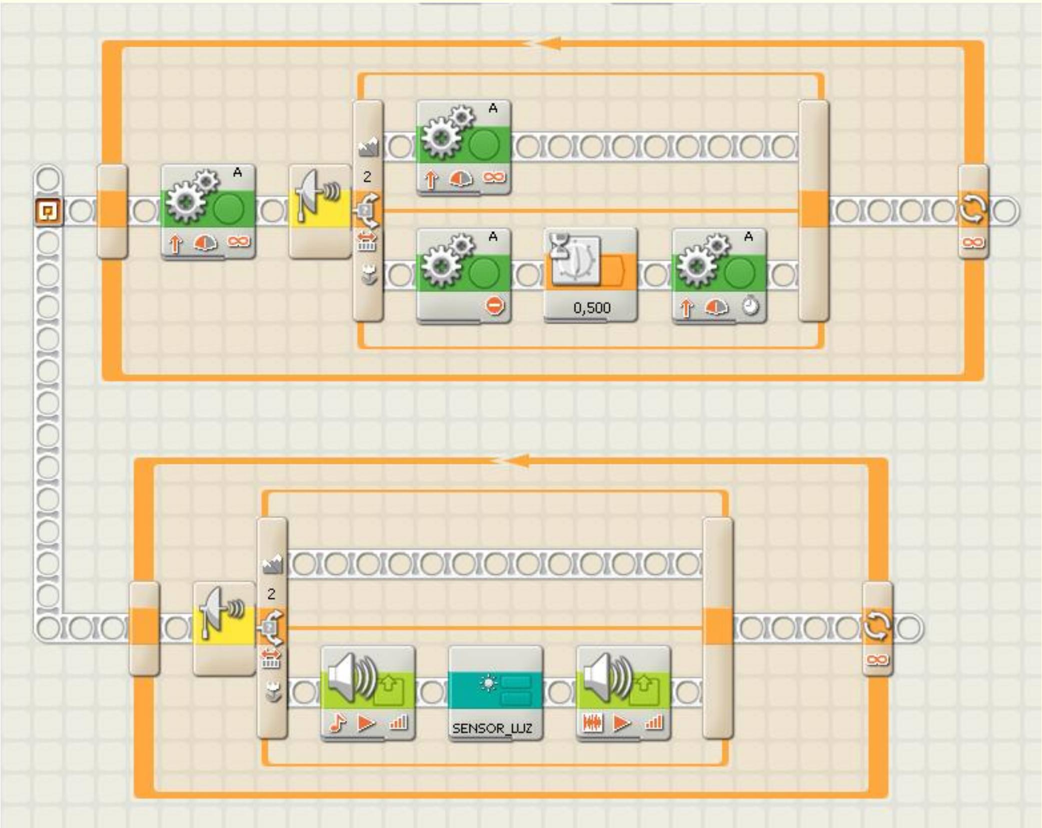
- El brazo robótico solo tiene 2 motores, uno controla el gripper y el otro mueve el eje del brazo, mientras que la banda transportadora es controlada por un tercer motor

  - Fig. 3.1.2 Garra controlada por servomotor



Prototipo Inicial

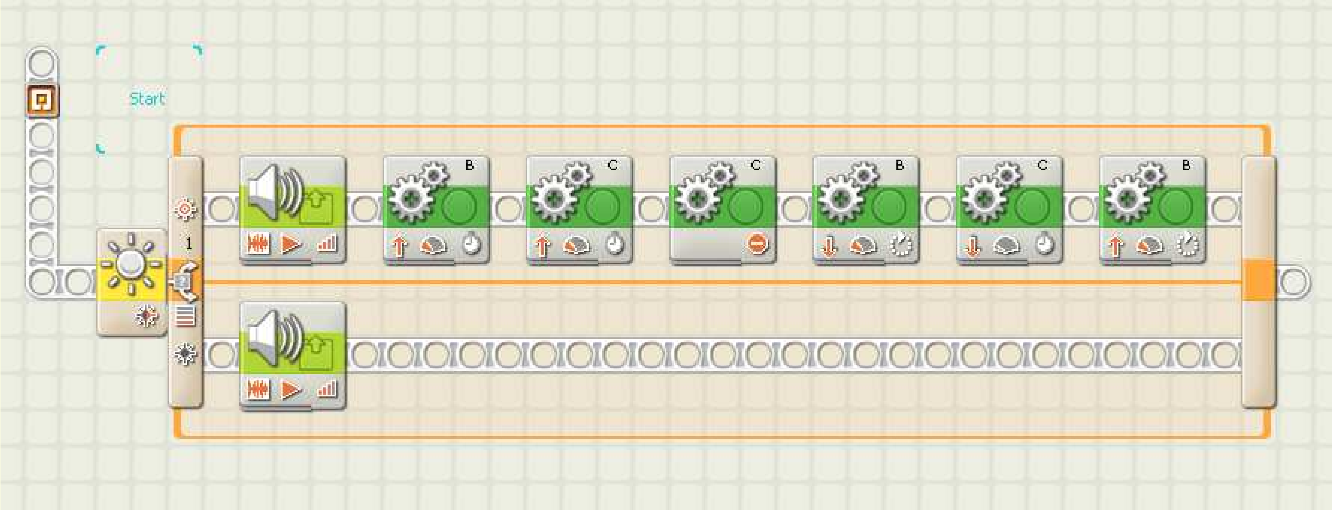
Programación:



- Fig. 3.1.3 Programación en Lego Mindstorms de la banda Transportadora

Prototipo Inicial

- Creamos un bloque llamado SENSOR_LUZ, aquí verificamos el nivel de líquido en la botella, si el nivel es el requerido, simplemente la banda continua girando, en caso de detectar un nivel inferior, activamos en brazo robótico



- Fig. 3.1.4 Programación en Lego Mindstorms del Brazo Robótico

Prototipo Inicial

Proyecto Final

- Luego de la construcción y programación del prototipo, notamos ciertos errores en el diseño del mismo, ya que el sensor de ultrasonido lo utilizábamos para medir presencia de objetos y no para su función básica que es medir distancias, entonces se decidió hacer un cambie en a ubicación de los sensores.

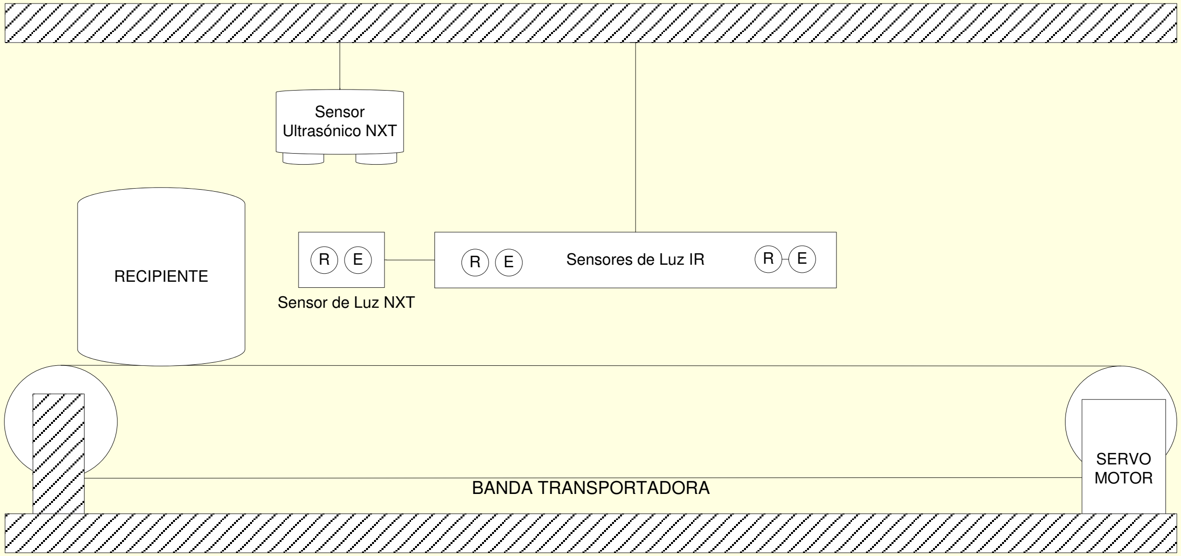


Fig. 3.2.1: Esquema, ubicación de los sensores y motores.

Diseño electrónico del controlador de la Banda.

Circuito del Sensor de luz

- Para obtener señales analógicas en el microcontrolador, proveniente de los sensores de luz, utilizaremos un amplificador operacional, para nuestro caso elegimos el LM324, el cual contiene 4 Opams (Amplificadores Operacionales) en su interior.

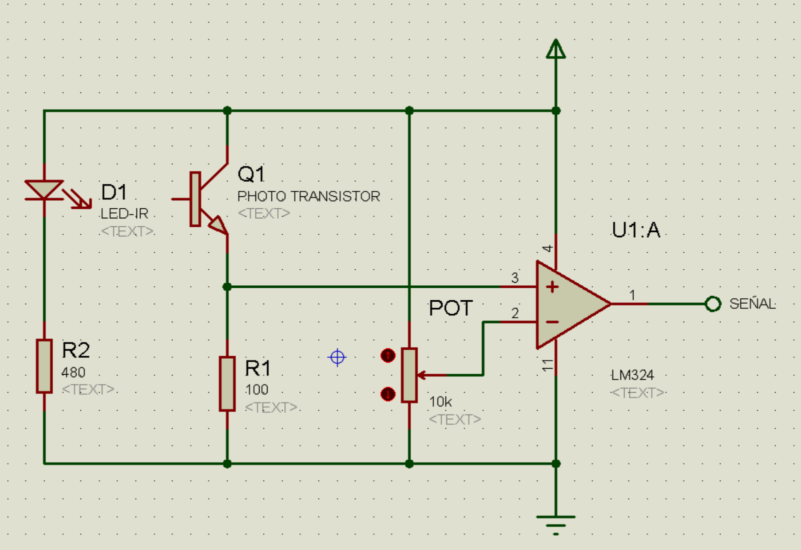
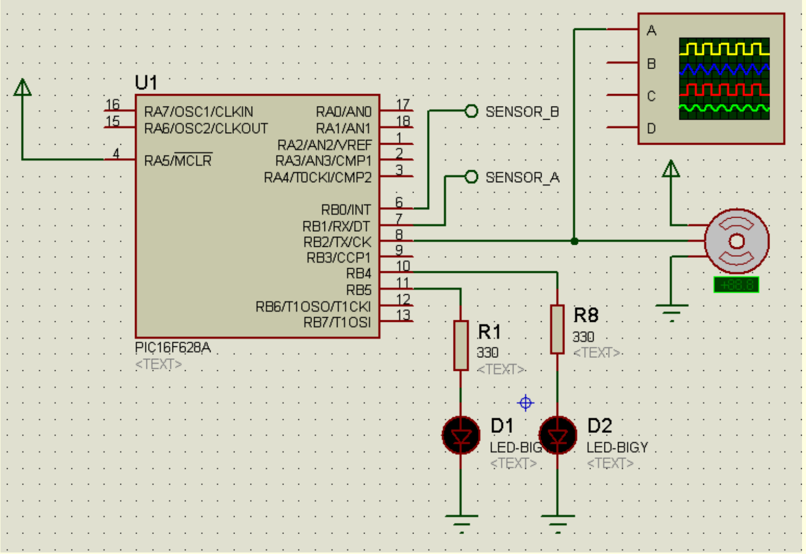


Fig. 3.4.1: Circuito acondicionador de un sensor infrarrojo.

Diseño electrónico del controlador de la Banda.

Circuito del Microcontrolador

- Para controlar la banda utilizamos el PIC16F628A, al tener muy pocas entradas y salidas que manejar se eligió este microcontrolador, también por su facilidad al momento de armar el circuito ya que este PIC posee un oscilador interno.

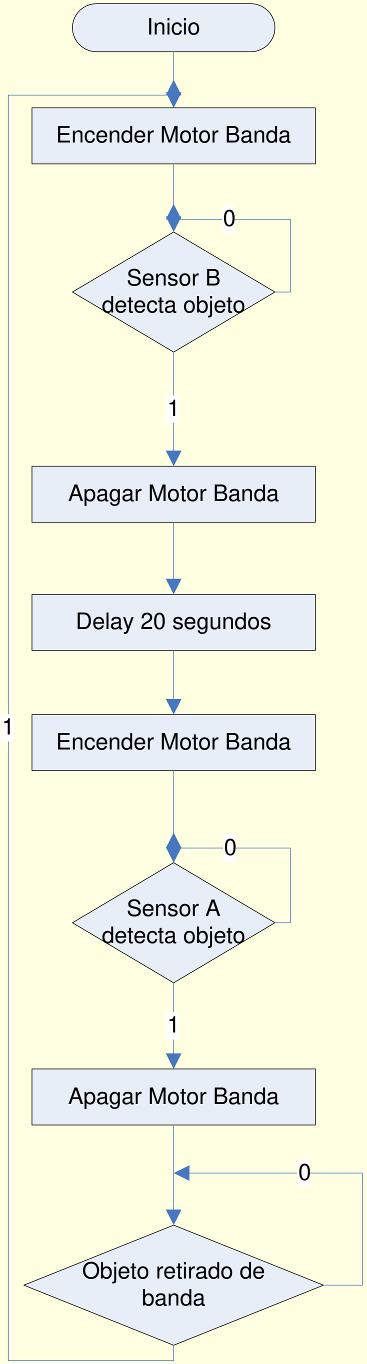


- Fig. 3.4.2: Circuito controlador de la Banda transportadora

Código de programa del microcontrolador

```
program servomotor
dim in as byte
dim Vel as byte
dim cont1 as word
Main:
INTCON = 0
TRISA = 1
TRISB = 0
TRISB.0 = 1
TRISB.1 = 1
PORTA = 0
PORTB = 0
PORTA = 0
PORTB.4 = 1
Delay_ms(500)
PORTB.4 = 0
Delay_ms(500)
PORTB.4 = 1
Delay_ms(500)
PORTB.4 = 0
Delay_ms(500)
PORTB.4 = 1
Delay_ms(500)
PORTB.4 = 0
Delay_ms(500)
```

```
While true
 if  (PORTB.0 = 0 ) then
     PORTB.4 = 1
     PORTB.2 = 1
     Delay_us(900)
     PORTB.2 = 0
     Delay_us(1100)

     if  (PORTB.1 = 1 ) then
        PORTB.5 = 1
        PORTB.2 = 0
        Delay_ms(2000)

              for cont1 = 0 to 700
             PORTB.2 = 1
             Delay_us(900)
             PORTB.2 = 0
             Delay_us(1100)
             next cont1
       else
       PORTB.5 = 0
     end if

   else
   PORTB.4 = 0
   Delay_ms(4000)
   PORTB.5 = 0

 end if
wend
 PORTB.2 = 0
 PORTB.4 = 1
 Delay_ms(2000)
 PORTB.4 = 0

End.
```



Inicio

Encender Motor Banda

Sensor B detecta objeto

0

1

Apagar Motor Banda

Delay 20 segundos

Encender Motor Banda

Sensor A detecta objeto

0

1

Apagar Motor Banda

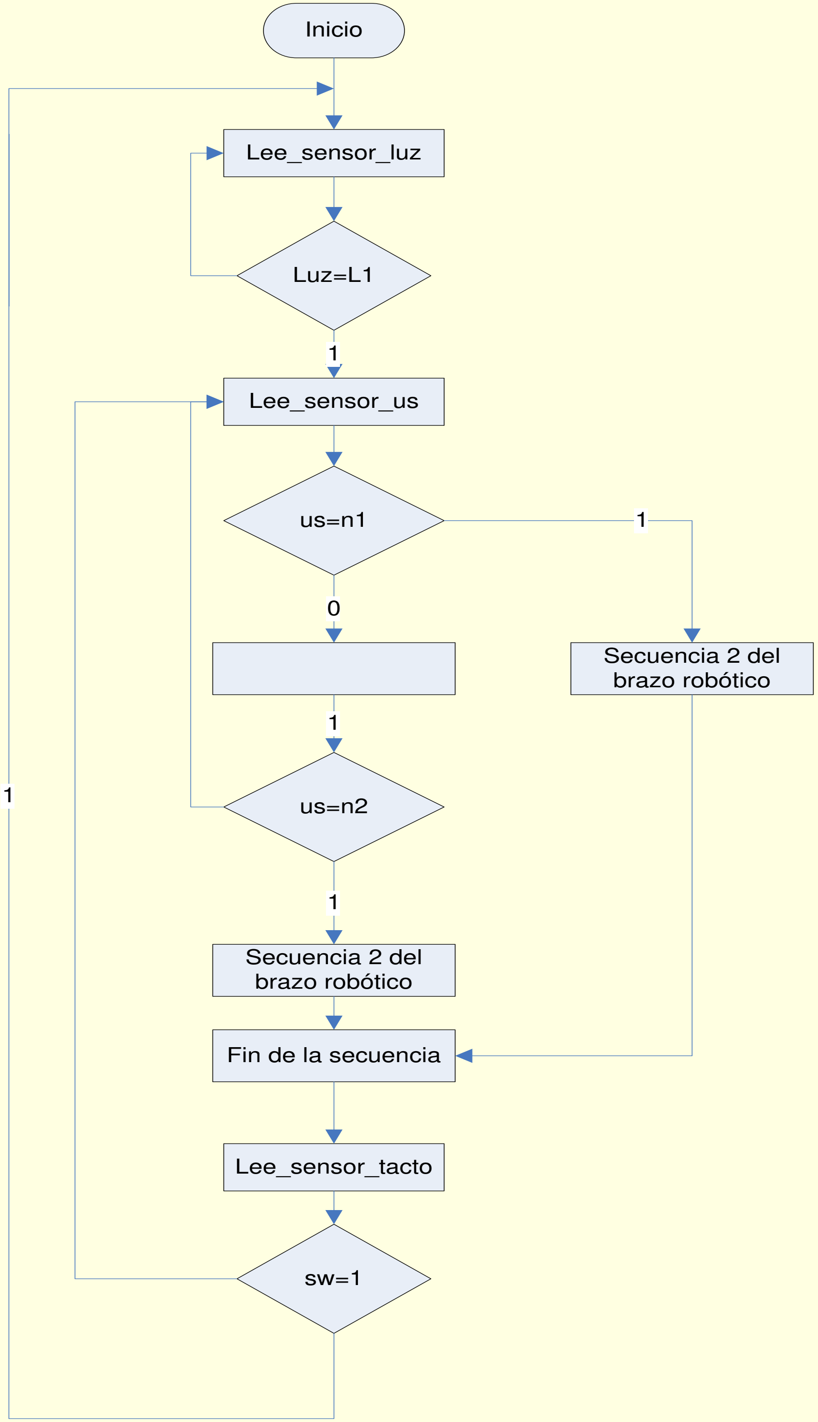
Objeto retirado de banda

0

1

PROGRAMACION EN MATLAB

DIAGRAMA DE FLUJO



PROGRAMACION EN MATLAB

- Rango Sensor ultrasónico: 0 – 255
- Rango Sensor de luz: 0 -1024
- Estado Sensor Tacto: 0 (off) – 1(on)

PROGRAMACION EN MATLAB

ENTRADAS NXT:

- Sensor Tacto = SENSOR_1
- Sensor Luz = SENSOR_3
- Sensor Ultrasonido = SENSOR_4

SALIDAS NXT:

- puerto1 = MOTOR_A;
- puerto2 = MOTOR_B;
- puerto3 = MOTOR_C;

PROGRAMACION EN MATLAB

CONFIGURACION SE SENSORES:

- **Sensor Luz:**

  Light > 400 » Presencia de Recipiente
- **Sensor Ultrasonido:**

  US < 7 » Nivel Recipiente Correcto

  US > 8 » Nivel Recipiente Incorrecto
- **Sensor Tacto:**

  SW = 1 » Se detiene Secuencia Brazo

PROGRAMACION EN MATLAB

COMANDOS ULIZADOS

- h = COM_OpenNXT()
- OpenLight()
- GetLight()
- OpenUltrasonic()
- GetUltrasonic()
- USMakeSnapshot()
- USGetSnapshotResults()

PROGRAMACION EN MATLAB

- OpenSwitch()
- GetSwitch()
- SetMotor()
- SetPower()
- SetAngleLimit ()
- NXT_PlayTone(Hz,time)
- CloseSensor();
- COM_CloseNXT();
- xlswrite (‘dta.xls’,name)

Simulación y Pruebas

El proyecto final esta equipado con 5 sensores distribuidos en diferentes posiciones, los sensores del Lego NXT presentan digitales, en el caso del sensor de ultrasonido 8 bits de resolución y el sensor de Luz 10 bits.

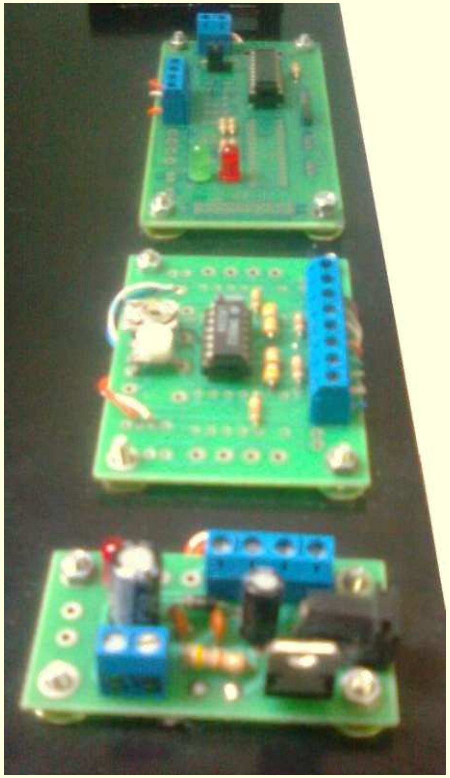
Fig. 4.1.1 Prueba de funcionamiento del nuevo Brazo robótico

Configuración y Funcionamiento del equipo

Tarjetas electrónicas (PCBs)

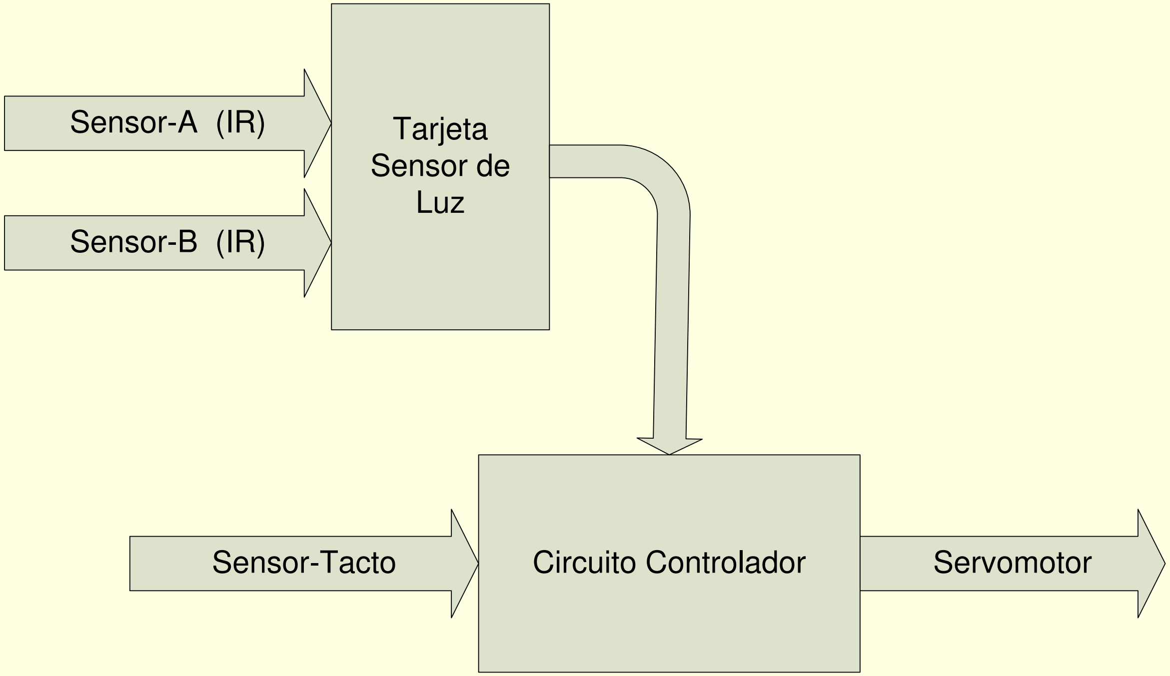
Básicamente el control de la banda contiene 3 tarjetas electrónicas independientes: Tarjeta Controladoras, Sensor Infrarrojo y fuente de Poder.

Los elementos de entrada al microcontrolador son 2 sensores infrarrojos y un sensor de tacto, mientras que la salida es un servomotor.



Configuración y Funcionamiento del equipo

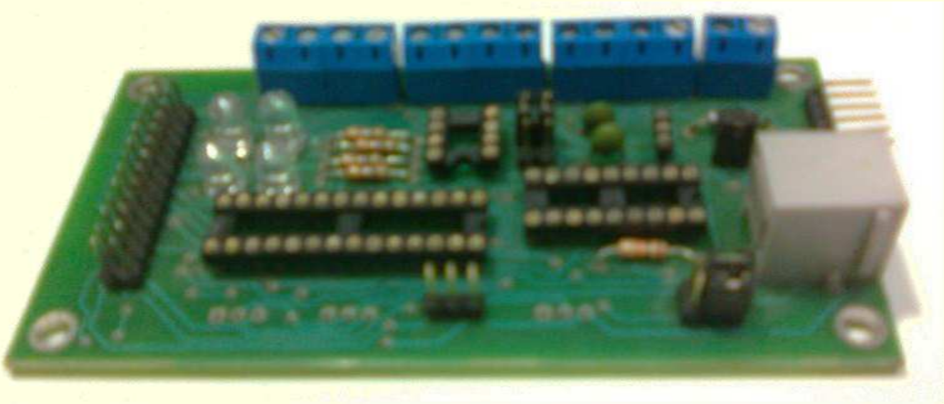
- **Fig. 4.1.4 Esquema de conexión entre las tarjetas electrónicas**



Configuración y Funcionamiento del equipo

Circuito Controlador

- El controlador está montado sobre la tarjeta universal Ctrl Pics V1.1, la cual me permite programar y probar los microcontroladores de microchip 16F84A, 16F628A, 16F876 y 16F88X.

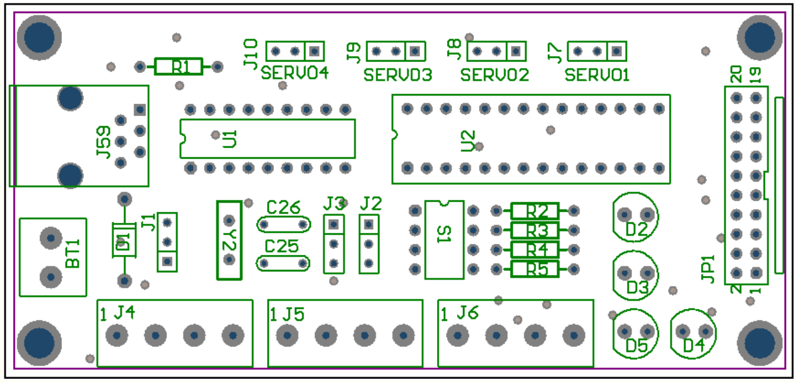


- Fig. 4.1.5a tarjeta universal Ctrl. Pics V1.1

Configuración y Funcionamiento del equipo

- **Fig. 4.1.5 Componentes de la tarjeta universal Ctrl. Pics V1.1**

Nomencl.	Tipo	Descripción
BT1	Bornera doble	Alimentación 5VDC
R1	Resistencia	33KOhm
U1	Zócalo 18 pines	Pic16F628A
J1	Jumper	Selector
R2, R3	Resistencia	220Ohm
D2, D3	Diodo Led	Puerto RB5, RB6
J5	Bornera 4 pines	Puerto RB0, RB1, RB2 y RB3
J8	3 Espadines	Servomotor RB2

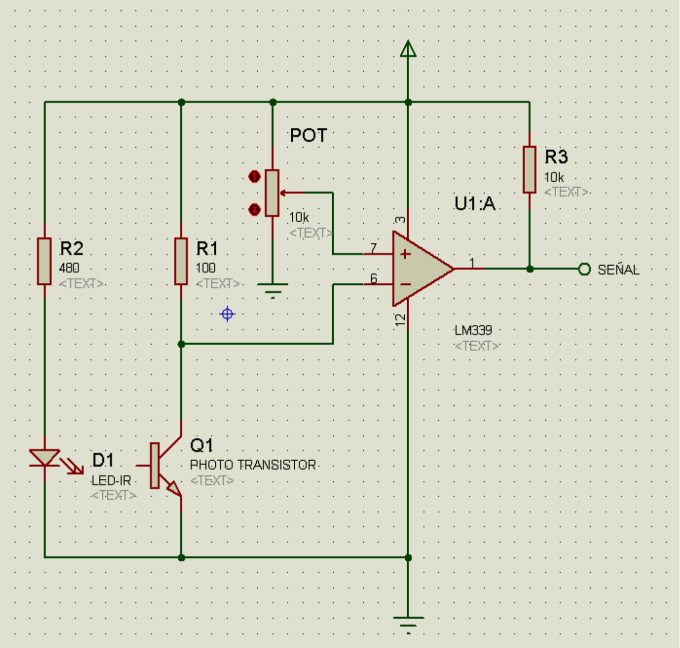


Configuración y Funcionamiento del equipo

Circuito Sensor Infrarrojo

- Esta tarjeta está diseñada para manejar hasta 4 sensores infrarrojos, en su interior contiene un amplificador operacional, el LM339N, este integrado trabaja como un comparador. La señal proveniente de Los fototransistores varia entre 2 y 4.7 voltios dependiendo del brillo de los objetos a medir.

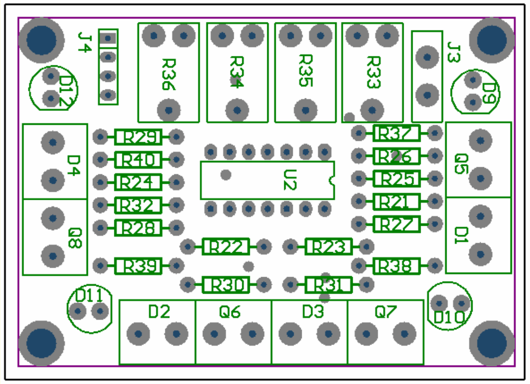
Fig. 4.1.8: Circuito del Sensor Infrarrojo



Configuración y Funcionamiento del equipo

Nomencl.	Tipo	Descripción
R27, R28, R31 y R30	Resistencia	10 KOhm
R23 y R22	Resistencia	470 Ohm
R34 y R35	Potenciómetro	10 KOhm
J4	Sip-4	Salidas del Opam
J3	Bornera doble	Alimentación 5Vdc
D2 y D3	Bornera doble	Diodos Infrarrojos
Q6 y Q7	Bornera doble	Fototransistores
U2	Zócalo 14 pines	LM339N

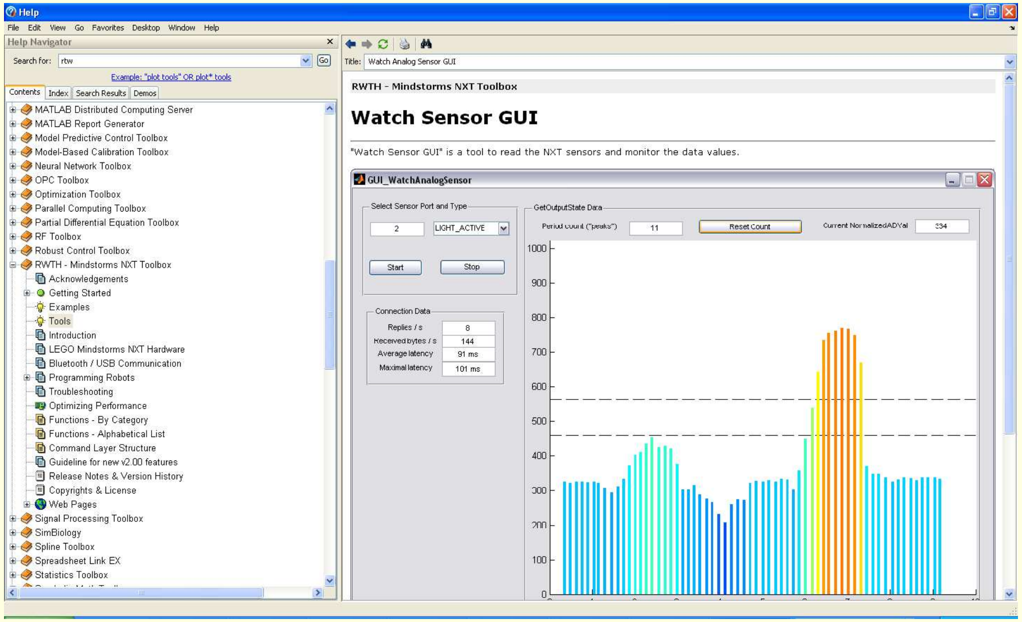
Fig. 4.1.9 Componentes de la tarjeta Sensor Luz V4.0



Seteo y Configuración de los sensores.

Calibración de Sensores del NXT

Dentro del toolbox de Matlab RWTH - Mindstorms NXT se encuentra la herramienta Watch Sensor GUI, esto nos permite una visualización gráfica de los valores analógicos medidos por los sensores de Lego NXT.



Title: Watch Analog Sensor GUI

RWTH - Mindstorms NXT Toolbox

Watch Sensor GUI

"Watch Sensor GUI" is a tool to read the NXT sensors and monitor the data values.

GUI_WatchAnalogSensor

Select Sensor Port and Type: 2, LIGHT_ACTIVE

Connection Data
Replies / s: 8
Received bytes / s: 144
Average latency: 91 ms
Maximal latency: 101 ms

GetOutputState Data
Period count ("peaks"): 11
Current NormalizedADVal: 334

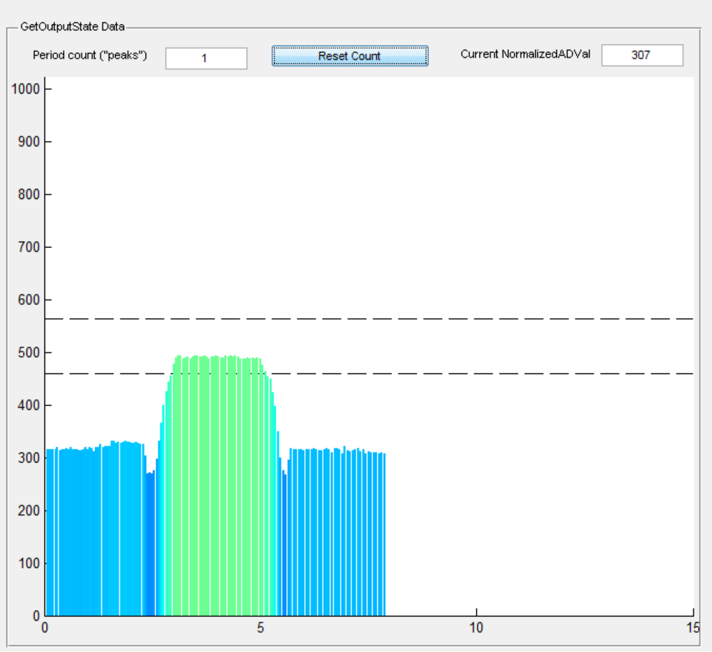
- Fig. 4.2.1 Help de Matlab en la librería RWTH - Mindstorms NXT

Seteo y Configuración de los sensores.

- El proyecto contiene 2 sensores, uno infrarrojo y otro de Ultrasonido.

- El sensor de luz tiene una resolución de 10 bits, es decir presenta valores de 0 a 1124, se lo ha conectado en el puerto 3 del NXT. En la figura 4.2.2 se presentan valores inferiores a 350 bits, todo lo que esta por debajo de este valor significa que no hay presencia de algún objeto frente al sensor.

Fig. 4.2.2 Rango de Valores de sensor de luz



Period count ("peaks") 1

Current NormalizedADVal 307

Seteo y Configuración de los sensores.

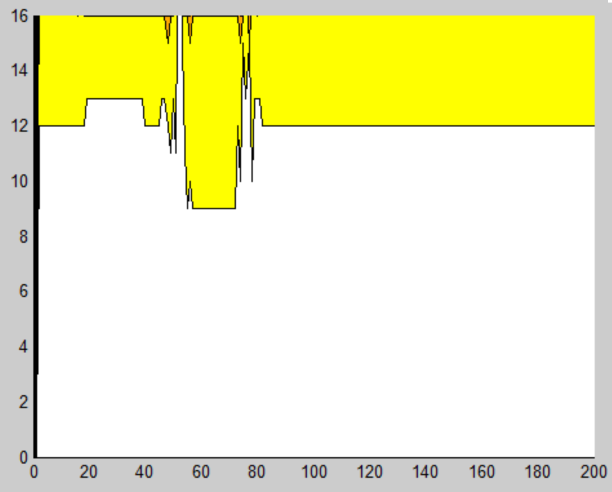


Fig. 4.2.3 Rango de Valores, Sensor Ultrasónico en nivel de producto Vacío

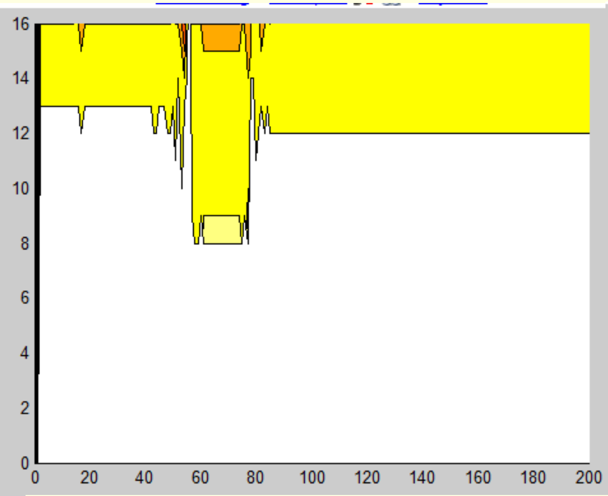


Fig. 4.2.4 Rango de Valores, Sensor Ultrasónico en nivel de producto Intermedio

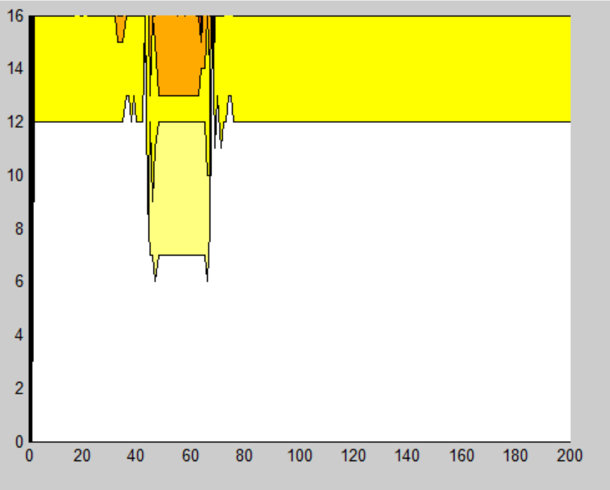


Fig. 4.2.5 Rango de Valores, Sensor Ultrasónico en nivel de producto Lleno

- El sensor Ultrasónico tiene 8 bits de resolución, es decir que presenta valores entre 0 a 256, para el proyecto utilizaremos envases con tres niveles de producto, vacío, medio y lleno respectivamente.
- El sensor está clocado a una altura de 9.5cm. de la banda transportadora.

Niveles	Valor Medido (Bits)
Cero	11
Vacío	9
Medio	8
Lleno	7

Seteo y Configuración de los sensores.

Calibración de Sensores de la banda transportadora

- Para el control de banda transportadora se utilizaron 2 diodos emisores de luz infrarroja y 2 fototransistores, con ello se diseño el circuito de sensores de luz, mencionado en la sección 3.4.1.

Tabla 4.2.2: Valores medidos en el sensor de Ultrasonido

Sensor	Sin presencia de Objeto (Vdc)	Con presencia de Objeto (Vdc)
A	4.86	3.67
B	4.88	3.71

Conclusiones y Recomendaciones

- Se usó Matlab para desarrollar la programación de la secuencia del Brazo Robótico y de igual manera para obtener la adquisición de datos del sensor de ultrasonido en tiempo real.

- Para la alimentación de la tarjeta de control de la Banda Trasportadora, se diseñó una fuente independiente, con este circuito también se consiguió tener la corriente necesaria de 180 mA para mover el servomotor debido que la alimentación que proporciona Lego Mindstorms NXT no es la requerida.

- Programar la secuencia del brazo robótico en Matlab, utilizando los comandos de rotación para mover los servomotores, los cuales dan un valor exacto de giro para el servo, ya que si es realizado con comandos de tiempo presenta margen de error en cuanto al posicionamiento del brazo.

Conclusiones y Recomendaciones

- Para calibrar el sensor de luz, tomar en cuenta que ésta es afectada por la luz del ambiente ya que el sensor del NXT funciona con luz visible roja, por lo que puede dar error al momento de la calibración.

- Se recomienda trabajar con sensores de luz infrarroja ya que estos no son afectados por la luz del ambiente.