

# **Control de Robot con sensores de distancia**



# OBJETIVOS

- › Aprender a utilizar y manipular de manera sencilla las rutinas de programación en la plataforma AVR Studio 4.
- › Diseño e implementación de un código óptimo para mantener equidistancia a referencia móvil.
- › Aprovechar todos los recursos que posee el POLOLU 3PI.
- › Uso adecuado de los sensores de distancia de SHARP.



# INTRODUCCIÓN

- › Cámaras de video, sensores no visuales, el radar, los sensores inerciales y los de activación por presión.
- › Sistema de detección de obstáculos



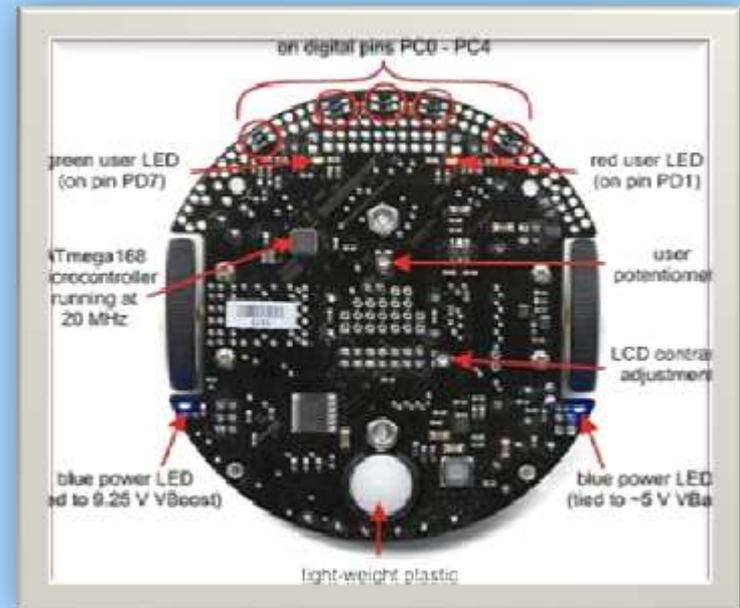
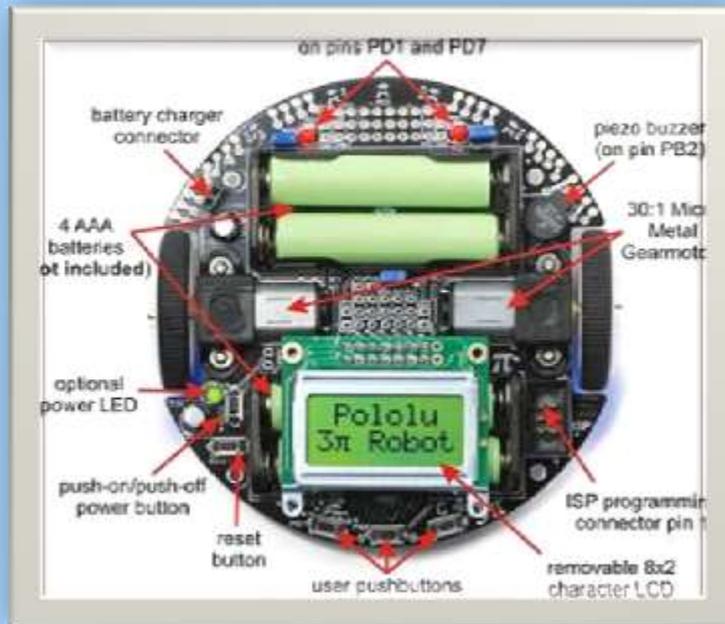
# DESCRIPCION DEL PROYECTO

- › Determinar la aplicación ejecutada por el Robot Pololu 3pi a desarrollar para este proyecto
- › Control de un robot Pololu con el microcontrolador ATmega328p
- › El funcionamiento del sistema se basa principalmente en las señales analógicas enviadas por los sensores



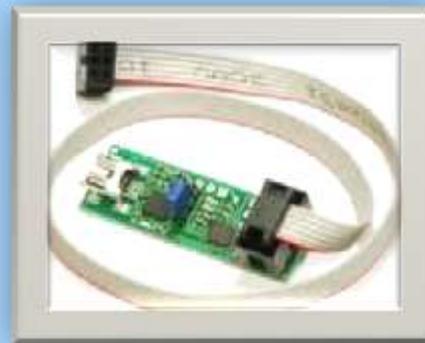
# HARDWARE DEL PROYECTO

## › ROBOT POLOLU 3PI



# CARACTERISTICAS POLOLU 3PI

- ❑ 4 pilas AAA
- ❑ 2 Motores que trabajan a 9.25V
- ❑ 5 sensores IR
- ❑ 1 pantalla LCD de 8x2 caracteres
- ❑ 1 alarma
- ❑ 3 botones.
- ❑ Conector ISP
- ❑ Entre otros



# HARDWARE DEL PROYECTO

## › SENSOR ANALÓGICO SHARP



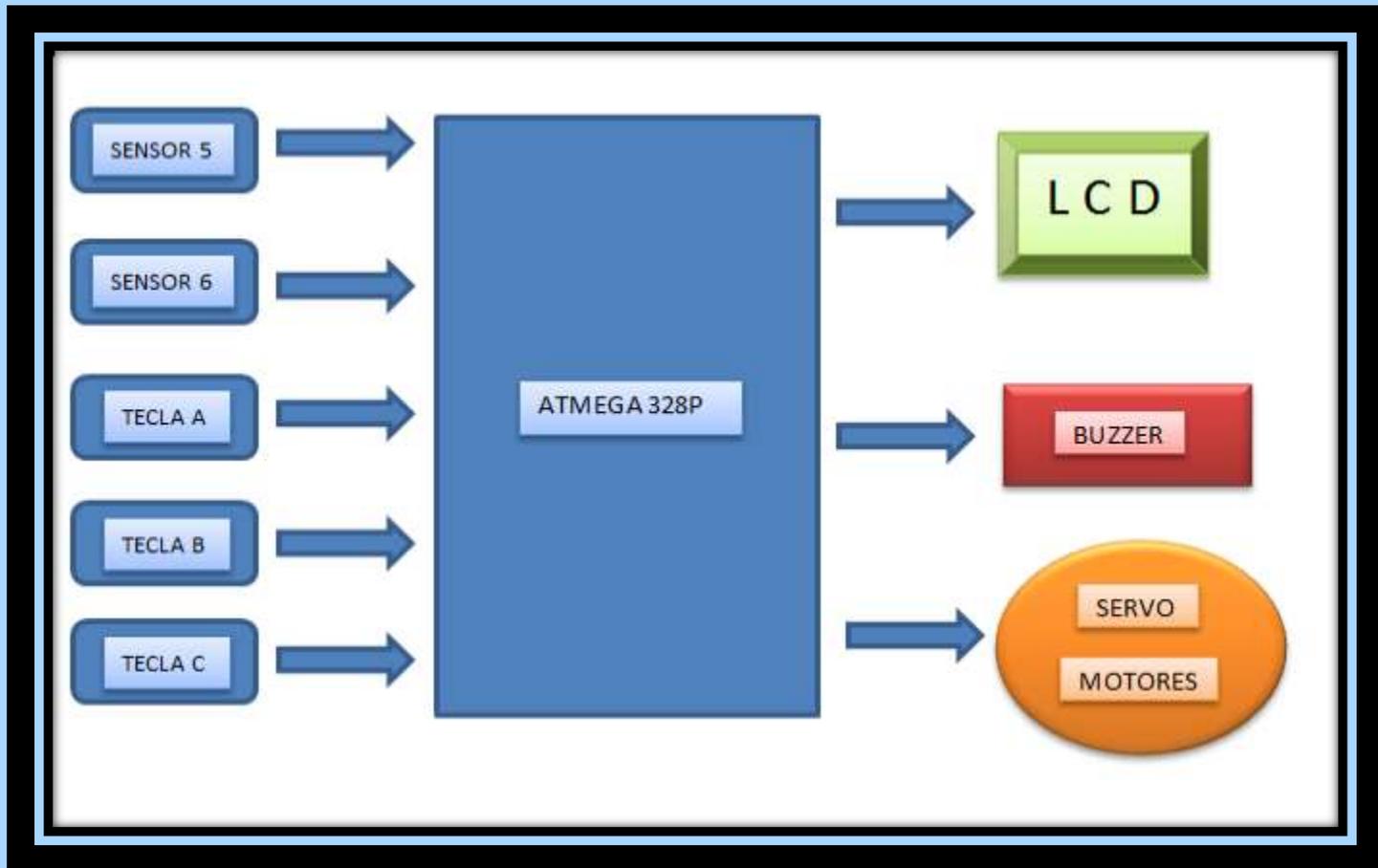
# CARACTERISTICAS SENSORES SHARP



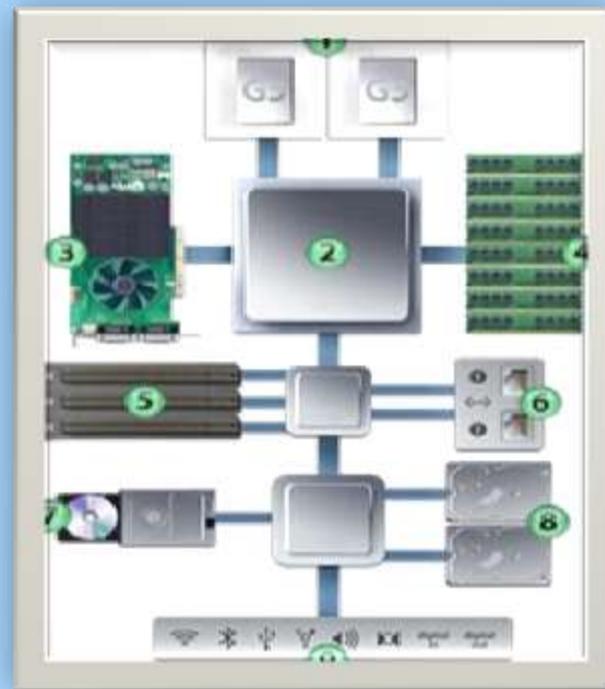
Los Sensores Sharp GP2Y0A21YK tienen características como:

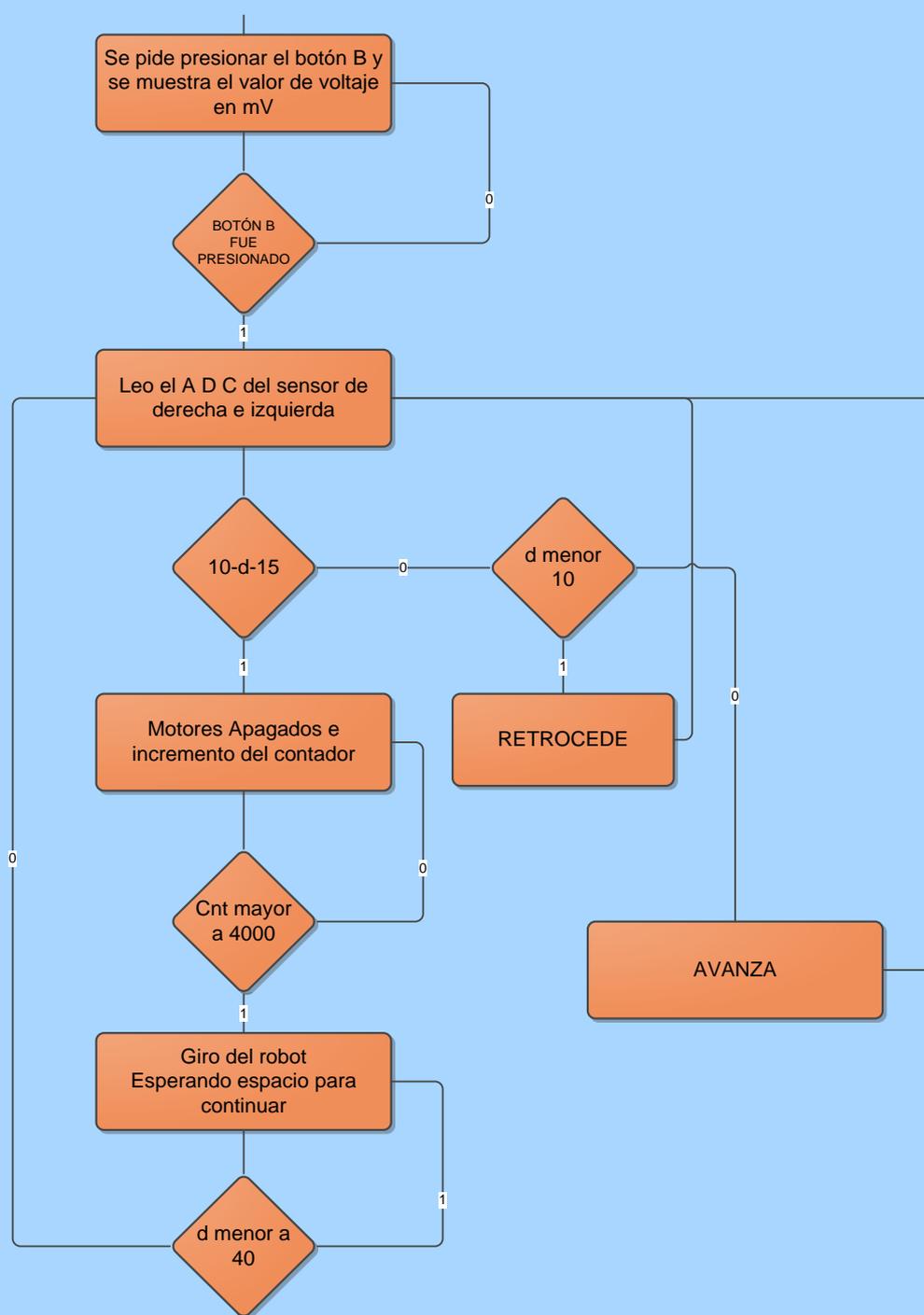
- Rango de detección
- La interfaz adaptable
- Salida analógica solo puede conectarse a un convertidor de analógico a digital
- Salida puede ser conectado a un comparador de umbral de detección
- Los sensores de distancia Sharp son una opción popular para muchos proyectos que requieren mediciones precisas de distancia
- Sensor de infrarrojos muy económico

# DIAGRAMA DE BLOQUES



# DIAGRAMA DE FLUJO





# DESCRIPCION DEL SOFTWARE



```
#include <pololu/3pi.h>
```

**Librería**

```
//El robot avanza hasta que vea un obstaculo (Puerta)  
y se detiene a una distancia  
//aprox 12 cm, espera a que se abra la puerta para salir  
pero si en 4 seg no sucede  
//nada el robot gira 45 grados y sigue avanzando  
hasta que ocurra algo similar y repite  
//el proceso, en el momento que se encuentre con una  
puerta y esta se abra el robot  
//avanza hasta que recuperar la distancia de 12 cm  
pero si la puerta retrocede el robot  
//tambien retrocede para mantener la distancia inicial  
de 12 cm si durante este  
//tiempo de espera sea retrocediendo o avanzando  
pasan 4 segundos sin cambios el robot gira  
//360 grados (el giro siempre lo hace hacia su derecha)
```

**Breve  
descripción del  
proyecto**

```
int main()
{

int s1,s2,cnt,cnt2;

s1=0;    //valor del sensor de la derecha
s2=0;    //valor del sensor de la izquierda
cnt=0; //valor del contador del tiempo que espera
frente de la puerta
cnt2=0; //valor del contador para giro 500-45 1000-90
1500-135
```

**Declaración de  
variables  
utilizadas en el  
programa**

```
print("JESSICA");  
lcd_goto_xy(0,1);  
print("SAAVEDRA");  
delay_ms(1500);  
clear();  
print("JOSE");  
lcd_goto_xy(0,1);  
print("CHAVEZ");  
delay_ms(1500);  
clear();
```

**Mensaje de  
presentación  
que aparece en  
la pantalla LCD**

```
while(!button_is_pressed(BUTTON_B))
{
clear ();
print_long(read_battery_millivolts());
print("mV");
lcd_goto_xy(0,1);
print("Press B");
delay_ms(100);
}
```

```
//Limpia la pantalla para borrar el mensaje que dice  
presionar b  
clear();
```

***Espera que se presione  
el botón b que inicia  
todo el proceso***

***Mientras muestra el  
valor de voltaje  
presente en la  
batería***

```
//Emite un sonido de la escala do re mi... a volumen 10  
(v10) con retardo de corche(L8)  
play("L8 V10 crrdrerrfrgrarrbrr");
```

```
while(1){
```

```
//Lee el analogico en el puerto 6 y 5
```

```
s1= analog_read(6);  
s2= analog_read(5);
```

**Instrucción para  
añadir tonos al  
Pololu 3pi**

**Lectura de los  
valores obtenidos  
por los sensores  
para iniciar su  
movimiento**

```
//Emite un sonido de la escala do re mi... a volumen 10  
(v10) con retardo de corche(L8)  
play("L8 V10 crrdrerrfrgrarrbrr");
```

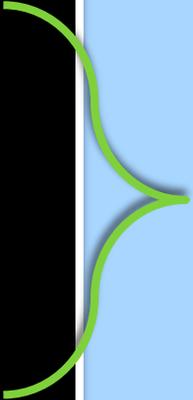
```
while(1){
```

```
//Lee el analogico en el puerto 6 y 5
```

```
s1= analog_read(6);  
s2= analog_read(5);
```



**Instrucción para  
añadir tonos al  
Pololu 3pi**



**Lectura de los  
valores obtenidos  
por los sensores  
para iniciar su  
movimiento**

```
print("s1 ");  
print_long(s1);  
lcd_goto_xy(0,1);  
print("s2 ");  
print_long(s2);
```

*//Si se encuentra a una distancia entre 10 a 15 cm se detiene a esperar*

```
while(s1<360 && s1>300){ //(d<15 && d>10) La relación es inversa
```

```
s1= analog_read(6);  
s2= analog_read(5);
```

**Imprime el valor de los sensores S1 y S2**

**Si se encuentra a una distancia entre 10 a 15 cm se detiene a esperar**

```
print("s1 ");  
print_long(s1);  
lcd_goto_xy(0,1);  
print("s2 ");  
print_long(s2);
```

```
set_motors(0,0);  
cnt++;  
cnt2=0;  
delay_ms(1);
```

***Imprime el valor  
de los sensores S1  
y S2***

```
if(cnt>=4000){
```

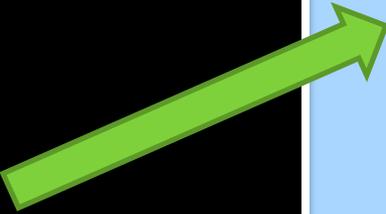
```
//Giro hasta detectar un espacio libre para seguir un camino
```

```
while(s1>100 && s2>100){
```

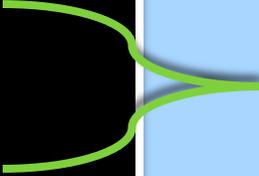
```
play("L8 V1o crdrerfrgrarbr");
```

```
s1= analog_read(6);
```

```
s2= analog_read(5);
```



**Instrucción para  
añadir tonos al  
Pololu 3pi**



**Lectura de los  
valores obtenidos  
por los sensores**

```
print("s1 ");  
print_long(s1);  
lcd_goto_xy(0,1);  
print("s2 ");  
print_long(s2);
```

**Imprime el valor  
de los sensores S1  
y S2**

```
delay_ms(100);  
//Signos diferentes para que gire  
set_motors(40,-40);
```

**Giro del Robot**

```
cnt2++;  
cnt=0;
```

```
~~~~~  
~~~~~  
~~~~~
```

```
if(s1<360 && s2<360){ //(d1>15 && d2>15)
s1= analog_read(6);
set_motors(40,40);
cnt=0;
}
```

```
if(s1>300){ \/(d1<10)
```

```
play("L8 V10 crrrerrrcrrrerrrcrrre");
```

```
s1= analog_read(6);
set_motors(-40,-40);
cnt=0;
}
```

**Si el sensor derecha e izquierda ve una distancia grande avanza**

**Si el sensor derecho ve una distancia corta retrocede para evitar colisiones mientras emite un sonido**

```
s1= analog_read(6);  
s2= analog_read(5);
```

```
print("s1 ");  
print_long(s1);  
lcd_goto_xy(0,1);  
print("s2 ");  
print_long(s2);
```

```
//Deja limpia la LCD  
clear();
```

```
}
```

```
return 0;
```

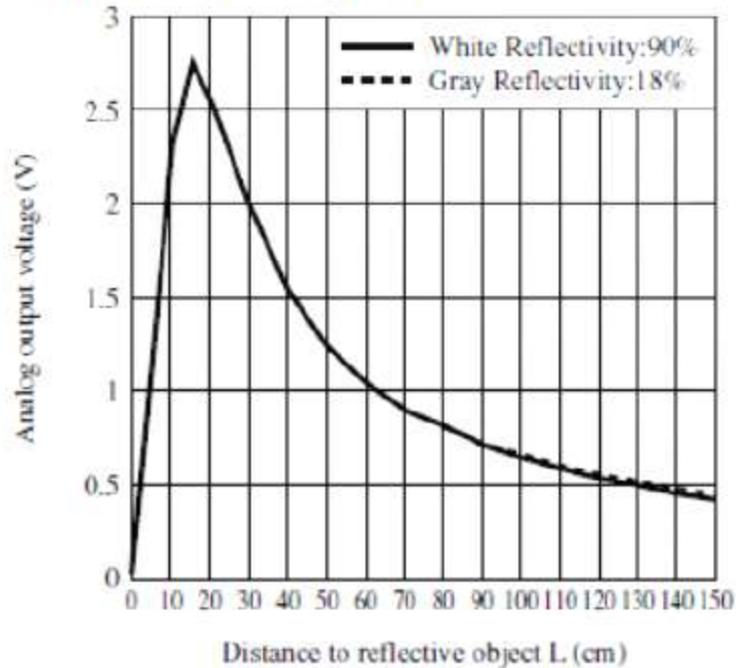
```
}
```

**Imprime el valor  
de los sensores S1  
y S2**

**Finalización del  
programa  
limpiando la  
pantalla**

# CURVA DE CONVERSION VOLTAGE ANALOGO VS DISTANCIA

**Fig.3 Analog Output Voltage vs. Distance to Reflective Object**



**Tabla de Conversión (Cm - ADC)**

Cm	ADC
1	204
2	307
3	409
4	512
5	614
10	471
20	266
30	184
40	153
50	122
60	102
70	81
80	61
90	40
100	20

# FUNCIONES PRINCIPALES

› `print("mensaje"):`



Permite presentar un mensaje en la pantalla LCD (8 columnas y 2 filas) del robot Pololu .

› `print_long ("valor de la distancia"):`

Permite mostrar el valor que se guarde en una variable.

› `set_motors(+/-vel, +/-vel):`



Modifica la velocidad de los motores de 600 a 0 rpm con solo darle a vel 255 para el primer caso ó "0" para el segundo caso, además se puede cambiar el giro cambiando su signo a negativo.

# FUNCIONES PRINCIPALES



› `play("Lx Vy caracteres"):`

Esta función permite emitir sonidos o series de sonidos con un control de tiempo  $x$  y volumen y de las notas musicales  $do=c$ ;  $re=d$  ;  $mi=e$ ;  $fa=f$ ;  $sol=g$ ;  $la=a$  ;  $si=b$ .

Y el corche "r" el cual permite hacerle un retardo o alargamiento a la nota que se coloque antes de esta.

# IMPLEMENTACION PRELIMINAR



- › La implementación del proyecto se la hará de la siguiente manera:
- › El Robot contará con la implementación de un control esquivador de objetos
- › El diseño de este proyecto se basa en controlar el Robot Pololu utilizando la ayuda de sensores de distancia trabajando a manera de convertidores de señal
- › El Pololu deberá conservar una equidistancia proporcionada por el usuario
- › Implementar rutinas para que el Robot pueda realizar un giro de búsqueda de salidas.
- › Programar la pantalla LCD del robot POLOLU para presentar datos específicos

# APLICACIONES PRINCIPALES

- › Entre las aplicaciones se puede observar el comercio, la seguridad de los autos y la exploración espacial
- › Pequeños vehículos autónomos enfocados hacia la minería y a exploración subterránea.
- › El campo de aplicación de este proyecto está relacionado con la industria automovilística.
- › Desarrollar un prototipo inteligente para poder evitar obstáculos y colisiones



# VENTAJAS PRINCIPALES



- ❑ Evita cualquier tipo de obstáculos: móviles o estáticos.
- ❑ Facilidad de busca de obstáculos gracias a los dos frentes de visión uno derecho y otro izquierdo para obstáculos presentes en estas direcciones.
- ❑ Búsqueda autónoma en su propio eje de hasta 360 grados para evitar obstáculos.

# DESVENTAJAS PRINCIPALES



- ❑ Al depender de los sensores el evadir obstáculos tiene un rango fijo de altura, es decir solo reconoce obstáculos a la altura que en que se ubiquen los sensores.
- ❑ Por su acción de evasión repentina el desplazamiento del robot es de mediana velocidad.
- ❑ Al ser energizado con baterías posee un tiempo limitado de funcionamiento de aproximadamente 4 horas de forma continua.

# CONCLUSIONES



- ❑ El proyecto realizado es un robot demostrativo el cual necesita mucho tiempo en su investigación, ya que es un ejemplar que contiene un sin numero de funciones, librerías y rutinas que sirven para mejorar su diseño y así convertirse en un producto de tipo comercial.
- ❑ Se logro diseñar un sistema que no solo detecta objetos u obstáculos fijos sino que además obstáculos móviles, es decir por medio de programación al robot Pololu se realiza un código para mantener siempre una distancia de modo que en su recorrido no permita ningún tipo de colisión.
- ❑ Se puede tomar a este prototipo como una base para futuros proyectos, similares en los cuales se puede mejorar la forma de detectar obstáculos y evitarlos usando nuevas tecnologías. Ya que este tipo de tecnología usada representa un auxiliar en cuanto a la prevención de accidentes, facilitando el desplazamiento de un sitio a otro con comodidad y tranquilidad.



# RECOMENDACIONES

- ❑ Se debe tomar en cuenta el medio donde se desarrolle el robot o donde se movilice, ya que los sensores son muy sensibles a la luz y eso impediría el fácil movimiento del robot, esto hace referencia a los objetos que el robot esquivaría.
- ❑ Es importante tener claro que el Pololu solo cuenta con 2 sensores frontales es decir se complicaría el movimiento al retroceder ya que primero debería dar el giro y luego el movimiento, ya que no cuenta con un sensor en la parte de atrás, es decir es preferible un lugar sin muchos obstáculos en la parte posterior.
- ❑ La velocidad a la que el robot se mueve debido a que los sensores necesitan un tiempo para cambiar su valor en la curva, el ADC haga la conversión y siga las ordenes del algoritmo.



# BIBLIOGRAFIA

- › 1).- Página principal donde se describe al pololu 3pi: partes, funciones, precios y guías que ayudaran a programar el robot pololu 3pi.
  - › <http://www.pololu.com>
  - › Fecha de consulta: 25/04/11.
  
- › 2).- Guía de usuario del robot pololu el cual da pautas para verificar el pololu asi como el de instalar software y drives que se usen para la programación del robot,
  - › [www.pololu.com/file/0J137/Pololu3piRobotGuiaUsuario.pdf](http://www.pololu.com/file/0J137/Pololu3piRobotGuiaUsuario.pdf)
  - › Fecha de consulta: 25/04/11.
  
- › 3).- Descripción y especificaciones breves del módulo Orangután,
  - › <http://www.pololu.com/catalog/product/1227/specs>
  - › Fecha de consulta: 25/04/11.
  - ›
  
- › 4).- Recursos dados para la utilización de las librerías, programadores para el pololu 3 pi,
  - › <http://www.pololu.com/catalog/product/1227/resources>
  - › Fecha de consulta: 27/04/11.

# GRACIAS POR SU ATENCION



# SEMINARIO DE GRADUACION



Control de Robot Pololu con sensores de distancia para mantener equidistancia a referencia móvil

Integrantes:

Jessica Saavedra Castro

José Luis Chávez Aguilar





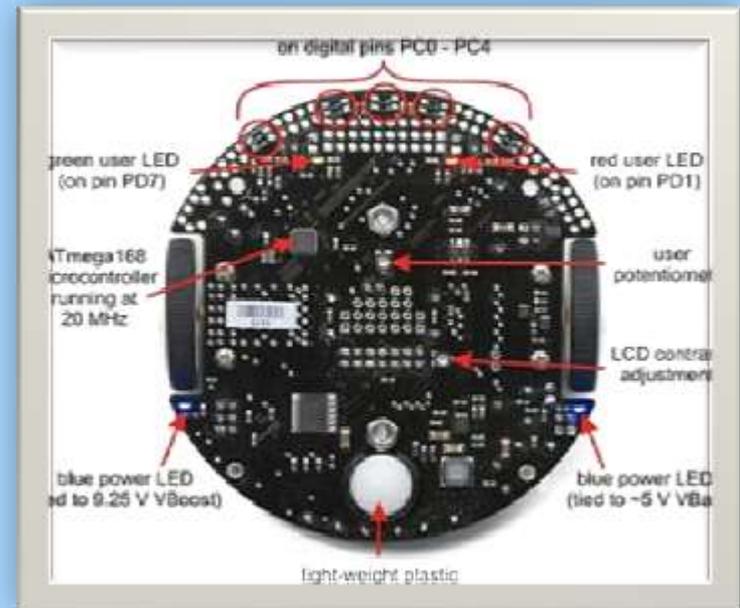
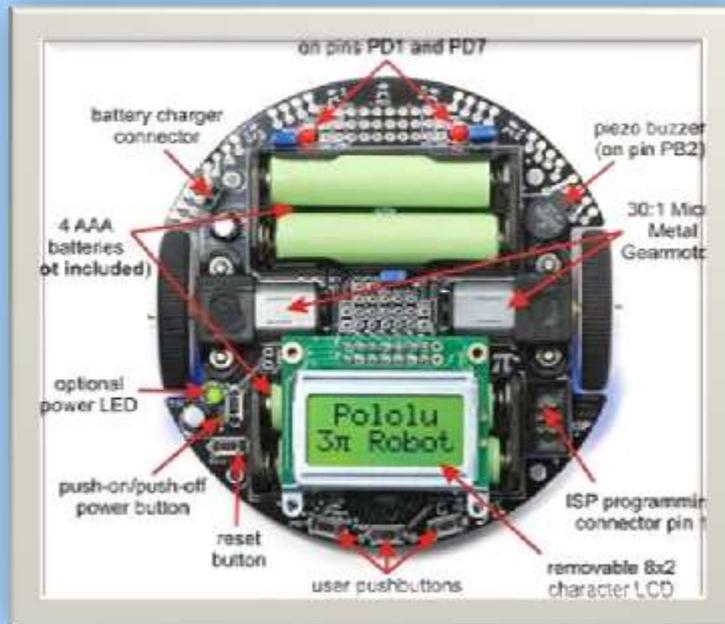
# OBJETIVOS

- › Aprender a utilizar y manipular de manera sencilla las rutinas de programación en la plataforma AVR Studio 4.
- › Diseño e implementación de un código óptimo para mantener equidistancia a referencia móvil.
- › Aprovechar todos los recursos que posee el POLOLU 3PI.
- › Uso adecuado de los sensores de distancia de SHARP.



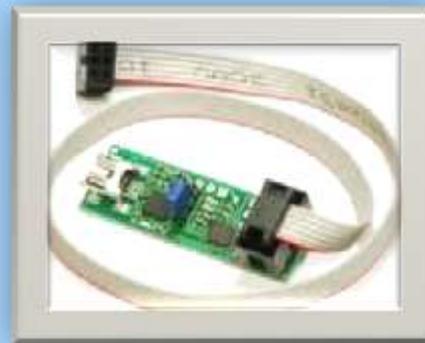
# HARDWARE DEL PROYECTO

## › ROBOT POLOLU 3PI



# CARACTERISTICAS POLOLU 3PI

- ❑ 4 pilas AAA
- ❑ 2 Motores que trabajan a 9.25V
- ❑ 5 sensores IR
- ❑ 1 pantalla LCD de 8x2 caracteres
- ❑ 1 alarma
- ❑ 3 botones.
- ❑ Conector ISP
- ❑ Entre otros



# HARDWARE DEL PROYECTO

## › SENSOR ANALÓGICO SHARP



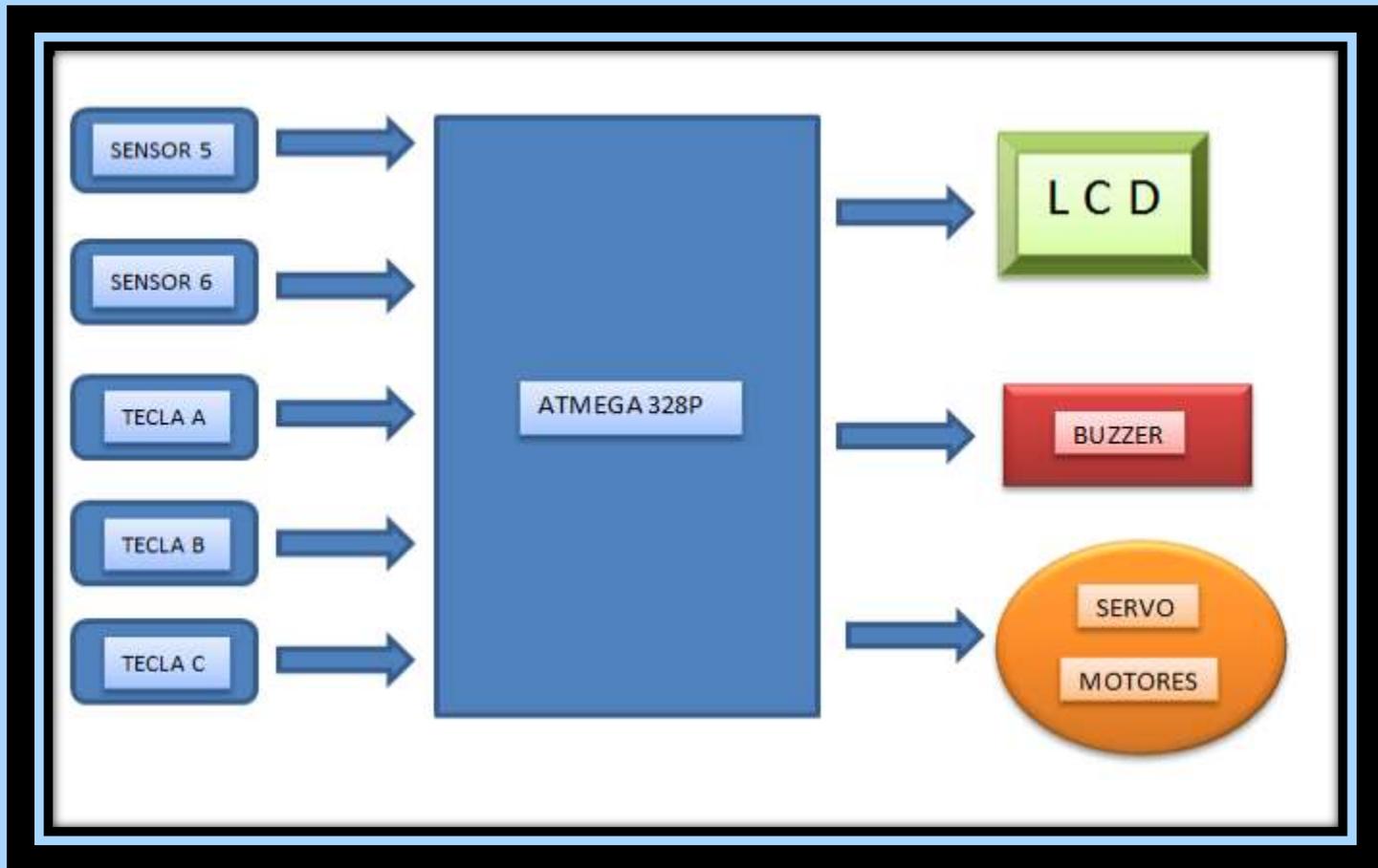
# CARACTERISTICAS SENSORES SHARP



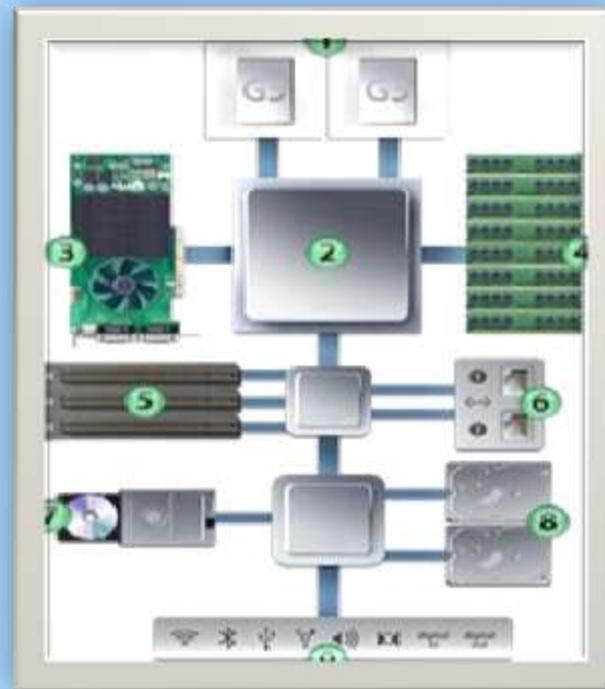
Los Sensores Sharp GP2Y0A21YK tienen características como:

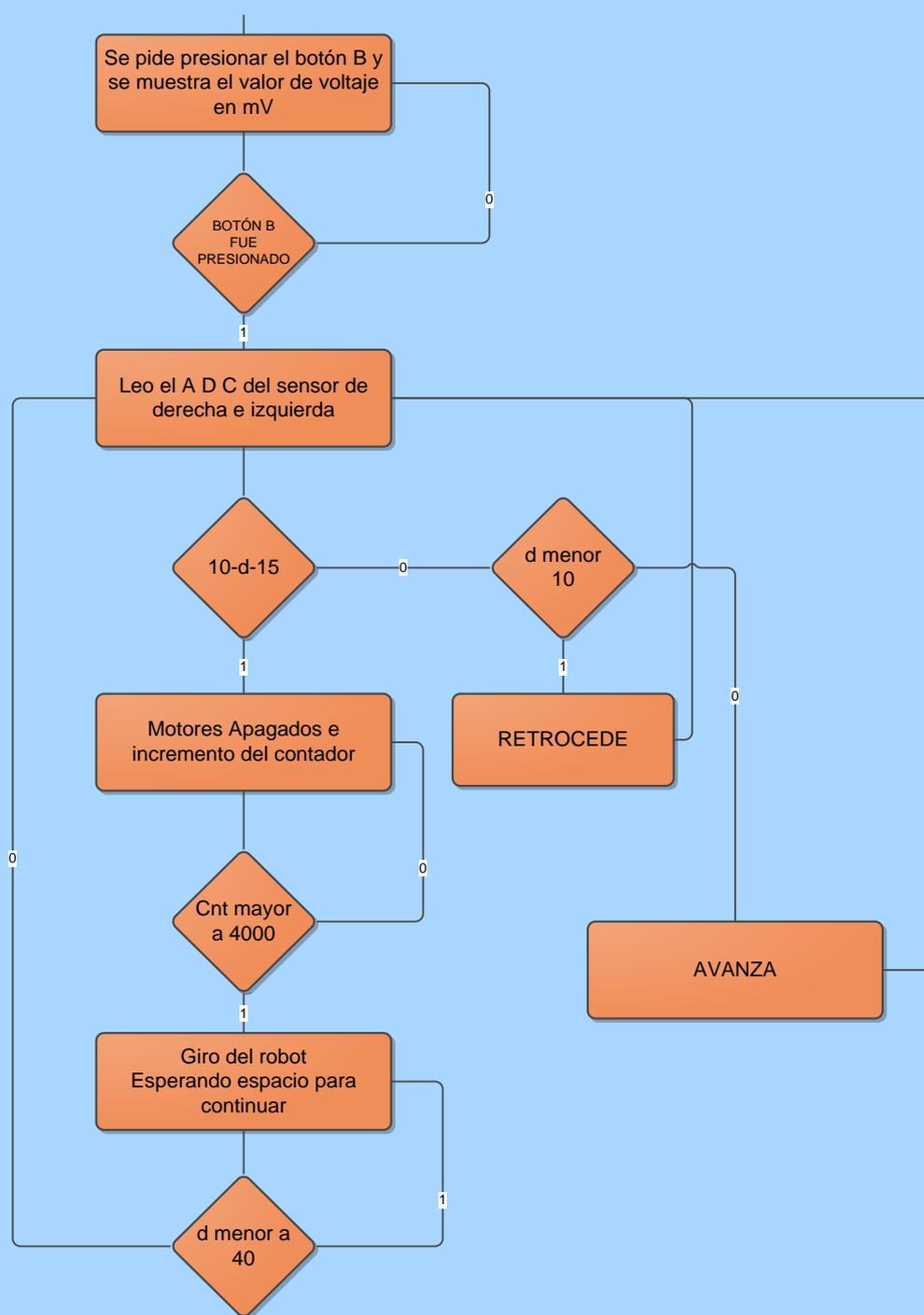
- Rango de detección
- La interfaz adaptable
- Salida analógica solo puede conectarse a un convertidor de analógico a digital
- Salida puede ser conectado a un comparador de umbral de detección
- Los sensores de distancia Sharp son una opción popular para muchos proyectos que requieren mediciones precisas de distancia
- Sensor de infrarrojos muy económico

# DIAGRAMA DE BLOQUES



# DIAGRAMA DE FLUJO





# CURVA DE CONVERSION VOLTAGE ANALOGO VS DISTANCIA

Fig.3 Analog Output Voltage vs. Distance to Reflective Object

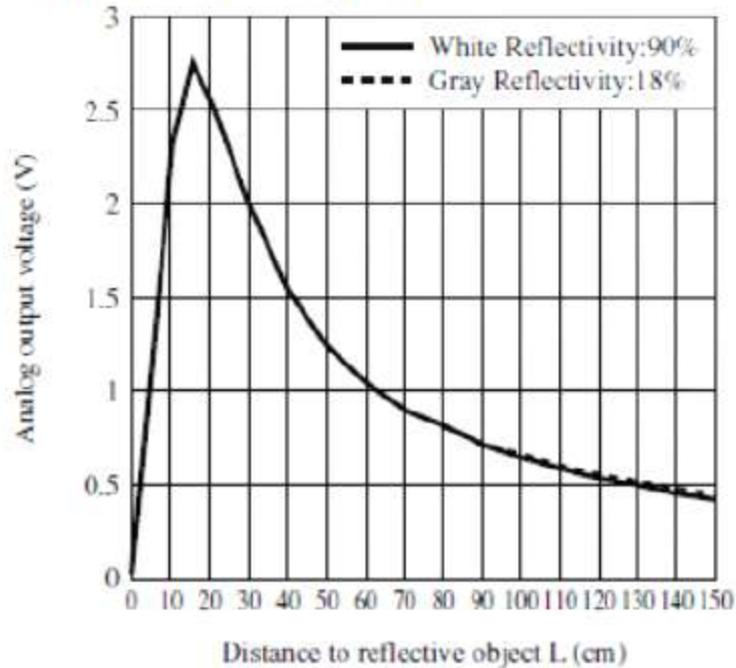


Tabla de Conversión (Cm - ADC)

Cm	ADC
1	204
2	307
3	409
4	512
5	614
10	471
20	266
30	184
40	153
50	122
60	102
70	81
80	61
90	40
100	20

# CONCLUSIONES



- ❑ El proyecto realizado es un robot demostrativo el cual necesita mucho tiempo en su investigación, ya que es un ejemplar que contiene un sin numero de funciones, librerías y rutinas que sirven para mejorar su diseño y así convertirse en un producto de tipo comercial.
- ❑ Se logro diseñar un sistema que no solo detecta objetos u obstáculos fijos sino que además obstáculos móviles, es decir por medio de programación al robot Pololu se realiza un código para mantener siempre una distancia de modo que en su recorrido no permita ningún tipo de colisión.
- ❑ Se puede tomar a este prototipo como una base para futuros proyectos, similares en los cuales se puede mejorar la forma de detectar obstáculos y evitarlos usando nuevas tecnologías. Ya que este tipo de tecnología usada representa un auxiliar en cuanto a la prevención de accidentes, facilitando el desplazamiento de un sitio a otro con comodidad y tranquilidad.



# RECOMENDACIONES

- ❑ Se debe tomar en cuenta el medio donde se desarrolle el robot o donde se movilice, ya que los sensores son muy sensibles a la luz y eso impediría el fácil movimiento del robot, esto hace referencia a los objetos que el robot esquivaría.
- ❑ Es importante tener claro que el Pololu solo cuenta con 2 sensores frontales es decir se complicaría el movimiento al retroceder ya que primero debería dar el giro y luego el movimiento, ya que no cuenta con un sensor en la parte de atrás, es decir es preferible un lugar sin muchos obstáculos en la parte posterior.
- ❑ La velocidad a la que el robot se mueve es debido a que los sensores necesitan un tiempo para cambiar su valor en la curva, el ADC haga la conversión y siga las ordenes del algoritmo.