



Control de robot mediante joystick utilizando el Kit AVR Butterfly en interfaz inalámbrica por radio frecuencia con el Pololu 3pi.

Gerardo Alberto Villagómez Galarza⁽¹⁾, Rodrigo Salomón Jurado Echeverría⁽²⁾, Carlos Valdivieso⁽³⁾
Facultad de Ingeniería Eléctrica y Computación (FIEC)
Escuela Superior Politécnica de Litoral (ESPOL)
Campus Gustavo Galindo, Km 30.5 Vía Perimetral
Apartado 09-01-5863 Guayaquil, Ecuador
gerviga_86@hotmail.com⁽¹⁾, rjurado1986@hotmail.com⁽²⁾, cvaldiv@espol.edu.ec⁽³⁾

Resumen

El propósito de este trabajo es realizar un sistema de robot controlado por un joystick en interfaz inalámbrica de radiofrecuencia con la finalidad de obtener un robot que se mueva a nuestras órdenes.

El robot utilizado para este proyecto es el pololu 3pi debido a ser uno de los más completos en comparación a otros ya que contiene varias partes de esta misma línea que se adaptan a este según el uso que uno quiera, como también su programación que se realiza en Lenguaje C que es uno de los lenguajes más conocidos en el ámbito de la programación.

Este sistema consta de un robot seguidor de línea comandado por el joystick de la tarjeta AVR Butterfly que hace que nuestro robot se mueva a todas las direcciones según como nosotros le demos la orden, y todo esto lo haremos por radiofrecuencia con la intención de evitarnos la conexión por cables y tener un mayor alcance de comunicación.

En el mundo de la robótica existen varios productos que realizan esta función y en comparación a nuestro proyecto su costo es muy alto debido a que estos hay que incorporarles cada una de las partes como son las ruedas, los motores, los sensores, etc. ya que todo hay que obtenerlo por separado y esto hace que su precio sea elevado, no como en el robot pololu que ya viene todo esto incorporado, por tal motivo se decidió utilizarlo en este proyecto como una mejor solución.

Palabras Claves: Robot, Joystick, Inalámbrica, Radiofrecuencia, Pololu, Lenguaje C, Butterfly, Direcciones.

Abstract

The purpose of this work is to control a robot by a joystick in RF wireless interface in order to get a robot to move to our orders.

The robot used for this project is a Pololu 3pi, one of the most complete compared to others because it contains parts of the same line that are tailored according to the use you want. Also, its programming is done in C language which is one of the most popular programming languages.

This system consists of a line follower robot commanded by a joystick from AVR Butterfly card which makes our robot move in all directions according to the order. All this is done by radio with the intention of avoiding cable connection and have a greater communication scope.

In the world of robotics, there are several products that perform this function and compared with our project their cost is very high because they must be incorporate different parts such as of wheels, motors, sensors, etc.

Since everything must be obtained separately, this makes their price higher in comparison to the pololu robot which is already built with all this, for this reason we decided to use it with this project as a better solution.

Keys words: Robot, Joystick, Wireless, Radiofrequency, Pololu, C language, Butterfly, Directions.

DESCRIPCION GENERAL DEL SISTEMA

1 Descripción del Proyecto

En esta tesis se aborda el problema del control de robots mediante la utilización de un joystick en interfaz de radiofrecuencia y se presenta la técnica usada normalmente para tratar este problema.

El desarrollo se realiza para un robot conocido en el ámbito académico, como es el Pololu 3pi de Atmel acompañado del kit AVR Butterfly. [5] En este caso, se aprovechan las facilidades que existen para el control de dicho robot, de forma que se llega a un modelo capaz de moverse a lo largo y ancho de la pista controlando su dirección de movimiento por medio de un joystick por radiofrecuencia.

1.1 Estrategia Implementada

Como estrategia implementada en este proyecto los elementos que utilizaremos son el robot pololu 3pi [1] por ser una plataforma completa de alto rendimiento móvil, además por tener algunos componentes interactuando entre ellos con los que se puede trabajar, tales como sensores, LCD, timbre, botones de usuario, todos estos conectados a un microcontrolador ATmega328 C-programable. Este robot es capaz de alcanzar velocidades superiores a 3 pies por Segundo.

Otro elemento utilizado es el KIT AVR Butterfly que es una tarjeta que consta con un microcontrolador Atmega169PV. Este KIT está formado por algunos componentes los más destacados son una LCD con 120 segmentos, un Joystick de cuatro direcciones, y un termistor NTC para medir temperatura del cual utilizaremos el joystick para controlar el movimiento del robot pololu a través de radiofrecuencia[3].

1.2 Alcance y limitaciones del proyecto

Mediante pruebas realizadas a nuestro proyecto hemos encontrado que tiene sus alcances y limitaciones entre las cuales podemos denotar que:

1.2.1 ALCANCES

- Su sistema único de potencia le permite moverse a velocidades de hasta 100 cm/segundo.
- Es de tamaño compacto mide 95 mm de diámetro y pesa 83 g sin baterías el cual facilita su funcionalidad.
- El robot viene completamente ensamblado es decir no hay necesidad de adaptar las ruedas, los motores, los sensores, etc.

- Es de fácil codificación ya que trabaja con uno de los más usados lenguajes de programación que es el “Lenguaje C”.

1.2.2 LIMITACIONES

- Es muy sensible en su manipulación ya que podemos inducirle estática y podemos dañarlo en su totalidad.
- Contiene elementos muy delicados que con cualquier golpe pueden ser averiados parcial o completamente.

FUNDAMENTACION TEORICA

2 Descripción

En el presente capítulo se exponen las herramientas de Hardware y Software que se usan para el correcto funcionamiento de nuestro proyecto, entre las cuales mostraremos las características más importantes como son:

El microcontrolador Atmega 328P, Micro Gearmotors, la comunicación por medio de la tarjeta AVR Butterfly, la alimentación de nuestro robot, el software AVR Studio versión 4.18 [2] y nuestro simulador Proteus versión 7.7.



Figura 2-1: Pololu Vista Superior

2.1 Herramientas de Hardware

2.1.1 Microcontrolador

El microcontrolador que usa este robot Pololu 3pi es de marca ATMEL ATmega 328P que corre a 20 MHz de frecuencia, funciona con una memoria Flash de 32 KB, 2KB de memoria en RAM y 1 KB de memoria EEPROM.

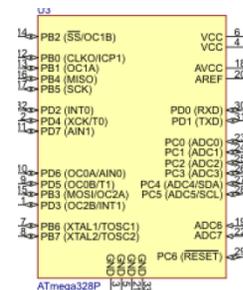


Figura 2-2: Microcontrolador Atmega 328P

2.1.2 Comunicación

La comunicación que utiliza nuestro proyecto es mediante interfaz inalámbrica por radiofrecuencia en la cual interviene nuestro kit AVR Butterfly que contiene el joystick con el que controlaremos los movimientos de nuestro robot pololu 3pi.

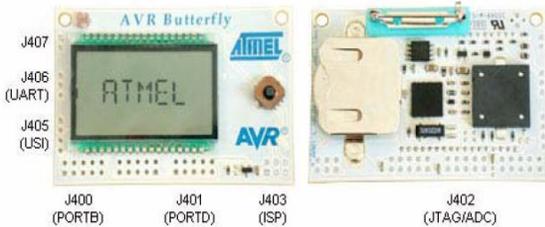


Figura 2-3: Tarjeta Butterfly Vista Superior e Inferior

2.1.3 Motores

El robot pololu está totalmente ensamblado con dos pequeños motores de metal para las ruedas los cuales tienen excelentes características para nuestro robot entre las cuales tenemos que:

Gear ratio:	30:1
Free-running speed:	700 rpm
Free-running current:	60 mA
Stall torque:	6 oz·in
Stall current:	540 mA

Tabla 2-1: Características del motor



Figura 2-4: Motor del pololu

2.1.4 Alimentación

La alimentación utilizada para nuestro robot consta de 4 pilas AAA que nos da un voltaje aproximado de 6 voltios DC que además son pequeñas, es decir no necesitamos de altos voltajes continuos que nos proporcionarían una gran cantidad de pilas ni usamos las pilas doble AA que son más pesadas.



Figura 2-5: Pilas Alcalinas AAA

2.2 Herramientas de Software



Figura 2-6: Logotipo del AVR Studio

2.2.1 Introducción

El Software AVR Studio es un entorno de desarrollo integrado para la escritura y depuración que tiene aplicaciones en Windows 98, XP, Me, 2000, NT, 7. AVR Studio proporciona una herramienta de gestión del proyecto, archivo de origen, editor y simulador de chip. También interactúa con Emuladores de circuitos.

Este software simplifica el desarrollo de tareas, permite a los usuarios reducir significativamente el tiempo de trabajo y genera un mejor ambiente para la realización de los proyectos.

Entre las características más notables tenemos:

- Entorno de desarrollo integrado para escribir, compilar y depurar.
- Reproducción de memorias configurables incluyendo SRAM, EEPROM, Flash, Registros y E/S.
- Completamente simbólico a nivel de depuración.
- Número ilimitado de puntos de separación (breakpoints).
- Ayuda HTML en línea.
- Amplio programa para opciones de control de flujo.
- Simulación de actividad del puerto y estímulos de entrada al pin.
- Apoyo para C, Pascal y lenguaje ensamblador.

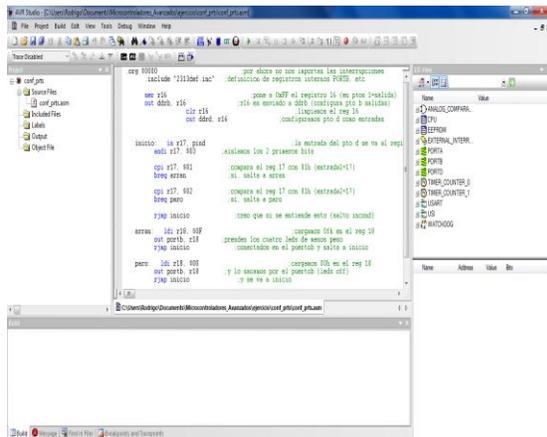


Figura 2-7: Entorno de AVR Studio

2.2.2 Simulador Proteus versión 7.7

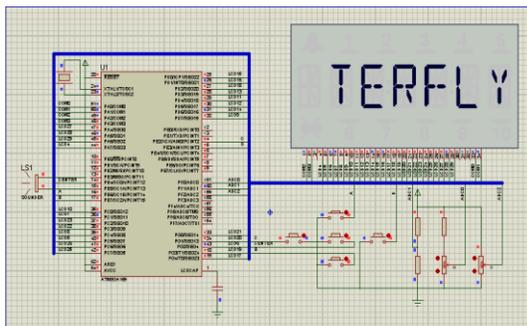


Figura 2-8: Entorno Proteus 7.7

Proteus versión 7.7 es un entorno completo diseñado para la realización de cualquier clase de proyecto que conste de elementos electrónicos en todas sus etapas: Diseño, Simulación y Construcción.

Consta de dos partes:

ISIS.-Permite diseñar y simular cualquier clase de proyecto electrónico para constatar su funcionamiento.

Ares.- Permite el diseño de ruteo del proyecto para ahora si plasmarlo en una placa electrónica. Esta herramienta te ayuda a buscar la mejor forma de cómo colocar los elementos y las líneas del circuito en la placa.

3. DESCRIPCION E IMPLEMENTACION DEL PROYECTO

3.1 Diseño Preliminar

La implementación del proyecto fue dividida en dos partes:

Transmisor:

1. Diseñamos una fuente de +5 voltios mediante un regulador de voltaje LM7805 que tiene como

función tomar un voltaje de entrada hasta de +40 voltios y setear a un voltaje de salida de +5 voltios.

2. Probamos nuestra tarjeta butterfly mediante el hyperterminal de nuestra PC y mediante un osciloscopio para verificar si realmente está transmitiendo como debe ser los caracteres y las tramas respectivamente.
3. Diseñamos un circuito conversor de señal de RS232 a señal TTL mediante un integrado MAX232 [4], en el cual ingresaremos la señal de salida de nuestra butterfly para obtenerla en TTL y así ingresarla a nuestro módulo de radiofrecuencia.
4. Usamos como módulos de radiofrecuencia los HMTR debido que son inmunes al ruido y por ende nuestra transmisión será casi en un 100%.

Receptor:

1. En la etapa receptora tenemos como parte principal nuestro robot que es el que va a recibir las ordenes de cómo y en qué dirección debe moverse.
2. Diseñamos el circuito para nuestro módulo receptor HTMR polarizado con el voltaje que nos proporciona el robot Pololu debido a que este tiene unos pines de VCC y GND donde podemos obtenerlo.

3.2 Implementación Física

Para comenzar la implementación física de nuestro proyecto, comenzamos armando en un protoboard toda nuestra circuitería empezando por nuestra fuente de +5v.

- La fuente de +5v será utilizada para la alimentación del módulo de radiofrecuencia y para la polarización de nuestro circuito max232.

Luego tenemos la implementación de nuestro circuito max232 que recibirá la señal de salida del UART de nuestra tarjeta AVR butterfly el cual la convertirá en señal TTL.

A continuación esta señal será trasladada a nuestro módulo transmisor HTMR que por radiofrecuencia se comunicará con nuestro módulo receptor HTMR que se encuentra en nuestro robot Pololu.

Debemos darnos cuenta en la conexión de nuestros módulos HTMR que entre los pines de transmisión y recepción estos no se conectan en serie sino cruzados.

3.3 Descripción del proyecto final

Al finalizar con lo que es la programación y la implementación del proyecto podemos hacer una breve descripción del funcionamiento del mismo:

1. En la tarjeta AVR butterfly se muestra al principio la frase “Control Remoto” y se espera hasta que el usuario presione o mueva en cualquiera de las direcciones el joystick para continuar.
2. Una vez presionado o movido el joystick según sea la orden que se haya dado se mostrará la palabra según sea la orden dada, es decir si se presiona saldrá la palabra “Centro”, si se mueve en cualquiera de las direcciones según sea se mostrará las palabras “Arriba”, “Abajo”, “Izquierda”, “Derecha”.
3. Una vez presionado o movido el joystick este por medio del UART transmitirá las ordenes hacia el receptor que es nuestro robot Pololu para indicarle que debe hacer.
4. Ahora en el receptor nuestro robot Pololu al encenderse mostrará “Press B to Start” en donde para inicializarlo debemos realizar esta orden, una vez presionado “B” nos mostrará la frase “Espero Orden” y esto nos indica que está esperando la orden del transmisor o sea de nuestra tarjeta Butterfly.
5. Una vez que está esperando el robot la orden debemos comandar la tarjeta, es decir mover o presionar el joystick según lo que quisiéramos que haga el robot para que este una vez recibida la orden cumpla con aquello.
6. Y listo nuestro robot se moverá por comunicación inalámbrica con interfaz de radiofrecuencia según la dirección en que quisiéramos que se mueva.

3.4 Diagrama de bloques del proyecto

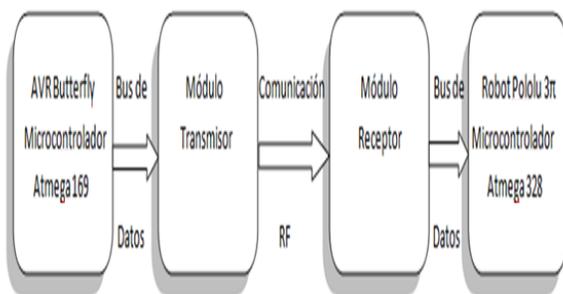


Figura 3-1: Diagrama de bloques del proyecto

4. SIMULACION Y PRUEBAS

4.1 Simulación en Proteus

Gracias a la ayuda de un excelente programa simulador llamado Proteus nosotros pudimos virtualmente ver cómo funciona nuestro proyecto, si todo lo que queremos que realice lo hace en su totalidad o si algo está fallando, darnos cuenta paso a paso lo que está pasando en las diferentes etapas del proyecto.

En esta primera parte vamos a notar como trabaja la parte del transmisor, es decir el funcionamiento de nuestra tarjeta Avr Butterfly como envía el caracter para que así nuestro receptor pueda recibir la orden de movimiento.

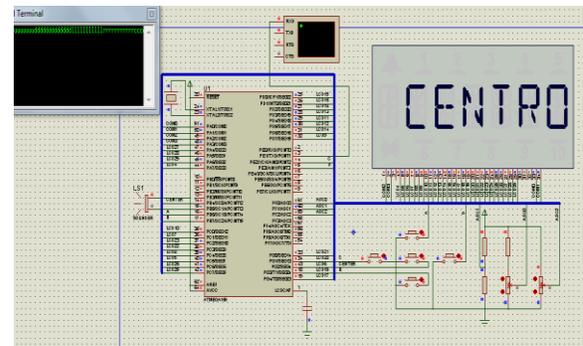


Figura 4-1: Simulación del transmisor

En la figura 4-1 podemos darnos cuenta claramente como al presionar cualquier botonera se muestra en la pantalla de la LCD el nombre de la dirección según cual hayamos manipulado por ejemplo fue presionada la botonera del centro se presenta la palabra centro.

Además podemos observar también la pantalla del hyperterminal del proteus que nos indica la aparición de un carácter relacionado con cada botonera es decir aparece el carácter “c” “u” “d” “l” “r”.

- “c” = botonera del centro
- “d” = botonera de abajo
- “u” = botonera de arriba
- “l” = botonera de la izquierda
- “r” = botonera de la derecha

En la figura 4-2 podemos ver la simulación del receptor o sea nuestro robot pololu 3Pi en la cual se muestra en la pantalla del robot la frase “Press B to Start” luego de esto debemos presionar el botón B para iniciar el programa, a continuación sale el mensaje “Espero orden” donde el robot está esperando la dirección a la cual debe moverse



Figura 4-9a: Robot se mueve en círculo



Figura 4-9b: Robot se mueve en círculo

5. CONCLUSIONES

1. El Kit AVR Butterfly es una poderosa herramienta de aprendizaje, es práctico, eficaz y muy amigable; que con el desarrollo de las prácticas va descubriendo progresivamente las características del microcontrolador ATmega169.
2. La característica de controlador LCD, del microcontrolador ATmega169, permite abaratar costos en la implementación de aplicaciones que necesitan despliegue de información a través de LCD. Esta característica permite controlar pantallas de cristal líquido de bajo costo, que por ser básicas no poseen ni driver interno, ni interfaces de comunicación como la mayoría de los costosos módulos LCD.
3. El Bootloader del ATmega169, combinado con el circuito convertidor de nivel RS-232 incluido en el Kit AVR Butterfly, permite la funcionalidad de Programación dentro del Sistema (In-System Programming, ISP); en este caso, entiéndase como Sistema al Kit AVR Butterfly. ISP permite programar al microcontrolador sin la necesidad de extraerlo del sistema.

4. En cuanto a nuestro robot pololu podemos decir que es muy dinámico ya que su pequeña dimensión y su liviano peso, le permite que se mueva sin inconvenientes según la orden que le demos, no tiene problema si su voltaje disminuye una pequeña cantidad como en otros robot que si no reciben el voltaje exacto que necesitan no trabajan con eficacia.

6. RECOMENDACIONES

1. No apoyar el Kit AVR Butterfly en superficies conductoras tales como metal, líquidos, etc., puesto que podrían causar daños en el mismo.
2. No conectar cables directamente en los espacios para conexiones externas del Kit, ya que podrían causar cortocircuito; en su lugar, colocar espadines hembras.
3. Simular el dispositivo y depurar el código fuente simultáneamente en AVR Studio, antes de descargar el software en el dispositivo. Con esto, se evita daños en el dispositivo por errores en la codificación del software.
4. Al momento de codificar software en lenguaje C, es recomendable segmentar el código fuente en funciones especializadas, esto quiere decir que cada función realice una sola tarea específica; de este modo se podrán utilizar las mismas funciones en otras aplicaciones.

BIBLIOGRAFIA

- [1]. Pololu Robotics & Electronics.
<http://www.pololu.com/catalog/product/975>
Enero 10 del 2011
- [2]. Atmel AVR Studio
<http://www2.atmel.com/>
Enero 13 del 2011
- [3]. Product news
http://www.atmel.com/dyn/products/tools_card.asp?tool_id=2725
Enero 20 del 2011
- [4]. Eduardo J. Carletti
http://robots-argentina.com.ar/Comunicacion_max232.htm
Enero 23 del 2011
- [5]. AVR Butterfly
http://en.wikipedia.org/wiki/AVR_Butterfly
Febrero 15 del 2011