



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

“Uso del sensor de distancia analógico SHARP GP2Y0A21YK0F en interfaz con el robot POLOLU 3PI en optimización de rutinas para evitar colisiones”

TESINA DE SEMINARIO

Previa la obtención del Título de:

INGENIERO EN
ELECTRÓNICA Y TELECOMUNICACIONES
INGENIERO EN ELECTRICIDAD
ESPECIALIZACIÓN ELECTRÓNICA Y AUTOMATIZACIÓN
INDUSTRIAL

Presentado por:

Angel Gustavo Naula Zhongor

Ronald David Solís Mesa

GUAYAQUIL – ECUADOR

AÑO 2011

AGRADECIMIENTO

A Dios.

A nuestra familia

A todas las personas que contribuyeron en el desarrollo de este trabajo.

A todos quienes apuestan por el desarrollo tecnológico en Ecuador.

Al Ing. Carlos Valdivieso.

A las personas que en conjunto forman a la ESPOL por brindarnos la oportunidad de adquirir conocimientos y formarnos profesionalmente.

DEDICATORIA

A Dios, nuestro creador por su amor y protección.

A nuestros padres, por sus sacrificios y desvelos, por los valores que nos inculcaron, por el ejemplo de tenacidad y perseverancia para alcanzar la meta anhelada.

A nuestra familia, por su afecto incondicional, estímulo permanente, apoyo y comprensión.

A nuestros maestros, por su paciencia, bondad y generosidad por iluminar nuestra senda con la luz del conocimiento, por enseñarnos a ser útiles a los demás.

A nuestros amigos y compañeros, por la alegría compartida, por ser espíritu de solidaridad, por animarnos constantemente hacia la culminación de nuestra carrera.

TRIBUNAL DE SUSTENTACION

A handwritten signature in black ink, appearing to read 'Carlos Valdivieso A.', written over a horizontal line.

Ing. Carlos Valdivieso A.

Profesor del Seminario de Graduación

A handwritten signature in black ink, appearing to read 'Hugo Villavicencio V.', written over a horizontal line.

Ing. Hugo Villavicencio V.

Delegado del Decano

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de este trabajo, nos corresponde exclusivamente; y el patrimonio intelectual del mismo a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL”.

(Reglamento de exámenes y títulos profesionales de la ESPOL)



Angel Gustavo Naula Zhongor

Ronald David Solís Mesa

RESUMEN

El objetivo de este proyecto de tesis es lograr la implementación de un sistema de protección del POLOLU 3PI, el cual pueda evitar las colisiones con obstáculos que se encuentren en el camino de su trayectoria reduciendo el riesgo de la pérdida material del equipo.

Lo que realiza este sistema es una es la detección de obstáculos que se encuentran en el camino del recorrido del robot POLOLU 3PI, a través de un sensor SHARP GP2Y0A21YK0F, el cual estará ubicado en el centro de la parte frontal del robot.

La ubicación de la posición de los sensores de distancia en frente del robot nos facilitará la correcta decisión cuando se pueda evitar un obstáculo debido a que si detecta un obstáculo podrá elegir entre girar a la izquierda o derecha siempre haciendo una correcta elección si se llegase a presentar que en una de esas direcciones estuviera algún obstáculo.

En el mercado existen sistemas de anti-colisiones que realizan una comparación similar en cuanto a la detección de los obstáculos, pero con sensores demasiado caros y con rutinas poco eficientes, por tal motivo se decidió implementar este proyecto como una solución moderna y de bajo costo y además que pueda ser compartida y se encuentre disponible en los laboratorios de electrónica de la FIEC.

ÍNDICE GENERAL

AGRADECIMIENTO.....	I
DEDICATORIA.....	II
TRIBUNAL DE SUSTENTACION	III
DECLARACIÓN EXPRESA.....	IV
RESUMEN.....	V
ÍNDICE GENERAL.....	VI
ÍNDICE DE FIGURAS	VIII
INTRODUCCIÓN.....	IX
CAPÍTULO I	1
1.-DESCRIPCIÓN GENERAL DEL PROYECTO	1
1.1. Antecedentes	1
1.2. Descripción del proyecto.....	2
1.3. Aplicación	4
1.3.1 SISTEMA DE DETECCIÓN.....	4
1.4. Descripción del problema.....	6
1.5. Proyectos Similares	7
2. FUNDAMENTO TEÓRICO.....	10
2.1 Herramientas de Software	10
2.1.1 Compilador AVR STUDIO 4	10

2.1.2	PROTEUS (Versión 7.7)	12
2.2	Herramientas de Hardware	14
2.2.1	POLOLU 3PI	14
2.2.2	SHARP GP2Y0A21YK0F	16
CAPÍTULO 3		17
3	DESCRIPCIÓN E IMPLEMENTACIÓN DEL PROYECTO	17
3.1	Diseño Preliminar	17
3.1.1	Implementación Física	18
3.2	Descripción del proyecto final	20
3.3	Programa principal del microcontrolador	23
CAPÍTULO 4		30
4	SIMULACIÓN Y PRUEBAS	30
4.1	Simulación en Proteus	30
4.2	Imágenes del proyecto Final	32
CONCLUSIONES		39
RECOMENDACIONES		40
BIBLIOGRAFÍA		41
ANEXOS		43

ÍNDICE DE FIGURAS

FIGURA 1-1: Diagrama de bloques básico del proyecto	4
FIGURA 1-2: Sistema de detección de obstáculos.....	5
FIGURA 1-3: Sistema Anticolisiones	6
FIGURA 1-4: Robot Calimaro.....	8
FIGURA 1-5: Robot Navibot SR8855.....	9
FIGURA 1-6: Robot Moway.....	10
FIGURA 2-1: Entorno AVR STUDIO 4.....	11
FIGURA 2-2: Entorno PROTEUS V 7.7	13
FIGURA 2-3: Robot POLOLU 3PI	14
FIGURA 2-4: Sensor de Distancia SHARP.....	15
FIGURA 3-1: Estimacion de la tabla de convesión.....	19
FIGURA 3-2: Diagrama de bloque del proyecto.....	21
FIGURA 3-2: Diagrama de flujo del proyecto.....	21
FIGURA 4-1: Simulacion del proyecto.....	27
FIGURA 4-2: Simulacion de los sensores Sharp.....	28
FIGURA 4-3: Imágenes del proyecto funcionando.....	32
FIGURA 4-4: Presentación de los nombres de los integrantes del proyecto.....	33
FIGURA 4-5: Presentación del estado de las baterías.....	34
FIGURA 4-6: Presentación del MENU secundario.....	35
FIGURA 4-7: Presentación de los mensajes de la opción A.....	36
FIGURA 4-8: Presentación de los distancia de los sensores de la opción C.....	37

INTRODUCCIÓN

Se tiene como objetivo de este proyecto, el diseño e implementación de un sistema para evitar colisiones acoplado con el robot POLOLU 3PI; teniendo la capacidad de escoger la mejor ruta cuando se presente algún obstáculo en la trayectoria del Robot. El sistema es controlado por el PIC AVR Atmega328 propio del Robot Pololu, que se encarga de convertir las señales analógicas en digitales que son enviadas del sensor SHARP GP2Y0A21YK0F para la detección de obstáculos y su posterior elección de la mejor ruta para evitar la colisión.

El primer capítulo contiene una descripción general del proyecto, las partes que lo componen y las funciones que es capaz de realizar, además contiene las aplicaciones más comunes en las cuales es utilizado el proyecto y sistemas de similares características.

El segundo capítulo contiene una descripción general del software y el hardware utilizado para la elaboración del proyecto; entre las herramientas de software de programación, tenemos el compilador AVR STUDIO 4, con sus respectivas características, el simulador de circuitos Proteus con sus funciones principales. Entre las herramientas de Hardware se encuentran: el sensor SHARP GP2Y0A21YK0F y el pic AVR Atmega328.

El tercer capítulo contiene todo lo referente a la descripción e implementación del proyecto, comenzando con el prototipo inicial, las pruebas realizadas y los cambios realizados para tratar de obtener un mejor resultado, de acuerdo a las especificaciones planteadas.

El cuarto capítulo, se muestra los esquemas utilizados para el correcto funcionamiento del proyecto. Además se adjuntan los diagramas y esquemas electrónicos del hardware utilizado para la implementación del proyecto.

CAPÍTULO 1

1.-DESCRIPCIÓN GENERAL DEL PROYECTO

1.1. Antecedentes

Un sistema que evite obstáculos y además posteriormente gracias a esa detección se pueda evitar colisiones se ha convertido en la actualidad uno de los mecanismos con mayor atención en cuanto a su investigación y desarrollo.

El uso de la simulación de estos mecanismos en robots es una importante herramienta, para poder desarrollar y cada día mejorar los mencionados sistemas. El prototipo que implementaremos un sistema que evite obstáculos será basado en un robot denominado POLOLU.

Lo que se plantea en con este proyecto, es introducir un mecanismo de sistemas poco difundido como el de evitar colisiones con un costo menos excesivos, además que su accesibilidad sea más fácil, se pueda compartir y después presentar nuevas ideas para mejorar la eficiencia del proyecto.

1.2. Descripción del proyecto

La idea principal de este proyecto es usar sensores de medición de distancia para luego de estos el robot POLOLU mediante algoritmos para evitar colisiones pueda escoger la mejor ruta.

El funcionamiento del sistema se basa principalmente en las señales analógicas que los sensores de distancia SHARP nos proporcionen los cuales junto con las rutinas para evitar colisiones del Robot POLOLU determinaran la mejor ruta para que el robot continúe con su trayectoria.

En la figura que se muestra a continuación el diagrama de bloques que muestra el principio básico general del funcionamiento de nuestro proyecto:



FIGURA 1-1: Diagrama de bloques del proyecto

La ventaja del uso de los sensores SHARP es su bajo costo en comparación con los telémetros SONAR, pero nos proporciona una gran ventaja en cuantos a los sensores infrarrojos.

Para hacer más eficiente el uso de las rutinas para evitar colisiones del Robot POLOLU podemos tener varias alternativas para su programación basada en varios modelos conocidos como “EL CHOQUE DE LA ABEJA”, “EL COMPORTAMIENTO DE LOS PECES” o “EL SISTEMA NEUROLOGICO APLICADO” ,todos estos modelos son basados en características propias de la naturaleza para evitar las colisiones.

El sistema de detección de obstáculos va a tener dos salida y una entrada. De acuerdo a esta entrada nos permitirá corregir la posición y así evadir el obstáculo. La salida se convertirá en un ángulo de dirección que se va aplicar a la orientación del móvil; si el obstáculo está a la izquierda entonces gira a la derecha con un giro progresivo determinado y si el obstáculo está a la derecha lo contrario de acuerdo al patrón que se obtenga en los sensores.

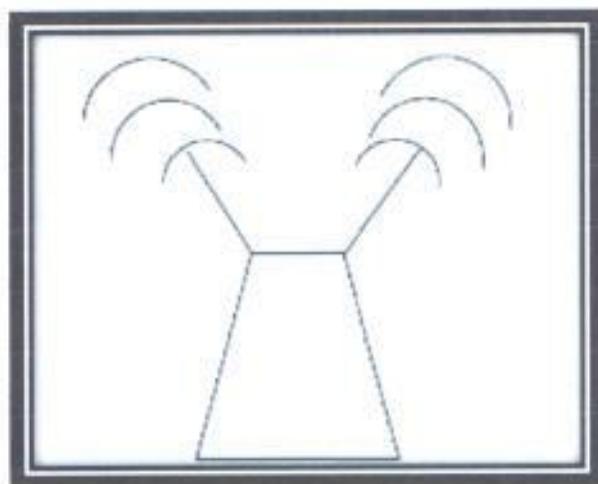


Fig. 1-2 Sistema de detección de obstáculo

1.3. Aplicación

El campo de aplicación de este proyecto está relacionado con la industria automovilística. Podremos desarrollar un prototipo inteligente para poder evitar obstáculos y colisiones, consiguiendo así reducir considerablemente la pérdida de daños materiales. Además podemos acoplar un sistema sofisticado de posicionamiento del móvil con el uso de GPS para tener un resultados más eficiente en cuanto al posicionamiento del móvil que nos permitira una mayor gama de opciones en cuanto al desvió de una trayectoria para evitar colisiones.

Podemos también construir un sistema de movilidad para personas discapacitadas, las cuales puede transportarse dentro de una clínica, residencia, o lugar donde se encontraran sin ninguna dificultad haciéndolos más independientes en cuanto a su movilización.

1.3.1 SISTEMA DE DETECCIÓN

Los sistemas de detección de obstáculos en la actualidad son uno de los parámetros más importantes para la construcción de vehículos aéreos, terrestres o marítimos.

En el sistema de detección, su tarea principal como lo indica es realizar una estimación de la distancia donde se encuentra algún objeto para de esta forma escoger la mejor ruta para poder evitar choques.

Para que un sistema de detección de obstáculos sea eficiente se debe tener una localización del vehículo precisa para con esto poder orientar al mismo en que ruta debe tomar cuando se encuentre con un objeto en su trayectoria.

Existen sensores visuales como las cámaras de video y algunos tipos de sensores no visuales como el sonar, el radar, los sensores inerciales y los de activación por presión, estos sensores se activan de acuerdo a la proximidad de un obstáculo, generando un patrón diferente de acuerdo al ángulo de choque.

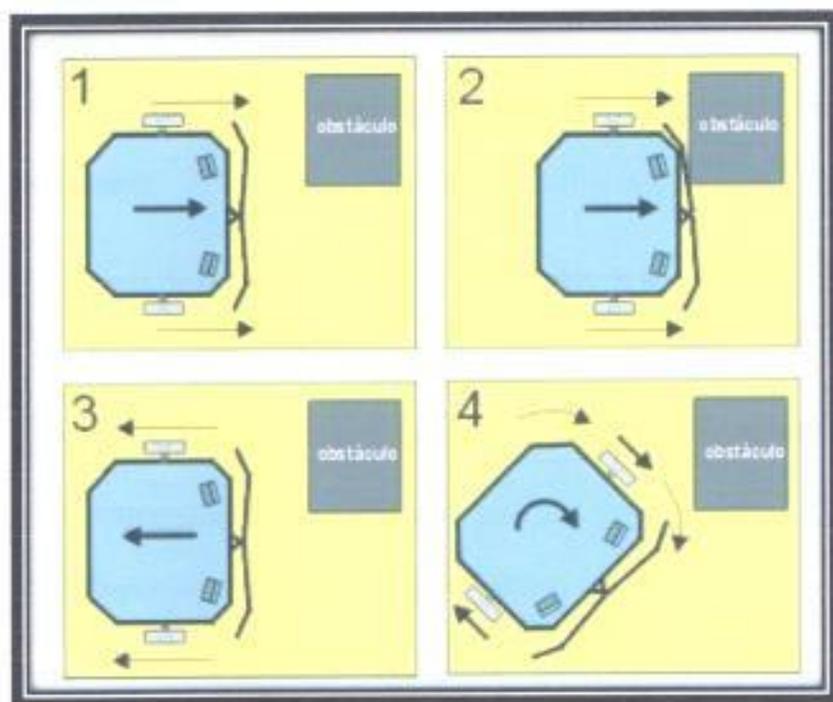


FIGURA 1-3: Sistema Anticolisiones

1.4. Descripción del problema

El uso de los sistemas de detección de obstáculos podrá ser de gran ayuda en muchos sectores como son los de seguridad, automotriz, médico, industrial, etc. De esta manera reduciendo considerablemente los accidentes por colisiones por fallas humanas.

Pero debemos tener en cuenta el entorno donde vamos aplicar nuestro sistema, ya que por lo general será muy complejo, debido a se encontrara funcionando en un entorno denominado "CARRETERAS NO ESTRUCTURADAS". Las principales características de una carretera no estructurada son:

- Ausencia de líneas que delimitan los bordes de la carretera.
- Bordes de carretera degradados.
- Posibles defectos, como grietas, en la superficie de la carretera.
- Problemas causados por sombras.
- No disponibilidad de un mapa del camino.

1.5. Proyectos Similares

1.5.1. Robot Calimaro



FIGURA 1-4: robot Calimaro

El robot Calimaro es un robot construido en su totalidad de transistores, motores, fotocélulas y componentes discretos. Un robot muy ingenioso, que logra comportarse como un sistema sofisticado. Sus movilidad es basada en los sistemas modernos con dos ruedas motrices independientes y una rueda libre, que solo sirve como apoyo complementando el tripote. Posee un seguidor de luz, detección de obstáculos que hacen un robot interesante de usar. Sus desventajas son la velocidad debido a su antiguo sistema de movilización. La detección de obstáculos es poco eficiente debido a que tiene que tocar el obstáculo para luego retroceder y escoger otra ruta.

Es un robot que se puede implementar con un costo de \$120

1.5.2. Navibot SR8855



FIGURA 1-5: Robot Navibot SR8855

Es un robot aspiradora fabricado por Samsung el cual posee 38 sensores para detectar obstáculos que evitan que el robot se choque con paredes o muebles además posee un sensor en la parte inferior que evita que se caiga por las escaleras. Posee una velocidad considerable y además se carga en tan solo 2 horas.

Se consigue en el mercado por un precio de \$ 580.

1.5.3. Moway



FIGURA 1-6: Robot Moway

Moway es una Robot educativo el cual tiene las facilidades de poder programarlo en lenguaje ensamblador, C, gráficamente. Etc. Se puede comunicar con varios Moways además de poseer una estructura mecánica y electrónica robusta.

Su desventaja son sus sensores anticollisiones, debido a que son sensores infrarrojos activos que necesitan la reflexión de la luz para poder funcionar.

El costo de un robot Moway es de \$ 240.

CAPÍTULO 2

2. FUNDAMENTO TEÓRICO

En el presente capítulo se muestran las herramientas de Software y Hardware utilizadas para el óptimo funcionamiento de nuestro proyecto. Además se mostrarán las características más relevantes de dichas herramientas Software y

Hardware como son:

El Compilador AVR STUDIO 4, el simulador Proteus versión 7.7, el robot Pololu 3PI, el sensor de distancia SHARP GP2Y0A21YK0F.

2.1 Herramientas de Software

2.1.1 Compilador AVR STUDIO 4



FIGURA 2-1: Entorno AVR STUDIO

El programa AVR STUDIO 4, desarrollado por ATMEL; es un compilador desarrollado para lenguaje ensamblador y lenguaje C para las series ATMEGA. AVR STUDIO es sin duda una de las mejores opciones para la realización de proyectos, ya que posee un entorno de trabajo más elaborado que los compiladores en Assembler y además tenemos la posibilidad de optimización en diferentes niveles. Una de sus ventajas es que podemos programar y depurar si se dispone de un J-TAG y adicionalmente simular el código.

2.1.1.1 Características del compilador AVR STUDIO 4:

- ❖ Se puede programar en distintos lenguajes como lenguaje ensamblador, C, C++ para microcontroladores.
- ❖ Al compilar genera un código de máquina muy compacto y eficiente que se integra perfectamente con algunos emuladores como PROTEUS para el proceso de depuración.
- ❖ La optimización de los códigos de programación usando la herramienta AVR-GCC logra una notoria disminución del tamaño del código.
- ❖ Incorpora drivers para dispositivos externos, tales como pantallas LCD, teclados numéricos, memorias EEPROM, conversores A/D, etc.
- ❖ Una de las características más interesantes es que trabaja bajo muchas plataformas, de entre ellas Windows, Linux, UNIX.
- ❖ Soporta todo tipo de microprocesadores de la ATMEL como los ATtiny, ATMEGA, etc.

2.1.2 PROTEUS (Versión 7.7)

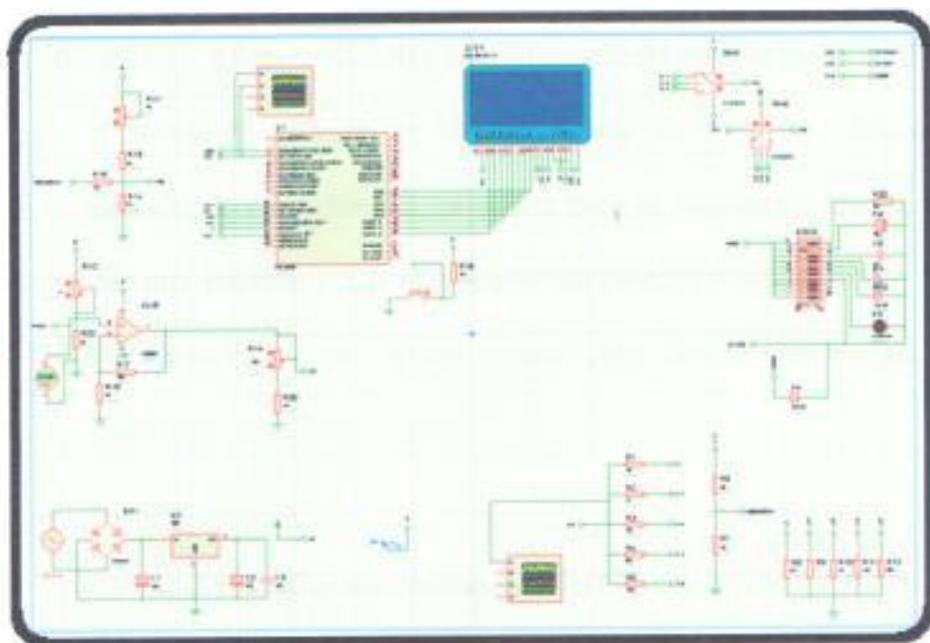


FIGURA 2-2: Entorno PROTEUS V 7.7

PROTEUS versión 7.7, desarrollado por Lab Center Electronics; es un entorno integrado diseñado para la realización completa de proyectos de construcción de equipos electrónicos en todas sus etapas: diseño, simulación, depuración y construcción. Este emulador consta de dos programas principales: Ares e Isis, y los módulos VSM y Electra.

- ❖ **ISIS.-** Intelligent Schematic Input System (Sistema de Enrutado de Esquemas Inteligente); es la herramienta que permite diseñar el esquema eléctrico del proyecto; incorporando una librería de más de 6.000 modelos de dispositivos digitales y analógicos como son:

resistencias, microcontroladores, leds, relés, microprocesadores, incluyendo fuentes de alimentación, generadores de señales, etc.

- ❖ ARES.- AdvancedRouting and Editing Software (Software de Edición y Ruteo Avanzado); es la herramienta de enrutado, ubicación y edición de componentes, se utiliza para la fabricación de placas de circuito impreso (PCB's); además nos permite visualizarlas en 3D (3 dimensiones) y esto ayuda mucho para obtener un acabado casi perfecto.

2.1.2.1 Características de PROTEUS V7.7:

- ❖ Entorno de diseño gráfico de esquemas electrónicos (ISIS) extremadamente fácil de utilizar y dotado de poderosas herramientas para facilitar el trabajo del diseñador.
- ❖ Entorno de diseño de placas de circuito impreso (ARES) con altas prestaciones con bases de datos de 32 bits, contiene un posicionador automático de elementos y generación automática de pistas con tecnologías de autocorte y regeneración.
- ❖ Moderno y atractivo interface de usuario estandarizado a lo largo de todas las herramientas que componen el entorno PROTEUS.
- ❖ Ejecutable en los diferentes entornos como Windows: 98, Me, 2000, Xp, Vista y W7.

2.2 Herramientas de Hardware

2.2.1 POLOLU 3PI

POLOLU 3pi es un robot seguidor de líneas básicamente. Es programable debido a que contiene un chip AVR ATmega 328/ATmega 168. Además de poseer un sistema de tracción de motores que hace desarrollar velocidades de 100cm/s, puede realizar vueltas precisas y cambios de sentido.

El POLOLU 3pi es una excelente plataforma para la introducción a los denominados robots programables. Teniendo una alta gama de accesorios que nos permitirán ampliar un buen número de aplicaciones y además presentando la opción de nosotros poder construir y acoplar nuestros propios diseños electrónicos.



FIGURA 2-3: POLOLU 3PI

2.2.1.1. Características de Pololu 3PI

El robot POLOLU 3PI posee las siguientes características:

- Posee compatibilidad con varios programadores para poder como el AVR STUDIO o el ARDUINO.
- Se puede programar su pantalla LCD para mostrar mensajes a gusto del usuario.
- Posee un buzzer para emitir sonidos.
- Posee 21 I/O disponibles para el usuario.
- Múltiples accesorios para acoplar y formar proyectos más sofisticados.
- Diseñado para resolver laberintos con gran precisión escogiendo la mejor ruta.
- Pequeño, compacto y ligero.
- Posee 3 botoneras lo cual nos permitiría crear menús y tener la opción de poder escoger diferentes opciones de aplicaciones haciendo una sola programación.
- Los sensores infrarrojos acoplados en el robot pololu permiten implementar varias aplicaciones sin tener que acoplar al robot más sensores de forma innecesaria.
- Al poseer el ATMEGAXX8 nos facilita en cuanto al uso de las librerías gratuitas para el uso del pololu, y el uso de más accesorios.

2.2.2 SHARP GP2Y0A21YK0F

El sensor de distancia SHARP GP2Y0A21YK0F nos permite detectar obstáculos y evitarlos. Este sensor puede detectar objetos a una distancia de 10 cm-80 cm dándonos como salida una señal analógica.

2.2.2.1. Características:

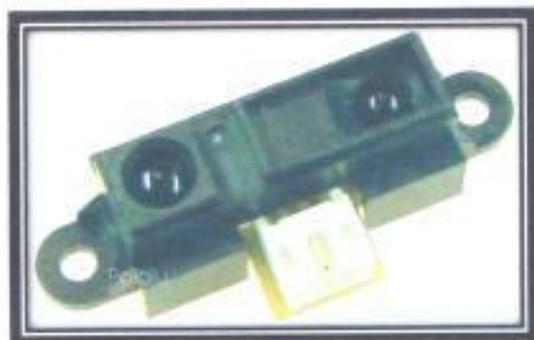


FIGURA 2-4: SENSOR DE DISTANCIA SHARP

- ❖ Tiempo de respuesta rápida 38 ms
- ❖ Rango de medición de distancias entre 10 cm – 80 cm
- ❖ Ligero
- ❖ Proporciona mayor seguridad de predicción de distancia que un sistema de rayos infrarrojos
- ❖ Rango de Voltaje de Operación: 4,5 a 5,5 Voltios.
- ❖ Consumo medio de corriente: 30 mA
- ❖ Linearización de la salida.

CAPÍTULO 3

3 DESCRIPCIÓN E IMPLEMENTACIÓN DEL PROYECTO

3.1 Diseño Preliminar

La implementación del proyecto fue dividida en siete partes:

- 1.- Implementar las rutinas para que el Robot pueda girar a la derecha o la izquierda dependiendo de un parámetro de control.
- 2.- Implementar una rutina para que el robot regrese a la trayectoria original después de haber girado a la derecha o izquierda respectivamente.
- 3.- Diseñar una tabla de conversión analógico-digital para convertir las señales analógicas que nos proporciona el sensor de Distancia Sharp.
- 4.- Diseñar el sistema de detección de obstáculos denominado "SISTEMA NEUROLOGICO APLICADO", basado en 2 sensores de distancia para que el robot POLOLU pueda evitar las colisiones.
- 5.- Probamos el sistema de detección de obstáculos programado en el robot POLOLU para mejorar las rutinas de evitar colisiones.

6- Programamos la pantalla LCD del robot POLOLU para presentar los nombres de los autores del proyecto e indique los giros que realiza el robot POLOLU.

7.- Programamos la pantalla LCD del robot POLOLU para presentar el estado de la trayectoria del Robot y además presentar la distancia en centímetros que detectan los sensores.

8.- Creamos unas rutinas para producir los respectivos sonidos que se emitirán por medio del buzzer cuando evite una colisión, o cuando este en el modo búsqueda.

3.1.1 Implementación Física

Para realizar la implementación física del proyecto, comenzamos armando en el protoboar del sistema con un sensor de detección de obstáculos.

- ❖ Los Sensores estarán ubicados en los lados derechos e izquierdos del Robot POLOLU.
- ❖ La ubicación de estos sensores es para tener una correcta detección de obstáculos y escoger el mejor desvío de la trayectoria del robot.

El valor del parámetro *Voltaje* se mide de forma indirecta a través de una tabla de conversión analógico-digital previamente cargada en el PIC ATmega328 para la precisa medición de distancia.

Los valores del parámetro de voltaje pueden variar desde 0.3 v-3.5, dependiendo de la distancia que permite determinar distancias comprendidas entre 10cm-80cm estos

valores pasan a las entradas disponibles en el PIC ATmega328 que son los pines ADC5 – ADC6 del PIC los cuales se encarga de transformar estos valores en digitales, luego con la programación del PIC nos encargamos de procesar los datos para que las distancias medidas sean las correctas.

La conversión de los valores de las señales a analógicas que es proporcionada por los dos sensores lo hace en los rangos de 40 a 330 que representan respectivamente los 10cm y los 80 cm que es los sensores son capaces de detectar.

CONVERSION ANALAGICO DIGITAL		
DISTANCIA	VOLTAJE DEL SENSOR	CONVERSION
10	2.4 [V]	330
20	1.3 [V]	300
30	0.9 [V]	260
40	0.7 [V]	215
50	0.6 [V]	189
60	0.5 [V]	148
70	0.4 [V]	95
80	0.3 [V]	40

Fig. 3-1 Estimación de la tabla de conversión

3.2 Descripción del proyecto final

Al finalizar con la programación y la implementación del prototipo, podemos hacer una descripción del funcionamiento del proyecto:

- 1.- Se muestra al encender el ROBOT POLOLU 3PI en la LCD un mensaje con los nombres de los autores del proyecto y presentando el menú principal para que el sistema de detección de errores empiece a funcionar presionando la tecla B.
- 2.- Si no se presiona la tecla B, en la pantalla LCD se presenta el valor del estado de las baterías.
- 3.- El Buzzer emite un sonido para indicar que se presione la tecla B al final de la presentación de los nombres.
- 4.- Una vez presionada la tecla B, el Robot POLOLU 3PI empieza su trayectoria en y los sensores Sharp empiezan a funcionar para poder detectar los obstáculos.
- 5.- Los motores del robot empiezan a moverse desde el reposo progresivamente hasta alcanzar una velocidad constante.
- 6.- Debemos esperar unos 3 segundos para que el sistema de detección de obstáculos empiece a funcionar.
- 7.- El sistema de detección de obstáculos indicara los giros que se deban hacer de acuerdo a las señales detectadas en el sensor.

8.- El robot después de girar a la derecha o izquierda intentará regresar a su trayectoria original girando la cantidad exacta de los grados que dio al giro inicial siempre que sea posible.

9.- El buzzer emite un sonido cada vez q se detecta un obstáculo y otro diferente cuando lo esquiva.

10.- Cuando el robot se encuentra "atrapado" genera un tono diferente para indicar que se encuentra en ese estado.

11.- Adicionalmente podemos presionar las teclas A o C que podrán presentarnos en la LCD información que nosotros deseamos.

12.- Si presionamos la tecla A en la LCD se presentará el estado del robot LCD en su trayectoria es decir, nos indicaran cuando en la trayectoria se presente un obstáculo o presente obstáculos en la izquierda o derecha de la trayectoria del robot o si no presenta ningún obstáculo.

13.- Si presionamos la tecla C en la LCD se presentará en la LCD la distancia en centímetros que los sensores SHARP detectan cuando se presenta un obstáculo en la trayectoria del ROBOT POLOLU 3PI.

3.2.1 Diagrama de bloques del proyecto

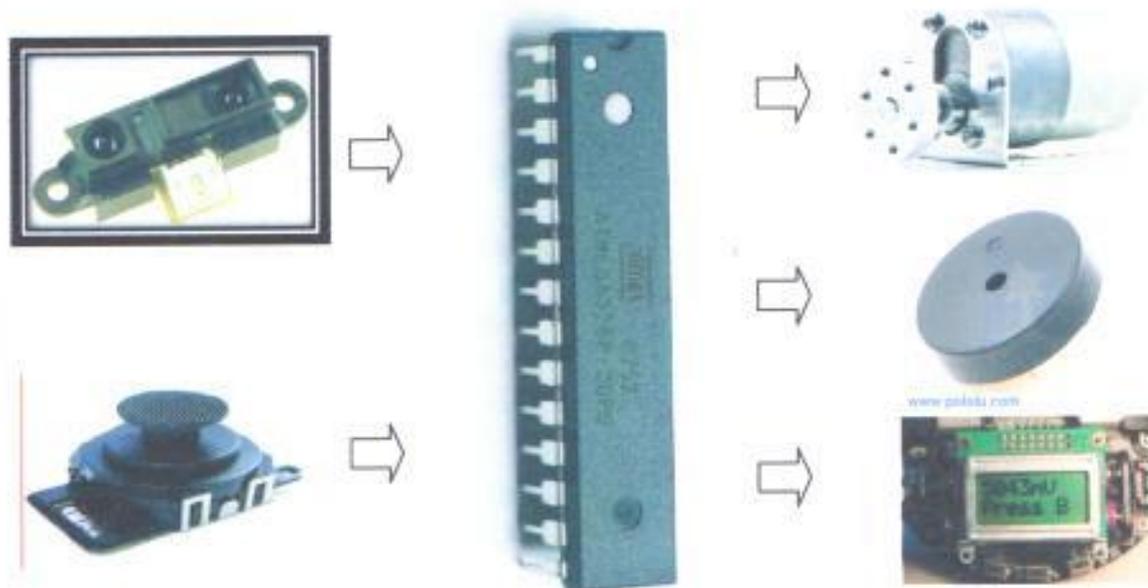


Fig. 3-2 Diagrama de bloque

3.2.2 Diagrama de flujo del proyecto

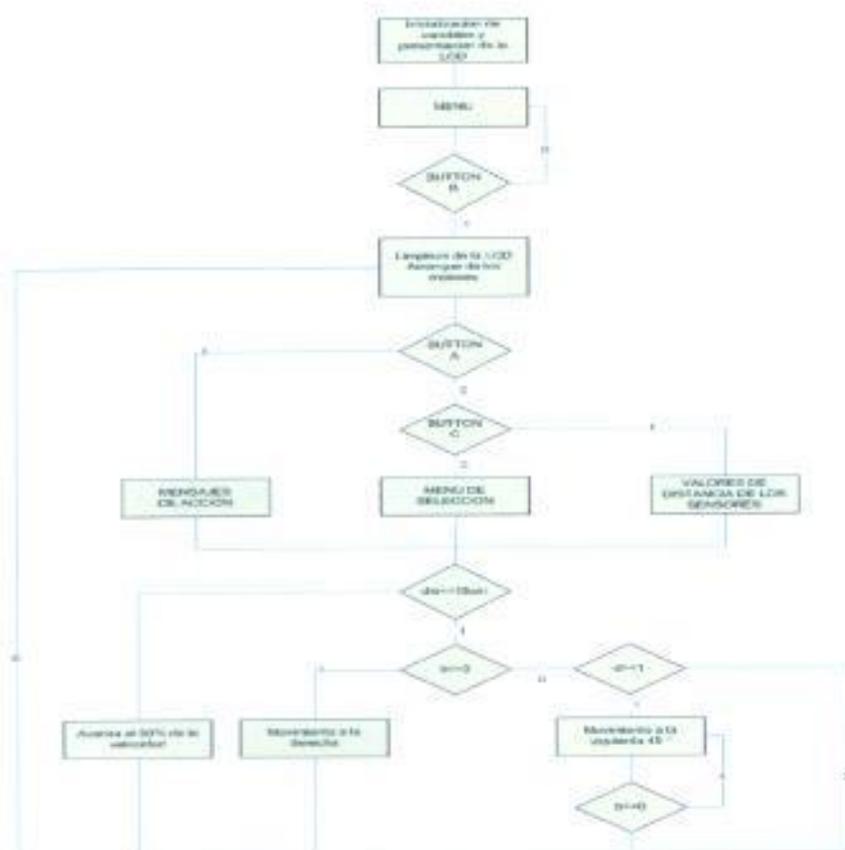


Fig. 3-3 Diagrama de flujo del proyecto

3.3 Programa principal del microcontrolador

```
/*
NOMBRE DEL PROYECTO:
ROBOT ESQUIVADOR V1
INTEGRANTES:
. RONALD SOLIS
. ANGEL NAULA
-----DESCRIPCION-----
PROGRAMA QUE CONVIERTE LAS SEÑALES ANALOGAS DEL SENSOR
SHARP EN DIGITALES REPRESENTADAS EN UN RANGO DE 40-330 QUE
EQUIVALEN A 10CM-80CM. PARA SU POSTERIOR UTILIZACION EN LA
IMPLEMENTACION DE LAS RUTINAS DE EVITAR COLISIONES
*/

#include<pololu/3pi.h> //LIBRERIA PARA EL USO DE LAS FUNCIONES DEL POLOLU
int main() //INICIO DE PROGRAMA

{
int a,a2,ax,ax2,an,an2,b,bandc,banda,c,d,tem; // DECLARACION DE VARIABLES

a=0;
a2=0;
b=0;
c=0;
d=0;
tem=0;
bandc=0;
banda=0;
an=0;
an2=0;
ax=0;
ax2=0;

/* ----- IMPRIME EN LA LCD LOS NOMBRES DE LOS MIEMBROS -----
----- DEL PROYECTO Y EL MENU PRINCIPAL DEL POLOLU -----*/
print("Ronald");
lcd_goto_xy(0,1);
print("Solis");
delay_ms(2000);
clear();
print("Angel");
lcd_goto_xy(0,1);
print("Naula");
delay_ms(2000);
clear();
```

```
// TONO DE INICIO AL TERMINAR LA PRESENTACION
```

```
play("L16 ceg>c");
```

```
/*--- EVALUA SI SE PRESIONO LA TECLA B PARA QUE EL -----  
----- SISTEMA DE DETECCION DE OBSTACULOS EMPIEZE -----*/
```

```
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);  
}
```

```
clear();
```

```
/*--- FUNCION DE LA DETECCION DE OBSTACULOS DONDE-----  
CONVIERTEN LAS SEÑALES ANALOGICAS DEL SENSOR SHARP PARA---  
ELEGIR LA MEJOR TRAYECTORIA DEL ROBOT-----*/
```

```
while(1)
```

```
{  
  a= analog_read(6); //LEO EL ADC6  
  a2= analog_read(5); //LEO EL ADC5  
  
  if(a>=330) //SI a ES MAYOR DE 330 ES 80CM  
  
  {  
    ax=373;  
  }  
  
  if(a>40 && a<330) //ASIGNO EL VALOR DIRECTO AL ADC  
  
  {  
    ax=a;  
  }  
  
  if(a<=40) //ASIGNA UN VALOR MENOS A 80 CM  
  
  {  
    ax=32;  
  }  
}
```

```
/*----- REPETIMOS LAS MISMAS FUNCIONES PARA EL-----  
----- SEGUNDO SENSOR UBICADO EN LA DERECHA-----*/
```

```
if(a2>=330)
```

```
{  
    ax2=373;  
}
```

```
if(a2>40 && a2<330)
```

```
{  
    ax2=a2;  
}
```

```
if(a2<=40)
```

```
{  
    ax2=32;  
}
```

```
//ASIGNO an y an2 LOS VALORES CONVERTIDOS ENTRE 10 Y 80 cm
```

```
an=((330 - ax)*8)/33)+10;  
an2=((330 - ax2)*8)/33)+10;
```

```
/*----- SEGUNDO MENU SI PRESIONAMOS LAS TECLAS A O C-----*/
```

```
if(button_is_pressed(BUTTON_C))
```

```
{  
    bandc=1;  
    banda=0;  
}
```

```
if(button_is_pressed(BUTTON_A))
```

```
{  
    bandc=0;  
    banda=1;  
}
```

```
/*----- SI NO SE PRESIONA NADA SE DEBE PRESENTAR -----  
----- EL SEGUNDO MENU EN LA LCD DEL POLOLU -----*/
```

```
if(bandc==0 &&banda==0 )
```

```
{  
    print("A. MSN");  
    lcd_goto_xy(0,1);  
    print("C.RANGE");  
    delay_ms(100);  
    clear();  
}
```

```
/*-----FUNCION QUE PRESENTA EN LA LCD LAS DISTANCIAS-----  
-----DETECTADAS POR LOS SENSORES EN CM-----*/
```

```
if(bandc==1)
```

```
{  
    print_long(an);  
    lcd_goto_xy(0,1);  
    print_long(an2);  
    delay_ms(80);  
    clear();  
    print(" cm SR");  
    lcd_goto_xy(0,1);  
    print(" cm SL");  
    delay_ms(80);  
    clear();  
}
```

```
/* ----- FUNCION PARA EVADIR OBSTACULOS-----*/
```

```
if(a>=320 && b<=3)
```

```
{  
    if(banda==1)  
    {  
        print(" WALL");  
        lcd_goto_xy(0,1);  
        print(" Front ");  
        delay_ms(100);  
        clear();  
    }  
}
```

```
play("o5110cre"); //ACTIVO UN SONIDO PARA LA ETAPA DE BUSQUEDA  
set_motors(35,-35); //SETEO LOS MOTORES PARA QUE GIREN  
b=b+1;  
d=0;
```

```
}  
/*-----FUNCION PARA QUE CONTINUE SU CAMINO-----*/
```

```
if(a<320 && a2<320)
```

```
{  
    if(banda==1){  
        print(" FREE");  
        lcd_goto_xy(0,1);  
        print(" GO ");  
        delay_ms(100);  
        clear();  
    }  
}
```

```
    d=1;  
    set_motors(50,50); //ASIGNO A LOS MOTORES CON VALORES IGUALES  
    play("o3150crc");  
}
```

```
if(a2>=320 && b<=3)// DEBE CONTINUAR SU TRAYECTORIA
```

```
{  
    if(banda==1){  
        print(" BIT");  
        lcd_goto_xy(0,1);  
        print("Space");  
        delay_ms(100);  
        clear();  
    }  
}
```

```
set_motors(35,-35); //GIRO A LA DERECHA
```

```
play("o5110crc");  
b=b+1;  
d=0;  
}
```

```
/*----- GENERA UN SONIDO CUANDO EL ROBOT NO TIENE CAMINO -----  
----- LIBRE PARA CONTINUAR Y GENERO UN SONIDO -----*/
```

```
if(a>=320 && a2>=320 && banda==1)
```

```
{  
    print("ROAD");  
    lcd_goto_xy(0,1);  
    print("CLOSED");  
    delay_ms(100);  
}
```

```
clear();
```

```
play("o5110ldl");
```

```
}
```

```
/*FUE BLOQUEADO A LA DERECHA
```

```
if(a2>=320 && a<=320 && banda==1 && b>=4)
```

```
{
```

```
print("Block");
```

```
lcd_goto_xy(0,1);
```

```
print("Right ");
```

```
delay_ms(100);
```

```
play("o5110cre");
```

```
clear();
```

```
}
```

```
/*FUE BLOQUEADO A LADO IZQUIERDO
```

```
if(a>=320 && a2<=320 && banda==1 && b>=4)
```

```
{
```

```
print("Block");
```

```
lcd_goto_xy(0,1);
```

```
print("LEFT");
```

```
delay_ms(100);
```

```
play("o5110cre");
```

```
clear();
```

```
}
```

```
/*--PARA QUE VAYA A LA RUTINA DE RETORNO DE POSICION--*/
```

```
if(b>=4 && d==1)
```

```
{
```

```
c=1;
```

```
}
```

```
tem=b;
```

```
/*-----HAGO REPETITIVO EL RETORNO HASTA REGRESERAR ---
```

```
-----A LA POSICION INICIAL-----*/
```

```
while(c==1)
```

```
{
```

```
if(banda==1)
```

```
{
```

```
print(" Return");
```

```
    lcd_goto_xy(0,1);  
    print("Posicion");  
    delay_ms(100);  
    clear();
```

```
    }
```

```
play("o3150cre");
```

```
if(b==tem)
```

```
{
```

```
    delay_ms(200);
```

```
}
```

```
delay_ms(115);
```

```
tem=1000;
```

```
set_motors(-35,35); //GIRO EN SENTIDO CONTRARIO AL GIRO DE ESQUIVADO
```

```
b=b-1;
```

```
if(b<=0)//CUANDO ACURRA ESTO REINICIALIZO LOS VALORES
```

```
{
```

```
    c=0;
```

```
    d=0;
```

```
    b=0;
```

```
}
```

```
}
```

```
}
```

```
return 0;
```

CAPÍTULO 4

4. SIMULACIÓN Y PRUEBAS

4.1 Simulación en Proteus

Gracias a la ayuda de la herramienta de simulación Proteus, se logró realizar la mayor parte de este proyecto, en ciertos casos fue necesario recurrir a la utilización de elementos ya existentes,

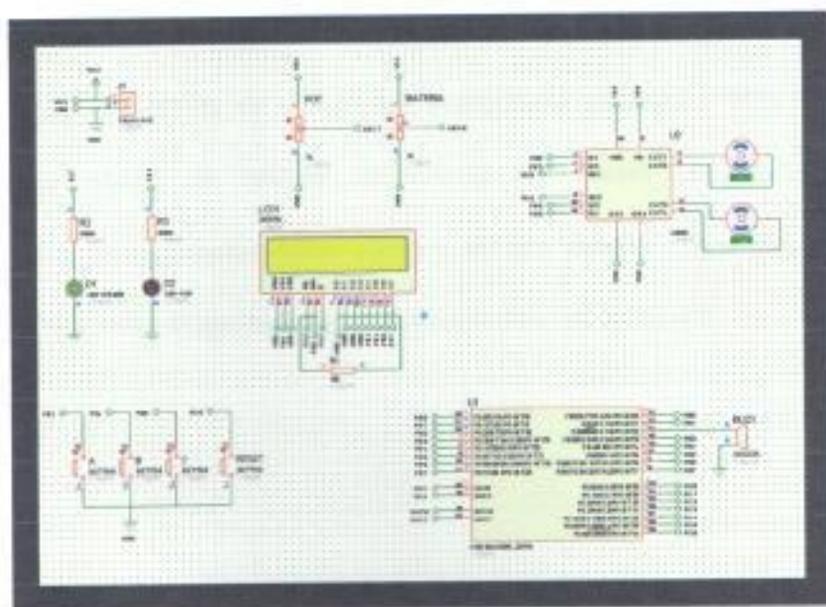


FIG. 4-1 SIMULACIÓN DEL PROYECTO

La simulación de nuestros sensores Sharp lo hemos implementado con el uso de unos potenciómetros conectados a una fuente y variamos el potenciómetro para simular la señal que emite al detectar un obstáculo a una distancia que emiten los sensores Sharp.

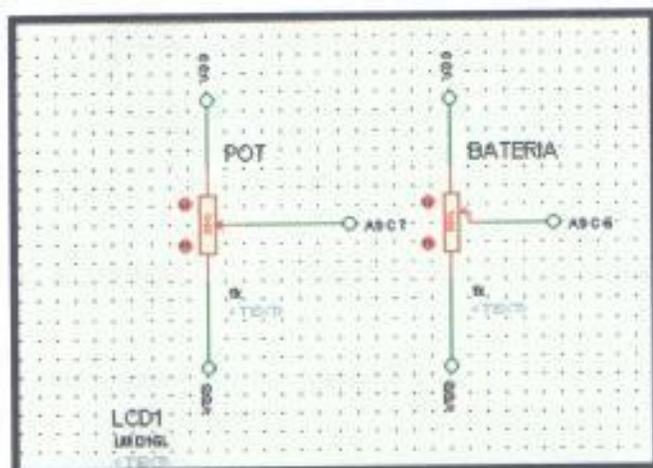


FIG. 4-2 SIMULACIÓN DE LOS SENSORES SHARP

El problema de la simulación de los sensores Sharp es por su curva característica la cual nos representa, en distancias comprendidas entre 0cm y 8cm nos presenta un rango de valores de voltaje los cuales se pueden ser mal interpretados en distancias entre 10cm y 80 cm debido a que son valores de voltaje de similares magnitudes para la respectivas distancias

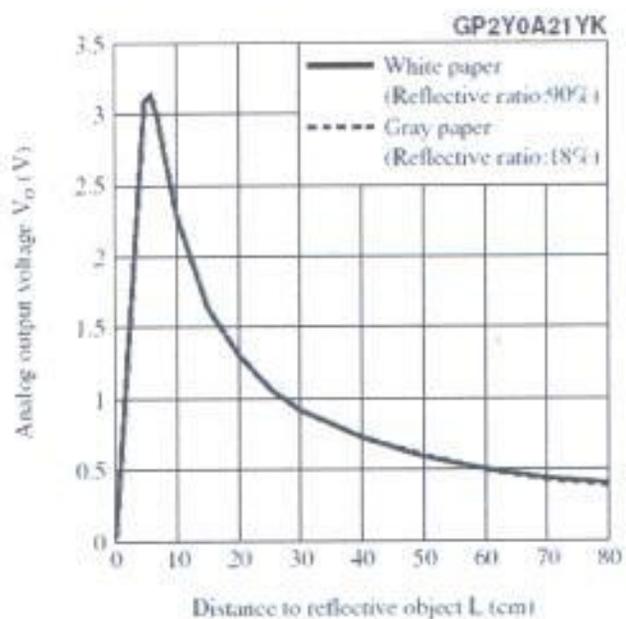


FIG. 4-2 CURVA CARACTERÍSTIA DEL SENSOR SHARP

4.2 Imágenes del proyecto Final

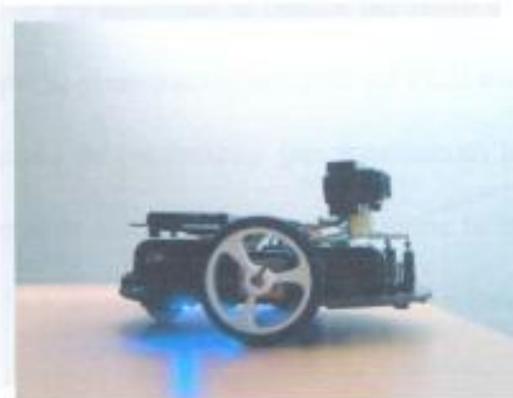
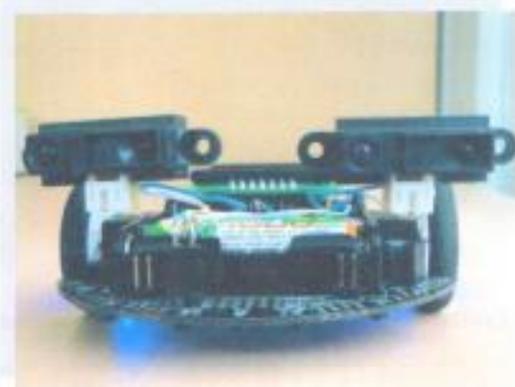


FIGURA 4-3: Imagen del proyecto funcionando

Cuando se prende el robot la LCD está configurada para presentar el nombre de los integrantes del proyecto. El formato de la presentación de los nombres es el primer nombre, seguido del apellido paterno.



FIGURA 4-4: Presentación de los nombres de los integrantes del proyecto

Después de presentar los nombres, el buzzer del robot emitirá un sonido el cual nos indicará que la presentación termino. La LCD está programada para que presente el estado de las baterías que suministran la energía al robot, adicionalmente presenta el mensaje que se presione la tecla B para que el sistema de detección y de evadir obstáculos empiece a funcionar.



FIGURA 4-5: Presentación del estado de las baterías

Después de presionar la tecla B el sistema de detección de obstáculos formado por los dos sensores SHARP, ubicados a la derecha e izquierda del robot empiezan a funcionar.

Los motores del robot POLOLU, comienzan a arrancar desde el reposo hasta alcanzar una velocidad constante.

La LCD también presentara un segundo menú donde nos pedirá presionar las teclas A y C. Adicionalmente el buzzer emite diferentes sonidos cuando los sensores empiezan a detectar obstáculos y los evita.

Los sonidos generados por el Buzzer son debidos la función 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

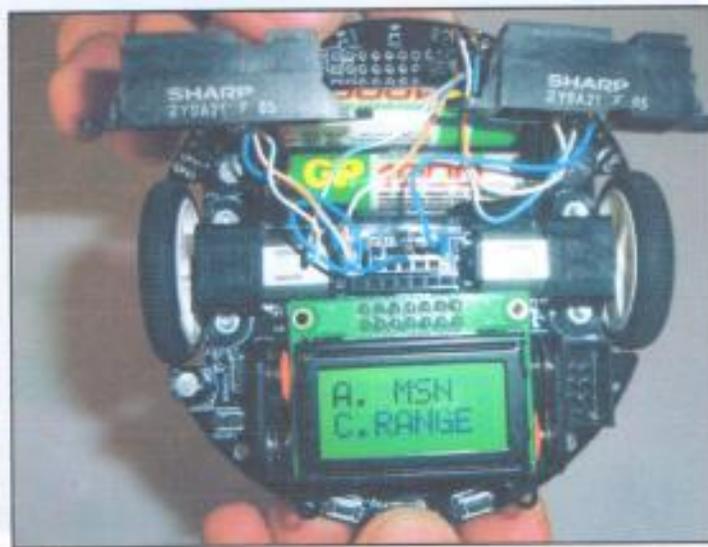


FIGURA 4-6: Presentación del MENU secundario

Una vez escogida la opción presentado en la LCD, el programa salta a la respectiva rutina seleccionada.

Cuando se presiona la Tecla A aparecen los diferentes mensajes mostrando el estado de la trayectoria del robot. Estos mensajes aparecen haciendo referencia a los objetos que pueda detectar los sensores analógicos SHARP.

Los diferentes mensajes que se presentan son:

- 1.- LEFT BLOCK: este mensaje nos indica cuando el lado izquierdo del robot tiene un obstáculo.
- 2.- RIGHT BLOCK: este mensaje nos indica cuando el lado derecho del robot tiene un obstáculo.
- 3.- ROAD CLOSE: este mensaje nos indica cuando el robot se encuentra encerrado, en este estado se activa el modo búsqueda.

4.- FREE GO: nos indica que en la trayectoria del robot no se presenta ningún obstáculo.

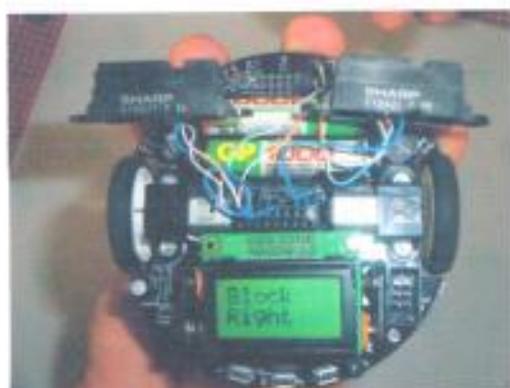




FIGURA 4-7: Presentación de los mensajes de la opción A

Quando los sensores detectan un obstáculo y el robot evita las colisiones se generan diferentes sonidos producidos por el Buzzer, para indicar las acciones que se encuentra realizando el robot en las respectivas condiciones.

Quando se presiona la Tecla C, en la LCD se presenta la distancia en centímetros que los sensores detectan a los obstáculos.





FIGURA 4-8: Presentación de los distancia de los sensores de la opción C

En la Figura se puede observar la presentación de la distancia en centímetros en el display del robot Pololu, el cual mediante el sensor analógico de distancia Sharp se realiza una conversión, en el cual mediante la curva característica del sensor se realiza la tabla de conversión de las distancia que se puede presentar un obstáculo en la trayectoria del robot Pololu.

CONCLUSIONES

1. Logramos diseñar e implementar un sistema que permite detectar los obstáculos que se encuentran en la trayectoria del robot POLOLU y así poder evitarlos.
2. El proyecto realizado es un prototipo de carácter demostrativo, requiere tiempo, dinero e investigación, para mejorar su diseño y de esta forma obtener un producto de tipo comercial.
3. El diseño y el prototipo forman una base didáctica, que sirve para futuros proyectos, similares en los cuales se puede mejorar la forma de detectar obstáculos y evitarlos usando nuevas tecnologías.

RECOMENDACIONES

1. El uso de los sensores SHARP debemos tenerlos un especial cuidado debido a su curva característica lo cual presenta un problema en el sensado de 0cm a 10 cm .
2. Debemos tener en cuenta la velocidad a la que se el robot puede moverse debido a que el sistema de la detección de obstáculos es un poco lento. Y al principio se debe incrementar la velocidad de formar progresiva hasta llegar a una velocidad constante para que los sensores puedan funcionar correctamente.
3. Es recomendable usar un sistema eficiente para poder energizar el robot, debido a que estamos consumiendo energía con el uso de los sensores. Y cuando las baterías están bajas el sistema tiende a disminuir su eficiencia.
4. Debemos tener en cuenta el entorno donde vamos aplicar nuestro sistema, ya que por lo general será muy complejo, debido a se encontrara funcionando en un entorno denominado CARRETERAS NO ESTRUCTURADAS.

ANEXOS

Features

- High Performance, Low Power AVR[®] 8-Bit Microcontroller
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 20 MIPS Throughput at 20 MHz
 - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
 - 4K/16/32K Bytes of In-System Self-Programmable Flash program memory
 - 256/512/512/1 K Bytes EEPROM
 - 512/1K/1K/2K Bytes Internal SRAM
 - Write/Erase Cycle: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/100 years at 25°C(1)
 - Optional Boot Code Section with Independent Lock Bits
 - o In-System Programming by On-chip Boot Program
 - o True Read-While-Write Operation
 - o Programming Lock for Software Security
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Six PWM Channels
 - 8-channel 10-bit ADC in TQFP and QFNMLF package
 - o Temperature Measurement
 - 6-channel 10-bit ADC in PDIP Package
 - o Temperature Measurement
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Byte-oriented 3-wire Serial Interface (Philips I²C compatible)
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
 - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
 - 23 Programmable I/O Lines
 - 28-pin PDIP, 32-lead TQFP, 28-pad QFNMLF and 32-pad QFNMLF
- Operating Voltage:
 - 1.8 - 5.5V
- Temperature Range:
 - -40°C to 85°C
- Speed Grade:
 - 0 - 4 MHz @ 1.8 - 5.5V, 0 - 10 MHz @ 2.7 - 5.5V, 0 - 20 MHz @ 4.5 - 5.5V
- Power Consumption at 1 MHz, 1.8V, 25°C
 - Active Mode: 0.2 mA
 - Power-down Mode: 0.1 µA
 - Power-save Mode: 0.75 µA (including 32 kHz RTC)



8-bit AVR[®]
Microcontroller
with 4/8/16/32K
Bytes In-System
Programmable
Flash

ATmega48A
ATmega48PA
ATmega88A
ATmega88PA
ATmega168A
ATmega168PA
ATmega328
ATmega328P

Summary

Rev. 027 (CS-AVR-09)0



1. Pin Configurations

Figure 1-1. Pinout ATmega48A/48PA/88A/88PA/168A/168PA/328/328P

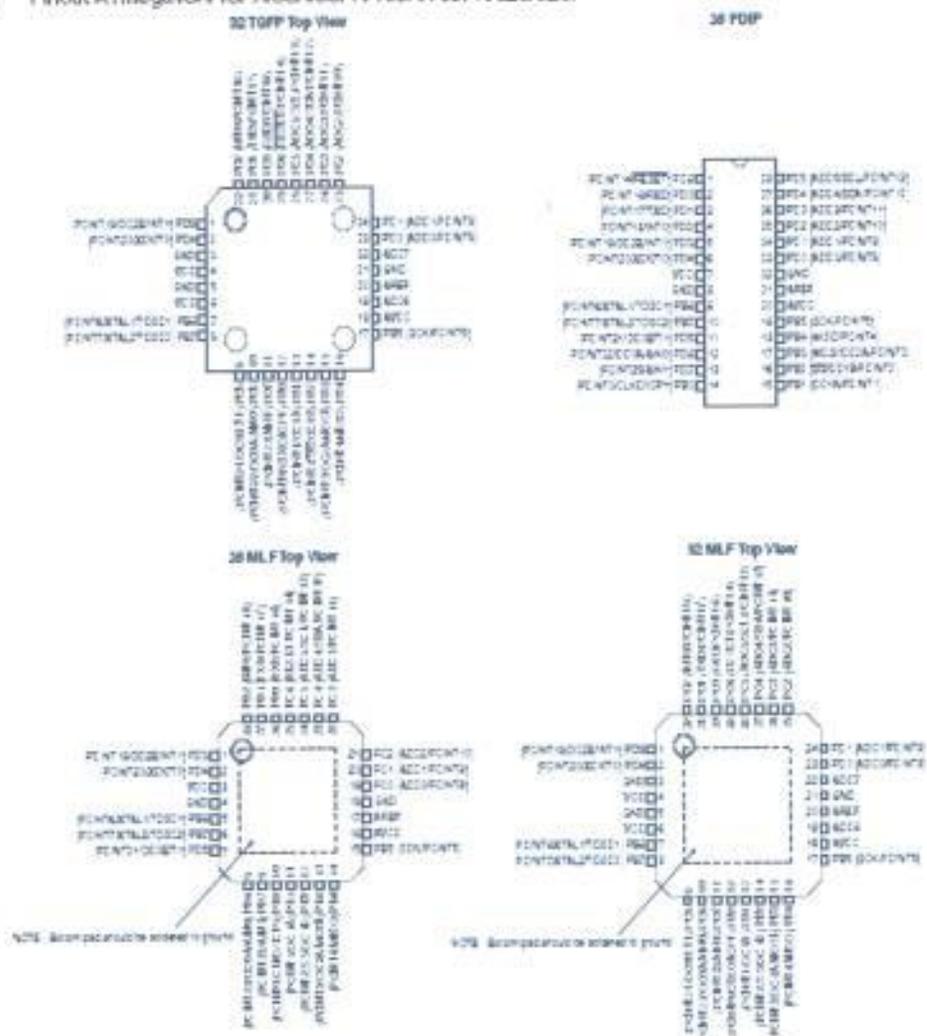


Table 1-1. 32UFPGA - Pinout ATmega48A/48PA/88A/88PA/168A/168PA

	1	2	3	4	5	6
A	PD2	PD1	PC6	PC4	PC2	PC1
B	PD3	PD4	PD0	PC5	PC3	PC0
C	GND	GND			ADC7	GND
D	VDD	VDD			AREF	ADC6
E	PB6	PD6	PB0	PB2	AVDD	PB5
F	PB7	PD5	PD7	PB1	PB3	PB4

1.1 Pin Descriptions

1.1.1 VCC

Digital supply voltage.

1.1.2 GND

Ground.

1.1.3 Port B (PE7:0) XTAL1/XTAL2/TOSC1/TOSC2

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB7...6 is used as TOSC2...1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

The various special features of Port B are elaborated in ["Alternate Functions of Port B"](#) and ["System Clock and Clock Options"](#) on page 26.

1.1.4 Port C (PC5:0)

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC5...0 output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

1.1.5 PC6/RESET

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.

If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. The minimum pulse length is given in [Table 28-12 on page 323](#). Shorter pulses are not guaranteed to generate a Reset.

The various special features of Port C are elaborated in ["Alternate Functions of Port C"](#) on page 26.

1.1.6 Port D (PD7:0)

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

The various special features of Port D are elaborated in "Alternate Functions of Port D" on page 89.

1.1.7 AV_{CC}

AV_{CC} is the supply voltage pin for the A/D Converter, PC3:0, and ADC7:6. It should be externally connected to V_{CC}, even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter. Note that PC6...4 use digital supply voltage, V_{CC}.

1.1.8 AREF

AREF is the analog reference pin for the A/D Converter.

1.1.9 ADC7:6 (TQFP and QFN/MLF Package Only)

In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

5. Instruction Set Summary

Mnemonic	Operands	Description	Operation	Flags	#Cycles
ARITHMETIC AND LOGIC INSTRUCTIONS					
ADD	Rd, Rr	Add Two Registers	$Rd \leftarrow Rd + Rr$	Z, C, NV, H	1
ADC	Rd, Rr	Add with Carry Two Registers	$Rd \leftarrow Rd + Rr + C$	Z, C, NV, H	1
ADDF	Rd, k	Add Immediate to Word	$Rd \leftarrow Rd + Rd + k$	Z, C, NV, S	2
SUB	Rd, Rr	Subtract Two Registers	$Rd \leftarrow Rd - Rr$	Z, C, NV, H	1
SUBI	Rd, k	Subtract Constant from Register	$Rd \leftarrow Rd - k$	Z, C, NV, H	1
SBC	Rd, Rr	Subtract with Carry Two Registers	$Rd \leftarrow Rd - Rr - C$	Z, C, NV, H	1
SBCI	Rd, k	Subtract with Carry Constant from Reg	$Rd \leftarrow Rd - k - C$	Z, C, NV, H	1
SRAR	Rd, k	Subtract Immediate from Word	$Rd \leftarrow Rd - Rd + k$	Z, C, NV, S	2
AND	Rd, Rr	Logical AND Registers	$Rd \leftarrow Rd \& Rr$	Z, NV	1
ANDI	Rd, k	Logical AND Register and Constant	$Rd \leftarrow Rd \& k$	Z, NV	1
OR	Rd, Rr	Logical OR Registers	$Rd \leftarrow Rd \vee Rr$	Z, NV	1
ORI	Rd, k	Logical OR Register and Constant	$Rd \leftarrow Rd \vee k$	Z, NV	1
EOR	Rd, Rr	Exclusive OR Registers	$Rd \leftarrow Rd \oplus Rr$	Z, NV	1
LSR	Rs	Logical Shift Right	$Rs \leftarrow Rs \gg 1$	Z, C, NV	1
NEG	Rd	Two's Complement	$Rd \leftarrow \text{NOT } Rd + 1$	Z, C, NV, H	1
SBR	Rd, k	Set Bits in Register	$Rd \leftarrow Rd \vee k$	Z, NV	1
CBR	Rd, k	Clear Bits in Register	$Rd \leftarrow Rd \& \text{NOT } k$	Z, NV	1
INC	Rd	Increment	$Rd \leftarrow Rd + 1$	Z, NV	1
DEC	Rd	Decrement	$Rd \leftarrow Rd - 1$	Z, NV	1
TST	Rd	Test for Zero or Minus	$Rd \leftarrow Rd \& Rd$	Z, NV	1
CLR	Rd	Clear Register	$Rd \leftarrow Rd \& 0$	Z, NV	1
SEI	Rd	Set Register	$Rd \leftarrow \text{NOT } Rd$	None	1
MUL	Rd, Rr	Multiply Unsigned	$Rd:Rr \leftarrow Rd \times Rr$	Z, C	2
MULS	Rd, Rr	Multiply Signed	$Rd:Rr \leftarrow Rd \times Rr$	Z, C	2
MULSU	Rd, Rr	Multiply Signed with Unsigned	$Rd:Rr \leftarrow Rd \times Rr$	Z, C	2
FMUL	Rd, Rr	Fastword Multiply Unsigned	$Rd:Rr \leftarrow (Rd \times Rr) \ll 1$	Z, C	2
FMULS	Rd, Rr	Fastword Multiply Signed	$Rd:Rr \leftarrow (Rd \times Rr) \ll 1$	Z, C	2
FMULSU	Rd, Rr	Fastword Multiply Signed with Unsigned	$Rd:Rr \leftarrow (Rd \times Rr) \ll 1$	Z, C	2
BRANCH INSTRUCTIONS					
RAMP	k	Relative Jump	$PC \leftarrow PC + k + 1$	None	2
LMP		Indirect Jump to Z	$PC \leftarrow Z$	None	2
JMP	k	Direct Jump	$PC \leftarrow k$	None	1
RCALL	k	Relative Subroutine Call	$PC \leftarrow PC + k + 1$	None	3
ICALL		Indirect Call to Z	$PC \leftarrow Z$	None	1
CALL	k	Direct Subroutine Call	$PC \leftarrow k$	None	4
RET		Subroutine Return	$PC \leftarrow \text{STACK}$	None	4
RETI		Interrupt Return	$PC \leftarrow \text{STACK}$	I	4
CPSE	Rd, Rr	Compare, Skip if Equal	$k \text{ (Rd) } \neq k \text{ (Rr)}$; $PC \leftarrow PC + 2 \text{ or } 3$	None	1, 2/3
CP	Rd, Rr	Compare	$Rd - Rr$	Z, N, V, C, H	1
CPSE	Rd, Rr	Compare with Carry	$Rd - Rr - C$	Z, N, V, C, H	1
CPH	Rd, k	Compare Register with Immediate	$Rd - k$	Z, N, V, C, H	1
SBRF	Rd, b	Skip if Bit in Register Cleared	$k \text{ (Rd)} \neq 0$; $PC \leftarrow PC + 2 \text{ or } 3$	None	1, 2/3
SBRD	Rd, b	Skip if Bit in Register is Set	$k \text{ (Rd)} = 1$; $PC \leftarrow PC + 2 \text{ or } 3$	None	1, 2/3
SBC	P, b	Skip if Bit in I/O Register Cleared	$k \text{ (P)} \neq 0$; $PC \leftarrow PC + 2 \text{ or } 3$	None	1, 2/3
SBS	P, b	Skip if Bit in I/O Register is Set	$k \text{ (P)} = 1$; $PC \leftarrow PC + 2 \text{ or } 3$	None	1, 2/3
BRBE	z, b	Branch if Status Flag Set	$k \text{ (SRF)} = 1$; $PC \leftarrow PC + k + 1$	None	1/2
BRBC	z, b	Branch if Status Flag Cleared	$k \text{ (SRF)} = 0$; $PC \leftarrow PC + k + 1$	None	1/2
BRBE	k	Branch if Equal	$k \text{ (Z)} = 1$; $PC \leftarrow PC + k + 1$	None	1/2
BRNE	k	Branch if Not Equal	$k \text{ (Z)} = 0$; $PC \leftarrow PC + k + 1$	None	1/2
BRCS	k	Branch if Carry Set	$k \text{ (C)} = 1$; $PC \leftarrow PC + k + 1$	None	1/2
BRCC	k	Branch if Carry Cleared	$k \text{ (C)} = 0$; $PC \leftarrow PC + k + 1$	None	1/2
BRSH	k	Branch if Same or Higher	$k \text{ (C)} = 0$; $PC \leftarrow PC + k + 1$	None	1/2
BRLO	k	Branch if Lower	$k \text{ (C)} = 1$; $PC \leftarrow PC + k + 1$	None	1/2
BRMI	k	Branch if Minus	$k \text{ (S)} = 1$; $PC \leftarrow PC + k + 1$	None	1/2
BRPL	k	Branch if Plus	$k \text{ (S)} = 0$; $PC \leftarrow PC + k + 1$	None	1/2
BRGE	k	Branch if Greater or Equal, Signed	$k \text{ (S)} \leq 1$; $PC \leftarrow PC + k + 1$	None	1/2
BRLT	k	Branch if Less Than Zero, Signed	$k \text{ (S)} \geq 1$; $PC \leftarrow PC + k + 1$	None	1/2
BRHS	k	Branch if Half Carry Flag Set	$k \text{ (H)} = 1$; $PC \leftarrow PC + k + 1$	None	1/2
BRHC	k	Branch if Half Carry Flag Cleared	$k \text{ (H)} = 0$; $PC \leftarrow PC + k + 1$	None	1/2
BRIS	k	Branch if I Flag Set	$k \text{ (I)} = 1$; $PC \leftarrow PC + k + 1$	None	1/2
BRIC	k	Branch if I Flag Cleared	$k \text{ (I)} = 0$; $PC \leftarrow PC + k + 1$	None	1/2
BRIS	k	Branch if Overflow Flag is Set	$k \text{ (O)} = 1$; $PC \leftarrow PC + k + 1$	None	1/2
BRIC	k	Branch if Overflow Flag is Cleared	$k \text{ (O)} = 0$; $PC \leftarrow PC + k + 1$	None	1/2

ATmega48A/48PA/88A/88PA/168A/168PA/328/328P

Mnemonics	Operands	Description	Operation	Flags	#Clocks
BRE	k	Branch if Interrupt Enabled	$Z(1=1) \text{ then } PC = PC + k + 1$	None	1/2
BRD	k	Branch if Interrupt Disabled	$Z(1=0) \text{ then } PC = PC + k + 1$	None	1/2
BIT AND BIT-TEST INSTRUCTIONS					
SBR	Pb	Set Bit in I/O Register	$IOP24 = 1$	None	2
CBR	Pb	Clear Bit in I/O Register	$IOP24 = 0$	None	2
LSL	Rd	Logical Shift Left	$Rdout(i) = Rdin, Rdin(i) = 0$	Z,C,N,V	1
LSR	Rd	Logical Shift Right	$Rdout(i) = Rdin(i), Rdin(i) = 0$	Z,C,N,V	1
RCL	Rd	Rotate Left Through Carry	$Rdout(i) = Rdin, Rdin(i) = Rdin(i-1) \oplus C$	Z,C,N,V	1
ROR	Rd	Rotate Right Through Carry	$Rdout(i) = Rdin(i-1) \oplus C, Rdin(i) = Rdin(i)$	Z,C,N,V	1
ASR	Rd	Arithmetic Shift Right	$Rdout(i) = Rdin(i), rd0 = 0$	Z,C,N,V	1
SWAP	Rd	Swap Nibbles	$Rdout(0) = Rdin(4), Rdout(4) = Rdin(0)$	None	1
SSET	s	Flag Set	$SREG(s) = 1$	SREG(s)	1
CLR	s	Flag Clear	$SREG(s) = 0$	SREG(s)	1
BST	Rd, b	Bit Store from Register to T	$T = Rd(b)$	T	1
BLD	Rd, b	Bit Load from T to Register	$Rd(b) = T$	None	1
SEC		Set Carry	$C = 1$	C	1
CLC		Clear Carry	$C = 0$	C	1
SEN		Set Negative Flag	$N = 1$	N	1
CLN		Clear Negative Flag	$N = 0$	N	1
SEZ		Set Zero Flag	$Z = 1$	Z	1
CLZ		Clear Zero Flag	$Z = 0$	Z	1
SEI		Global Interrupt Enable	$I = 1$	I	1
CLI		Global Interrupt Disable	$I = 0$	I	1
SES		Set Signed Test Flag	$S = 1$	S	1
CLS		Clear Signed Test Flag	$S = 0$	S	1
SEV		Set Two's Complement Overflow	$V = 1$	V	1
CLV		Clear Two's Complement Overflow	$V = 0$	V	1
SET		Set T in SREG	$T = 1$	T	1
CLT		Clear T in SREG	$T = 0$	T	1
SEH		Set Half-Carry Flag in SREG	$H = 1$	H	1
CLH		Clear Half-Carry Flag in SREG	$H = 0$	H	1
DATA TRANSFER INSTRUCTIONS					
MOV	Rd, Rr	Move Between Registers	$Rd = Rr$	None	1
MOVW	Rd, Rr	Copy Register Word	$Rd1:Rd0 = Rr1:Rr0$	None	1
LDI	Rd, K	Load Immediate	$Rd = K$	None	1
LDI	Rd, X	Load Indirect	$Rd = (X)$	None	2
LDI	Rd, X+	Load Indirect and Post-Inc.	$Rd = (X), X = X + 1$	None	2
LDI	Rd, -X	Load Indirect and Pre-Dec.	$X = X - 1, Rd = (X)$	None	2
LDI	Rd, Y	Load Indirect	$Rd = (Y)$	None	2
LDI	Rd, Y+	Load Indirect and Post-Inc.	$Rd = (Y), Y = Y + 1$	None	2
LDI	Rd, -Y	Load Indirect and Pre-Dec.	$Y = Y - 1, Rd = (Y)$	None	2
LDD	Rd, Y+q	Load Indirect with Displacement	$Rd = (Y + q)$	None	2
LDD	Rd, Z	Load Indirect	$Rd = (Z)$	None	2
LDD	Rd, Z+	Load Indirect and Post-Inc.	$Rd = (Z), Z = Z + 1$	None	2
LDD	Rd, -Z	Load Indirect and Pre-Dec.	$Z = Z - 1, Rd = (Z)$	None	2
LDD	Rd, Z+q	Load Indirect with Displacement	$Rd = (Z + q)$	None	2
LDS	Rd, k	Load Direct from SRAM	$Rd = Rk$	None	2
ST	X, Rr	Store Indirect	$(X) = Rr$	None	2
ST	X+, Rr	Store Indirect and Post-Inc.	$(X) = Rr, X = X + 1$	None	2
ST	-X, Rr	Store Indirect and Pre-Dec.	$X = X - 1, (X) = Rr$	None	2
ST	Y, Rr	Store Indirect	$(Y) = Rr$	None	2
ST	Y+, Rr	Store Indirect and Post-Inc.	$(Y) = Rr, Y = Y + 1$	None	2
ST	-Y, Rr	Store Indirect and Pre-Dec.	$Y = Y - 1, (Y) = Rr$	None	2
STD	Y+q, Rr	Store Indirect with Displacement	$(Y + q) = Rr$	None	2
STD	Z, Rr	Store Indirect	$(Z) = Rr$	None	2
STD	Z+, Rr	Store Indirect and Post-Inc.	$(Z) = Rr, Z = Z + 1$	None	2
STD	-Z, Rr	Store Indirect and Pre-Dec.	$Z = Z - 1, (Z) = Rr$	None	2
STD	Z+q, Rr	Store Indirect with Displacement	$(Z + q) = Rr$	None	2
STS	k, Rr	Store Direct to SRAM	$Rk = Rr$	None	2
LPM		Load Program Memory	$R0 = (Z)$	None	3
LPM	Rd, Z	Load Program Memory	$Rd = (Z)$	None	3
LPM	Rd, Z+	Load Program Memory and Post-Inc.	$Rd = (Z), Z = Z + 1$	None	3
SPM		Store Program Memory	$(Z) = R1:R0$	None	-
IN	Rd, P	In Port	$Rd = P$	None	1
OUT	P, Rr	Out Port	$P = Rr$	None	1
PUSH	Rr	Push Register on Stack	$STACK = Rr$	None	2

Mnemonic	Operands	Description	Operation	Flags	#Cycles
POP	Rd	Pop Register from Stack	Pop ← STACK	None	2
MCU CONTROL INSTRUCTIONS					
RFCF		No Operation		None	1
SLEEP		Sleep	(see specific device for Sleep function)	None	1
WDR		Watchdog Reset	(see specific device for WDR function)	None	1
BREAK		Break	For On-chip Debug Only	None	8A

Note: 1. These instructions are only available in ATmega168PA and ATmega328P.

GP2Y0A21YK0F

Distance Measuring Sensor Unit
 Measuring distance: 10 to 80 cm
 Analog output type



■ Description

GP2Y0A21YK0F is a distance measuring sensor unit, composed of an integrated combination of PSD (position sensitive detector), IRED (infrared emitting diode) and signal processing circuit.

The variety of the reflectivity of the object, the environmental temperature and the operating duration are not influenced easily to the distance detection because of adopting the transmission method.

This device outputs the voltage corresponding to the detection distance. So that sensor can also be used as a proximity sensor.

■ Features

1. Distance measuring range: 10 to 80 cm
2. Analog output type
3. Package size: 29.5 x 13 x 10.5 mm
4. Consumption current: Typ. 30 mA
5. Supply voltage: 4.5 to 5.1 V

■ Agency approvals/Compliance

1. Compliance with RoHS directive (2002/95/EC)

■ Applications

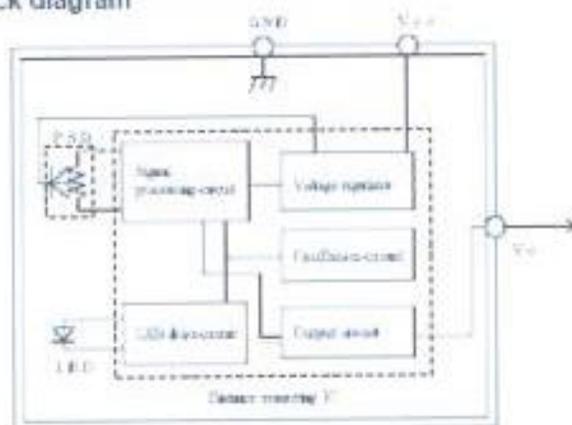
1. Touch-less switch
(Sanitary equipment, Control of illumination, etc.)
2. Robot cleaner
3. Sensor for energy saving
(ATM, Copier, Vending machine)
4. Amusement equipment
(Robot, Arcade game machine)

Notes: The content of this sheet is subject to change without prior notice.

In the absence of contradictory device specifications sheet, SHARP takes no responsibility for any defects that occur in equipment using this SHARP device. Device failure is possible due to such as: Cause: (1) Use of a device in an environment that is not specified in the device specifications sheet. (2) Use of a device in an environment that is not specified in the device specifications sheet.

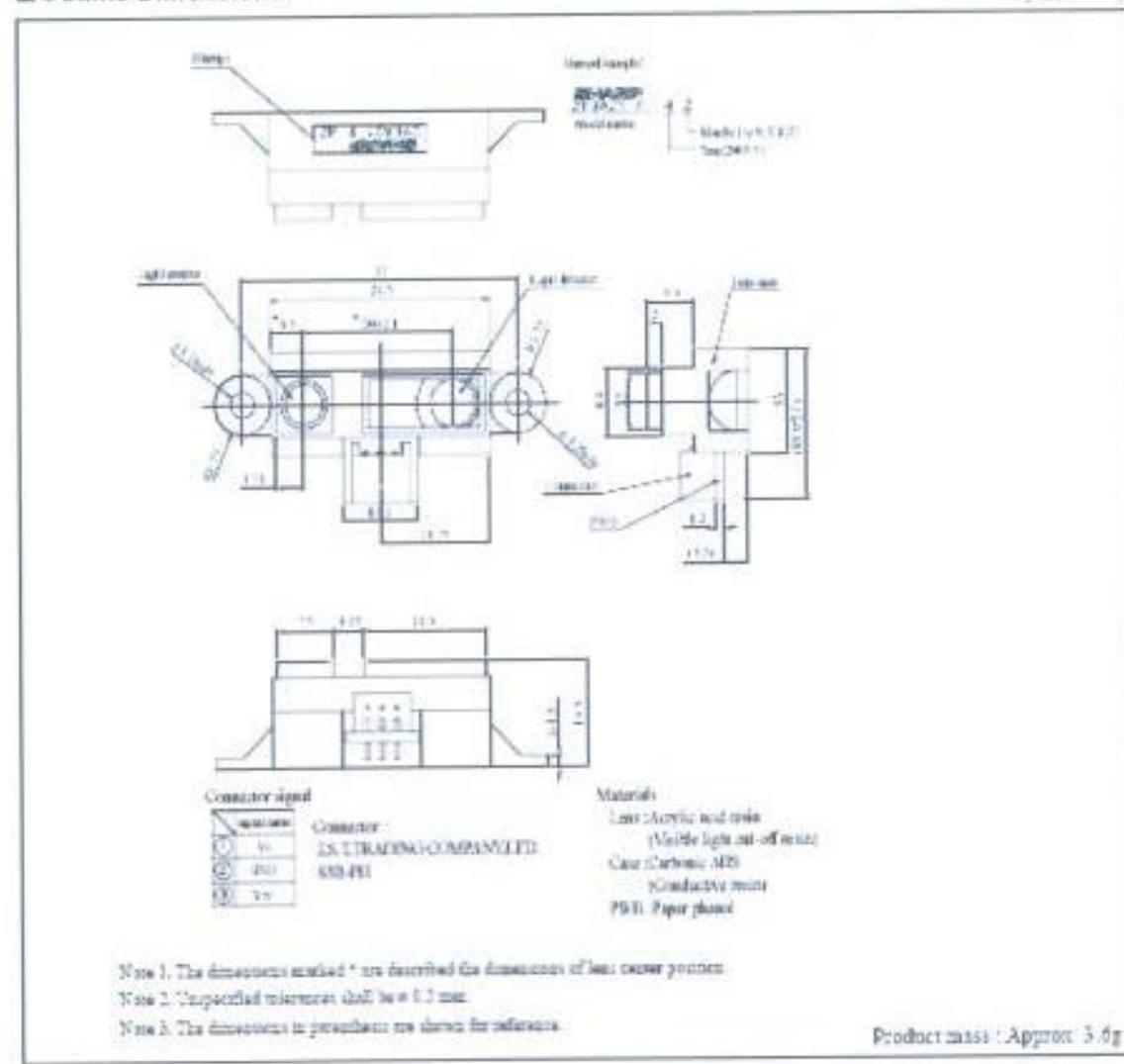
Class 5D E-ACC21185
 Date: Dec. 11, 2014
 © SHARP Corporation

■ Block diagram



■ Outline Dimensions

(Unit: mm)



Sheet 1/1 GP2Y0A21YK0F

■ Absolute Maximum Ratings (T_a=25°C, V_{CC}=5V)

Parameter	Symbol	Range	Unit
Supply voltage	V _{CC}	-0.3 to 7	V
Output terminal voltage	V _{OL}	-0.3 to V _{CC} +0.3	V
Operating temperature	T _{OPR}	-10 to +60	°C
Storage temperature	T _{STG}	-40 to +70	°C

■ Electro-optical Characteristics (T_a=25°C, V_{CC}=5V)

Parameter	Symbol	Conditions	MIN.	Typ.	MAX.	Unit
Average supply current	I _{CC}	L=80cm (Note 1)	—	30	40	mA
Distance measuring	ΔL	(Note 1)	30	—	60	cm
Output voltage	V _{OL}	L=80cm (Note 1)	0.25	0.4	0.55	V
Output voltage differential	ΔV _{OL}	Output voltage difference between L=10cm and L=80cm (Note 1)	1.65	1.8	2.15	V

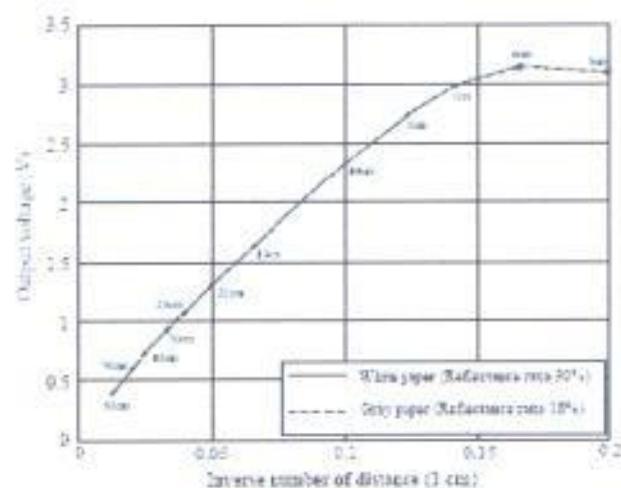
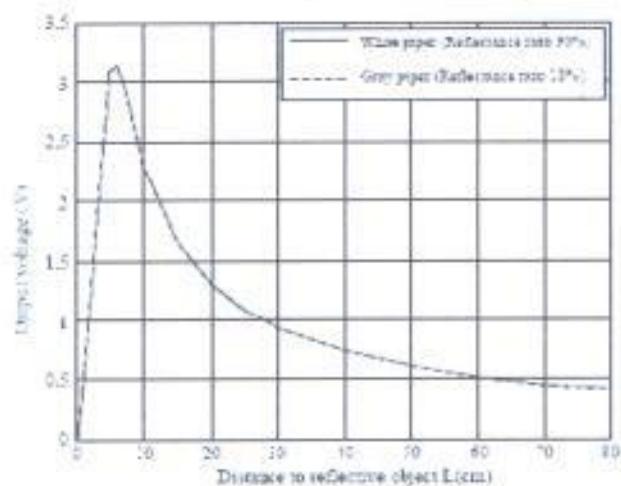
* L : Distance to reflective object

Note 1 : Using reflective object: White paper (Made by Kodak Co., Ltd. gray card K-17) white face, reflectance: 90%

■ Recommended operating conditions

Parameter	Symbol	Range	Unit
Supply voltage	V _{CC}	4.5 to 5.5	V

Fig. 2 Example of distance measuring characteristics(output)



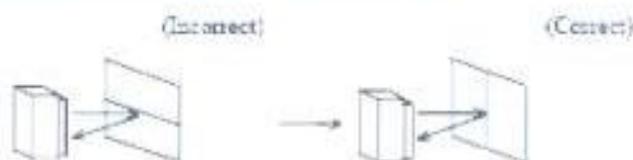
Notes

● Advice for the optics

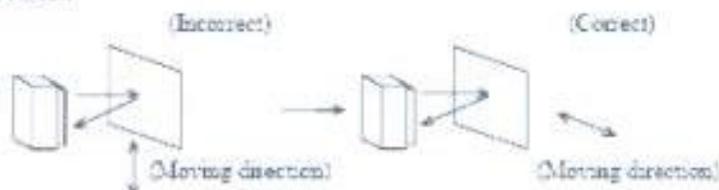
- The lens of this device needs to be kept clean. There are cases that dust, water or oil and so on deteriorate the characteristics of this device. Please consider in actual application.
- Please don't do washing. Washing may deteriorate the characteristics of optical system and so on. Please confirm resistance to chemicals under the actual usage since this product has not been designed against washing.

● Advice for the characteristic

- In case that an optical filter is set in front of the emitter and detector portion, the optical filter which has the most efficient transmittance at the emitting wavelength range of LED for this product ($\lambda = 870 \sim 900\text{nm}$) shall be recommended to use. Both faces of the filter should be mirror polishing. Also, as there are cases that the characteristics may not be satisfied according to the distance between the protection cover and this product or the thickness of the protection cover, please use this product after confirming the operation sufficiently in actual application.
- In case that there is an object near to emitter side of the sensor between sensor and a detecting object, please use this device after confirming sufficiently that the characteristics of this sensor do not change by the object.
- When the detector is exposed to the direct light from the sun, mercury lamp and so on, there are cases that it can not measure the distance exactly. Please consider the design that the detector is not exposed to the direct light from such light source.
- Distance to a mirror reflector can not be sometimes measured exactly.
- In case of changing the mounting angle of this product, it may measure the distance exactly.
- In case that reflective object has boundary line which material or color etc. are excessively different, in order to decrease deviation of measuring distance, it shall be recommended to set the sensor that the direction of boundary line and the line between emitter center and detector center are in parallel.



- In order to decrease deviation of measuring distance by moving direction of the reflective object, it shall be recommended to set the sensor that the moving direction of the object and the line between emitter center and detector center are vertical.



● Advice for the power supply

- In order to stabilize power supply line, we recommend to insert a by-pass capacitor of 10 μF or more between Vcc and GND near this product.

● Notes on handling

- There are some possibilities that the internal components in the sensor may be exposed to the excessive mechanical stress. Please be careful not to cause any excessive pressure on the sensor package, and also on the PCB while assembling this product.

BIBLIOGRAFÍA

1. Oscar González

COMO EMPEZAR CON EL POLOLU

<http://blog.bricogeek.com/noticias/robotica/como-empezar-con-el-sigue-lineas-3pi-de-pololu/>

05 de Enero del 2011.

2. Pololu Corporation

SENSOR SHARP

<http://www.pololu.com/catalog/product/136>

12 Febrero 2010

3. Pololu Corporation

SAMPLE PROJECT POLOLU

http://www.sparkfun.com/datasheets/Robotics/3pi_wall_follower.pdf

15 de febrero 2011

4. Quickanddirty

POLOLU 3PI

<http://code.google.com/p/quickanddirty/wiki/Pololu3piRobot>

23 de febrero de 2011

5. R.Steven Rainwater, Roger Arrick y Jim Brown

ROBOT POLOLU

<http://robots.net/article/2866.html>,

25 de febrero de 2009

6. BricoGeek

SEGUIDOR DE LINEA POLOLU

<http://www.bricogeek.com/shop/106-seguidor-de-lineas-pololu-3pi.html>,

27 febrero 2011.

7. Simple Machines

LLC BLOG SHARP

<http://www.societyofrobots.com/robotforum/index.php?topic=12011.0>,

01 Septiembre de 2010.

8. Atmel

AVR STUDIO 4

http://www.atmel.com/dyn/products/tools_card.asp?tool_id=2725,

02 MARZO 2008

9. Atmel

DATASHEET ATTINY 13

http://www.atmel.com/dyn/resources/prod_documents/doc2535.pdf,

15 OCTUBRE, 2009

10. Rob's ElectroSoft aka

RES ATTINY PROJECT

<http://www.attiny.com/>

25 SEPTIEMBRE DE 2010

11. The PNphpBB Group

ATTINY EXAMPLE

[http://www.avrfreaks.net/index.php?name=PNphpBB2&file=viewtopic&the
me=print&p=805256](http://www.avrfreaks.net/index.php?name=PNphpBB2&file=viewtopic&the
me=print&p=805256)

5 ENERO DE 2011

12. Adafruit

ATTINY EASY EXAMPLE

<http://forums.adafruit.com/viewtopic.php?f=24&t=4800>

15

FEBRERO

DE

201