

BRAZO ARTICULABLE TALADRADOR MOVIBLE CONTROLADO POR MODULO BLUETOOTH

A. Carrión, R. Sánchez, Adviser R. Palomares, members Ricardo Palma University-Perú,
hucatho@hotmail.com

Abstract: *This project is developed with the intention of simulating drilling tasks in mining areas; Remote communications can prevent accidents with the operator. The implementation of this project is with components accessible in the Lima-Peru market.*

Palabras Claves.- Arduino, Acelerómetro, Puente H, Motor Reductor, Bluetooth.

Introducción

El desarrollo de maquinarias que facilitan tareas al hombre, son implementadas primero en simuladores, maquetas, proyectos a escala, etc. Es por esto, el interés de ejecutar un brazo articulable taladrador móvil controlado por bluetooth. Este proyecto, permitirá facilitar corregir todos los posibles problemas que pudiera implicar en su construcción oficial; además, descartar los atentados, al integridad física del operario, que puede haber en la conducción del brazo robot.

Este proyecto, de cuatro grados libertad, es compuesto por motores reductores DC para la articulación de cada eslabón del brazo robot; además, tiene cuatro motores que le da la opción de desplazamiento de un área a otra y un motor de altas revoluciones que simula la tarea de un taladro. Todos estos actuadores están controlados por tres arduinos; estos trabajan de manera independiente; a su vez, tiene como periférico de entrada a módulos bluetooth. Todo esto, montado en una estructura de aluminio de cuatro extensiones.

Se espera llegar a cumplir con todos los objetivos posibles y descartar algunos inconvenientes posibles que se puedan suscitar en el desarrollo del mismo.

Se espera cumplir todas las expectativas posibles del observador y ser competitivo con las otras máquinas del mercado.

Desarrollo

El desarrollo del proyecto se va a ejecutar en tres etapas: parte mecánica, electrónica y programación. Fig.1.

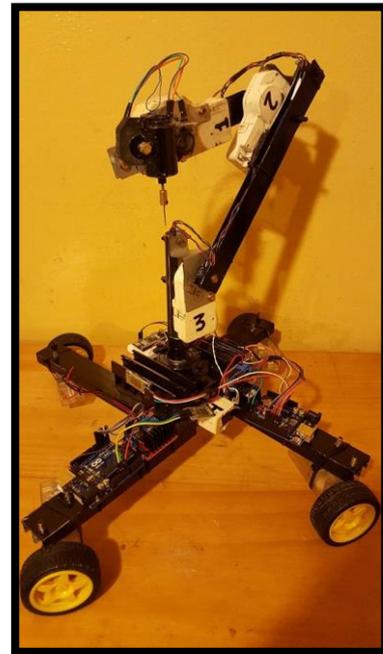


Figura 1. Vista principal del robot

Parte Mecánica:

Primeramente, se tiene una base con cuatro extensiones y en cada extensión está montado un motor que, en conjunto con otros, le da la facultad de desplazamiento al robot. Esta base tiene la particularidad de tener un eje central con un rodamiento radial, a su vez, se instaló un actuador; en el cual, se muestra el brazo del robot. Fig.2

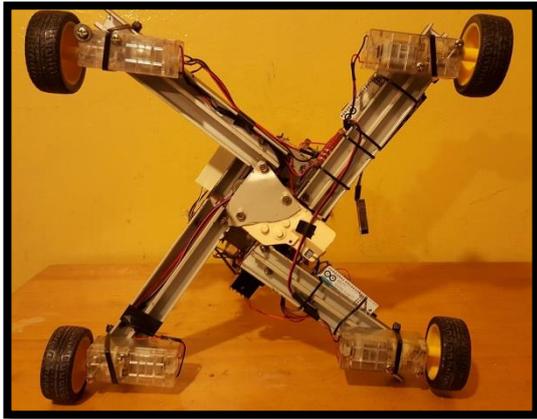


Figura 2. Vista del sistema locomotor del robot

Posteriormente, se tiene tres eslabones de aluminio, en cada uno de ellos, se implementó a un actuador por cada eslabón; estos proporcionan movimientos radiales.

Por último, en el extremo del robot, se instaló el motor de altas revoluciones, acompañado, por un acelerómetro; donde este último, tendrá la función de dar la posición del taladro; el cual, tiene que estar verticalmente. Fig. 3



Figura 3. Vista del extremo del robot

Parte Electrónica:

Todos estos actuadores están subordinados con tres tarjetas de control denominado comercialmente Arduino.

Primera tarjeta, se tiene al Arduino Nano que controla el extremo del robot; este tiene que mantener una posición vertical; El controlador

recibe información, en coordenadas cartesianas (x, y, z), del acelerómetro ADXL 345; para luego procesarlas y dar orden al actuador del extremo del robot. Fig.4

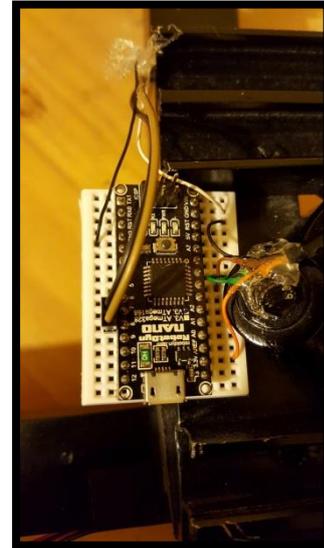


Figura 4. Vista del acelerómetro ADXL 345

Segunda tarjeta, se tiene al Arduino Uno que controla el desplazamiento del robot de un lugar a otro. Además, tiene el mando de manejo a un módulo bluetooth como periférico de entrada de datos; este módulo, se conecta a un celular inteligente de sistema operativo android. Se desarrolló una aplicación (apk.) para este control. Fig.5

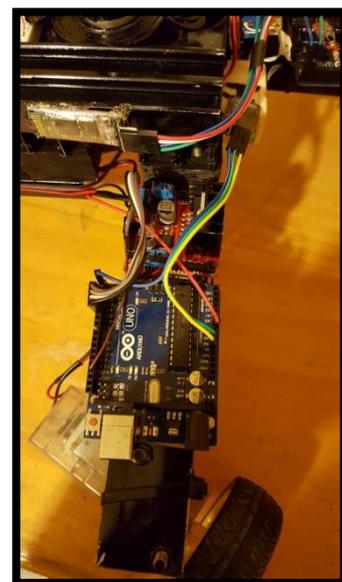


Figura 5. Vista del módulo Bluetooth

Tercera tarjeta, se tiene al Arduino Uno que controla los tres actuadores restantes cada uno de manera independiente y el encendido y apagado del taladro. Además, tiene el mando de manejo a un módulo bluetooth como periférico de entrada de datos; este módulo, se conecta a un celular inteligente de sistema operativo android. Se desarrolló una aplicación (apk.) para este control. Fig.6



Figura 6. Tarjeta Arduino Uno

Acotando, el sentido de giro de cada actuador tiene como mando a un módulo Puente-H de 4 Amperios. Fig.7; además, todo el sistema está impulsado por una fuente de 12 voltios DC a 15 amperios con alimentación de AC 220. Fig.8



Figura 7. Puente H de 4 Amperios.



Figura 8. Fuente de alimentación DC

Parte Programación

Para cada tarjeta Arduino se tiene una programación distinta.

Primera tarjeta:

```

acelerometro Arduino 1.6.5
Archivo Editar Programa Herramientas Ayuda

acelerometro
#include <Wire.h>
#include <ADXL345.h>
ADXL345 adxl; //variable adxl is an instance of the ADXL345 library
int x,y,z,a,b;
void setup(){
  pinMode(7, OUTPUT); // 1 logico para giro de Motor Horario
  pinMode(8, OUTPUT); // 1 logico para giro de Motor Anti horario
  pinMode(9, OUTPUT);
  Serial.begin(9600);
  adxl.powerOn();
}
void loop()
{
  a=15; // limites para que el eje este en posicion vertical
  b=15; // limites para que el eje este en posicion vertical
  adxl.readAccel(&x, &y, &z);
  if (x<-a) {
    digitalWrite(8,1); // Giro Motor Anti horario
  }
  else{
    digitalWrite(8,0);
  }
  if (x>b)
  {
    digitalWrite(7,1); // Giro Motor Horario
  }
  else{
    digitalWrite(7,0);
  }
  // lectura de valores; para un sensado correcto de posicion
  Serial.print("\n x: "); Serial.print(x); // raw data without offset
  Serial.print(" y: "); Serial.print(y); // raw data without offset
  Serial.print(" z: "); Serial.print(z); // raw data without offset
}

Guardado.

Global variables use 436 bytes (21%) of dynamic memory, leaving 1.612 bytes
for local variables. Maximum is 2.048 bytes.

1 Arduino Nano, ATmega328 on COM6
  
```

Segunda tarjeta:

```

movimiento_brazo_4G Arduino 1.6.5
Archivo Editar Programa Herramientas Ayuda

movimiento_brazo_4G
//bluetooth hc-05
int bandera2 = 0; // usamos un pin de salida al LED
int state = 0; // Variable lectura dato serial
int state2 = 0;
int bandera=0;
void setup() {
  pinMode(13, OUTPUT); //Declara pin de Salida
  pinMode(12, OUTPUT);
  pinMode(11, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(9, OUTPUT);
  pinMode(8, OUTPUT);
  pinMode(7, OUTPUT);
  digitalWrite(13, LOW); //Normalmente Apagado
  Serial.begin(9600);
}
void loop() {
  //***** Motor 1 *****
  if(Serial.available() > 0){
    state = Serial.read();
  }
  if (state == 'A' && bandera==0) {
    digitalWrite(13, HIGH);
    bandera = 1; state = 0;
  }
  if (state == 'A' && bandera==1) {
    digitalWrite(13, LOW);bandera = 0;state = 0;
  }
  if (state == 'B' && bandera==0) {
    digitalWrite(12, HIGH);
    bandera = 1; state = 0;
  }
  if (state == 'B' && bandera==1) {
    digitalWrite(12, LOW);bandera = 0;state = 0;
  }
  // ***** Motor 2 *****
  if (state == 'C' && bandera==0) {
    digitalWrite(11, HIGH);
    bandera = 1; state = 0;
  }
  if (state == 'C' && bandera==1) {
    digitalWrite(11, LOW);bandera = 0;state = 0;
  }
}

```

```

movimiento_brazo_4G Arduino 1.6.5
Archivo Editar Programa Herramientas Ayuda

movimiento_brazo_4G
digitalWrite(12, HIGH);
bandera = 1; state = 0;
if (state == 'B' && bandera==1) {
  digitalWrite(12, LOW);bandera = 0;state = 0;
}
// ***** Motor 2 *****
if (state == 'C' && bandera==0) {
  digitalWrite(11, HIGH);
  bandera = 1; state = 0;
}
if (state == 'C' && bandera==1) {
  digitalWrite(11, LOW);bandera = 0;state = 0;
}
if (state == 'D' && bandera==0) {
  digitalWrite(10, HIGH);
  bandera = 1; state = 0;
}
if (state == 'D' && bandera==1) {
  digitalWrite(10, LOW);bandera = 0;state = 0;
}
// ***** Motor 3 *****
if (state == 'E' && bandera==0) {
  digitalWrite(9, HIGH);
  bandera = 1; state = 0;
}
if (state == 'E' && bandera==1) {
  digitalWrite(9, LOW);bandera = 0;state = 0;
}
if (state == 'F' && bandera==0) {
  digitalWrite(8, HIGH);
  bandera = 1; state = 0;
}
if (state == 'F' && bandera==1) {
  digitalWrite(8, LOW);bandera = 0;state = 0;
}
//***** TALADRO *****
if(Serial.available() > 0){
  state= Serial.read();
}
if (state == 'I' && bandera2==0) {
  digitalWrite(7, HIGH);
  bandera2 = 1; state = 0;
}
if (state == 'I' && bandera2==1) {
  digitalWrite(7, LOW);bandera2 = 0;state = 0;
}
}

```

Ventana de la aplicación en sistema operativo Android, la cual se muestra en la Figura 9.



Figura 9. Ventana de aplicación en Android

Tercera tarjeta:

```

desplzamiento_superficie Arduino 1.6.5
Archivo Editar Programa Herramientas Ayuda

desplzamiento_superficie
char Carácter;
void setup() {
  Serial.begin(9600);
  pinMode(2, OUTPUT);
  pinMode(3, OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(5, OUTPUT);
  digitalWrite(2, LOW);
  digitalWrite(3, LOW);
  digitalWrite(4, LOW);
  digitalWrite(5, LOW);
}
void loop() {
  if (Serial.available()) {
    Carácter=Serial.read();
    if(Carácter=='A'){
      digitalWrite(2, HIGH);
      digitalWrite(3, LOW);
      digitalWrite(4, HIGH);
      digitalWrite(5, LOW); }
    else if (Carácter=='R') {
      digitalWrite(2, LOW);
      digitalWrite(3, HIGH);
      digitalWrite(4, LOW);
      digitalWrite(5, HIGH);}
    else if (Carácter=='D'){
      digitalWrite(2, LOW);
      digitalWrite(3, HIGH);
      digitalWrite(4, HIGH);
      digitalWrite(5, LOW); }
    else if (Carácter=='I') {
      digitalWrite(2, HIGH);
      digitalWrite(3, LOW);
      digitalWrite(4, LOW);
      digitalWrite(5, HIGH);}
    else if (Carácter=='N'){
      digitalWrite(2, LOW);
    }
  }
}

```

Compilado

Global variables use 183 bytes (8%) of dynamic memory, leaving 1.865 bytes for local variables. Maximum is 2.048 bytes.

36 Arduino Nano, ATmega328 on COM6

Ventana de la aplicación en sistema operativo Android. Fg.10

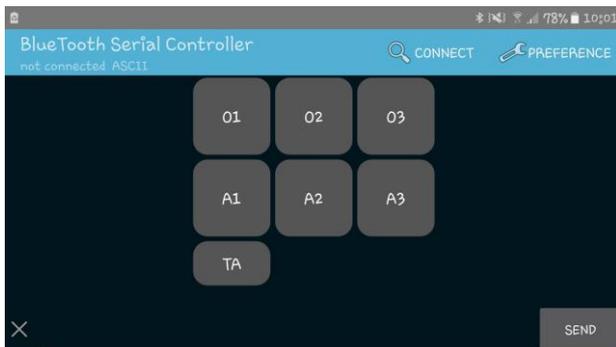


Figura 10. Venta de aplicación en Andorid

Resultados

Podemos apreciar que los resultados son positivos al probar el robot móvil porque es más fácil el traslado vía bluetooth para taladrar en cualquier tipo de superficie; ya que cualquier persona sin conocimientos en programación o electrónica puede usar este robot móvil vía bluetooth desde un Smartphone.

Conclusiones

1. Permite la eliminación total o parcial de la intervención del hombre en el momento de taladrar
2. Reduce los gastos de mano de obra directos, con la eliminación de estas el proceso de taladrado.
3. Aumenta la calidad de producción ya que los robot dirigidos a distancia son más fáciles de maniobrar

Observaciones

Se logró de una forma general entender cómo se controlan el robot móviles a través de arduino.

Es necesario el uso de las protecciones en todas las etapas ya que estas protecciones evitan la destrucción total de los equipos, así se evita pérdidas materiales que involucra a la economía de la empresa.

El presente proyecto será utilizado como base para la tesis para obtener el Título de Ingeniero Mecatrónico.

Referencias

1. Ramón Pallas, Sensores y Acondicionadores de señal edición 4ta editorial Alfaomega.
2. W. Boston, Sistema de Control Electrónico en la ingeniería mecánica y eléctrica, editorial Alfaomega.
3. Jose Ignacio Suarez, Sensores Actuadores microboticos,
4. Electrónica, Teoría de Circuitos y Dispositivos. Robert L. Boylestar Louis Nashelky Editorial México Pearson 10ma Edición
5. Dispositivos Electrónicos Thomas L. Floyd Editorial Preston México Education 8va Edición 2008.
6. Diseño Electrónico: Circuitos y sistemas CJ Savant S Roden L. Carpenetr USA Addison – Wesley Iberoamericana 2da Edición.

7. Principios de electrónica Manuel Ramos Álvarez Editorial México Red Tercer Milenio 1ra Edición 2012.
8. Electrónica Básica para ingenieros Gustavo Ruiz Robedro. España servicio de Repografía, 1ra Edición 2001.
9. Electrónica Alan R Hmbley México Editorial Prince Hall 2da Edición 2011
10. Circuitos y Dispositivos Microelectronicos Mark N Horenstein México Pretince Hall 2da Edición 1997.
11. Principios de electrónica Malvino, editorial McGraw Hill, 7ma Edición
12. Electrónica de Potencia Fundamentos Básicos, Chilet Salvador Seguí, Editorial AlfaOmega.
13. Electrónica de potencia, convertidores Ned Mohan, Tore M. Undeland, William P. Robbins, editorial McGraw Hill, 3ra Edición

Universidad Ricardo Palma.

Conocimientos en SolidWorks, AutoCAD.

Celular : +51942441288

Bibliografía

Anthony Carrión Hurtado



Estudiante de ingeniería de Mecatrónica del 9no ciclo, su área de interés es la automatización automotriz, domótica. Tiene conocimientos de importación y amante de las motos y velocidad.

Celular: +051984817547



Sánchez Cubas, Roger Javier

Estudiante de la carrera de Ingeniería Mecatrónica en la