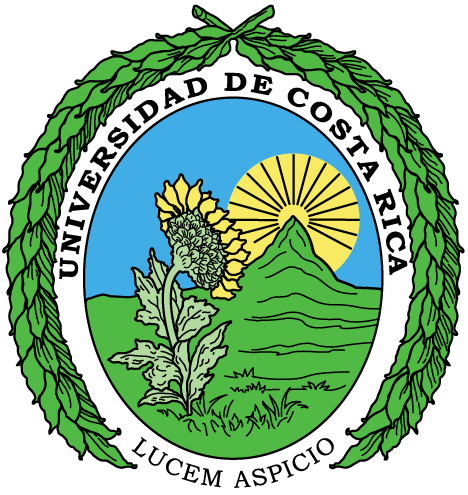
****

**Universidad de Costa Rica**

**Facultad de Ingeniería**

**Escuela de Computación e Informática**

**Laboratorio de Circuitos Digitales**

**CI1211**

**Grupo 002**

**Tema: Construccion de carro.**

**Profesor:**

**Sanders Pacheco Araya**

**Alumnos:**

**Fernando Pereira Leiva- B04751**

**Johan Durán Cerdas - B42319**

**Juan Diego Ramírez Carranza - B25391**

**I - 2015**

**Objetivo:**

Utilizar los conocimientos de Diseño Digital para construir el circuito controlador de un carro con tracción trasera y dirección, controlado por el puerto serial de un computador.

**Descripción:**

El proyecto consiste en diseñar un circuito digital y un programa de computador que permita controlar un carro de construido con piezas de LEGO técnico o piezas similares.

**Materiales utilizados para el desarrollo del proyecto:**

* Tabla protoboard
* Arduino UNO rev3
* Modulo bluetooth hc-06
* Puente H L293e
* 4 diodos 1N4004
* Unas resistencia de 10k ohm
* 3 ligas
* 1 gasa plástica grande y 10 pequeñas
* 30 puntas de cable (10 largas y 20 cortas)
* 1 metro de cable UTP cat-5
* Un carro a control remoto que ya no funcionaba. (cuenta con dos motores)
* Una lata, dos clip (para sensores)
* Dos baterías de celular Nokia 2730.
* Silicona
* Estaño
* Cautín
* 1 jumper hembra cuatro entradas para el bluetooth

**Introducción, plataformas utilizadas:**

” La electrónica moderna usa [electrónica digital](https://es.wikipedia.org/wiki/Electr%C3%B3nica_digital) para realizar perfeccionamientos en la tecnología, muchas veces nos vemos frente a éstos sin darnos cuenta, el llamado efecto "Caja Negra".

En el circuito lógico digital existe transmisión de información binaria entre sus circuitos. A primera instancia esto nos parece relativamente simple, pero los circuitos electrónicos son bastante complejos ya que su estructura está compuesta por un número muy grande de circuitos simples, donde todos deben funcionar de la manera correcta, para lograr el resultado esperado y no obtener una información errónea.”(Wikipedia)

Arduino: Arduino fue creado en el año 2005 por el entonces estudiante Massimo Banzi del instituto IVRAE, en un inicio surgió solamente como la necesidad de brindarle a sus estudiantes de computación y electrónica un medio accesible para aprender sobre microcontroladores dado que en esa época era muy caro obtener un dispositivo similar agregando de que no ofrecían un soporte adecuado. Nunca pensó en lo que llegaría a convertirse, el más importante circuito en los llamados do it yourself. En su inicio la placa tenía el precio de 1 euro y los mismos estudiantes los vendían dentro de la institución para el beneficio de la misma.

App inventor: app inventor es una aplicación web de código abierto propiedad de google pero actualmente mantenido por el instituto tecnológico de Massachusetts, fue creada para el diseño de aplicaciones para dispositivos Android de forma sencillo ya que solamente se utilizan bloques para realizar la construcción de la app, sin embargo provee una herramienta muy útil si se desea crear algo rápido y funcional. Esta plataforma a pesar de su simpleza para programar cuenta con herramientas muy modernas como lo son sensores lectores de códigos de barras, ubicación, proximidad y muchas más herramientas con lo cual podríamos realizar cosas muy interesantes.

**Desarrollo del proyecto**

Lo primero que haremos será explicar todos los procedimientos que utilizamos para concluir este proyecto durante este punto iremos nombrando el número de la figura donde se encuentra el diagrama o calculo respectivo de algún valor en el circuito. Algunos de los diagramas mostrados fueron realizados en Fritzing una herramienta de código abierto.

**Procedimiento:**

Antes aún de tocar cualquier componente lo que hicimos fue darnos la tarea de investigar qué era lo que íbamos a necesitar, durante el proceso visitamos muchas páginas que nos brindaron información muy valiosa para empezar a comprender el funcionamiento de muchos circuitos controlados por arduino.

No agrego dichas páginas en la bibliografía porque en este documento no viene incluido información tomada de ellos sin embargo nos sirvió de guía para lograr el objetivo por lo tanto agregamos luego de la bibliografía los links importantes que nos ayudaron y podrían ayudar otras personas.

Excluyendo los 4 diodos y la resistencia de 10k ohm todos los otros materiales mencionados al inicio fueron con los que iniciamos. Esto dos materiales que excluimos de la lista es por el hecho de que en un inicio decidimos comprar el puente H l293D que no necesita diodos de protección sin embargo a la hora de comprar nos dieron el puente erróneo. En el caso de la resistencia fue porque nos tomó tiempo encontrar la forma de manipular los sensores pero esto lo explicaremos más adelante.

Entrando más de lleno al circuito en sí, utilizamos el arduino UNO rev3 por motivo de que en la red abundan muchos tutoriales referidos a arduino y en su mayoría el arduino Uno es el más utilizado (ver figura 1). En cuanto a los pines el pin 0 y 1 son los pines que van conectados con el modulo bluetooth. Transmisión con recepción del bluetooth y recepción con transmisión. El modulo bluetooth debe ir conectado el pin 1 a tierra y el pin 2 a 3.3V entregados por el respectivo pin del arduino. Pines 3,4 son transmisión y recepción respectivamente. El pin Vin es por donde damos el voltaje de entrada al arduino 6-20 V ademas podemos utilizar el puerto serial o el conector a fuente de voltaje ademas, tenemos tres puntos para conexión a tierra. En la figura 2 podemos ver las conexiones básicas iniciales del arduino con la batería y el modulo bluetooth. En la figura 3 podemos ver el diagrama del puente H que utilizamos el mismo al controlar el motor en ambas direcciones solo tiene capacidad para dos motores, en el caso de este puente H el mismo utiliza dos diodos por motor para proteger el resto del circuito ya que el puente H l293e no cuente con diodo de protección y por lo tanto sabiendo que los motores también pueden generar corriente si llegara a generarla estos diodos evitarían que causen daño. Estos diodos están conectados entre la salida del puente H y la entrada hacia los motores y la conexión que observamos evita que entre voltaje proveniente de los 5 voltios del arduino pero a la misma vez evita que si se genera un voltaje este tenga comunicación con tierra. El puente H l293e es un integrado de 20 pines, dos de sus pines los sense nos fueron inútiles ya que desde un inicio no los necesitábamos los mismos fueron conectados a tierra como la hoja de especificaciones lo decía. En la figura 4 mostramos el integrado L293e montado completamente. Los motores están siendo alimentados por 7.4 voltios ya que utilizamos dos baterías de celular que nos entregaban 3.7 V cada uno con 1020 mAh, para esto realizamos una conexión en serie la cual podemos ver en la figura 5.

Las conexiones de las luces son muy sencillas, en el carro tenemos 8 luces 6 de ellas se encuentran en gráficamente en la figura 6. En cuanto a las resistencias utilizamos resistencias de 550 ohm (asociación en serie de una de 220 con una de 330 ohm) para todos los leds de direccionales ya que dados los 5 voltios que entrega el arduino y los aproximadamente 0.04 A con una resistencia de 550 ohm evitamos que se nos queme tanto el led y que nos pueda llegar a causar algún daño en el arduino. Las luces frontales no tienen relación alguna con el arduino ya que nuestra idea es colocar una LDR y que nuestro carro solo encienda las luces frontales durante la noche, las mismas cuentan con una resistencia de 330 ohm y están conectadas directamente a los 5 V entregados por el arduino, al ser estas leds blancos consumen más que los rojos de las direccionales por esta razón es que utilizamos resistencia más pequeña, el voltaje de salida va a ser más pequeño y por ende con una resistencia más pequeña logramos hacer la reducción de corriente. Los dos diodos leds resultantes están conectados a los pines 12 y 18 del puente ya que por aquí se recibe el voltaje proveniente de los pines del arduino cuando se quiere ir hacia adelante o hacia atrás, los mismos tienen una resistencia de 440 ohm (dos de 220 ohm en serie).

Las resistencias utilizadas puede que no sean las perfectas pero utilizamos las que teníamos a disposición en ese momento.

Ahora proseguimos con los sensores, para los sensores utilizamos al igual que para casi todo el carro materiales reciclados, de una lata de cerveza lijada ya que viene con una cubierta que evita la conducción y con dos clip, 1 resistencia de 10k ohm. En la figura 7 y 8 se muestra la conexión realizada para los sensores y el cálculo de la resistencia. La resistencia al estar en serie solamente la utilizamos para reducir la corriente lo más posible y evitar quemar el arduino. Los dos sensores están interconectados por lo tanto el carro solamente detecta una colisión pero no la ubicación de la misma.

En cuanto a la circuitería creemos que ya hemos mencionado todos los circuitos utilizados, antes de finalizar esta parte queremos recalcar el hecho de que utilizamos estas baterías y en serie por motivo de que el carro es muy pesado y una batería de 9 V no lo movía como se debe. Sin embargo esta asociación que hicimos por ahora tiene un problema y es que entrega 1020 mAh y el puente L293e está diseñado para 600 por lo tanto necesitamos conectar dos resistencias de 1 ohm en serie para reducir este valor a la mitad. A pesar de que el voltaje y corriente que se consume depende de que tanto solicite el motor, al nosotros utilizar un carro reciclado no sabemos esos valores de los motores.

En la parte fisica la única modificación que hicimos fue el de recortar una pieza en los engranajes de la direccion esto para lograr un mejor viraje.

**Código:**

**Código aplicación:**

Para la aplicación del carro utilizamos APP inventor un ambiente de desarrollo del instituto tecnológico de Massachusetts, con este medio obtuvimos una forma sencilla de construir nuestra APP ya que su programación es por bloques ademas con solo un código QR lo escaneamos desde nuestros móviles y obtenemos la aplicación.

En esta parte redactaremos sobre la teoría detrás de la aplicación y el los diagramas con nombre de “diagrama código UCR MosterCar” se encuentran todos los bloques utilizados para este diseño.

En general el código es muy sencillo, al presionar un botón la aplicación envía mediante el bluetooth un carácter y con este carácter el arduino decide que acción tomar. Lo cual se encuentra dentro de la programación del arduino que describiremos más adelante.

Lo más “complejo” es el hecho de que usamos un presionar y soltar para los botones derecha, izquierda. En el momento que se presiona realiza una acción y al momento de soltar otra. Esto lo hicimos para lograr la mayor similitud a los carros a control remoto analógicos comúnmente utilizados.

Estas son las teclas y su respectivo carácter que se envía:

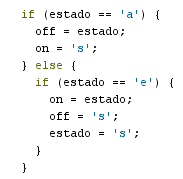
|  |  |
| --- | --- |
| Función | Carácter |
| Adelante | G |
| Izquierda presionado | I |
| Izquierda suelto | M |
| Detenerse | S |
| Derecha presionado | N |
| Derecha suelto | D |
| Atrás | R |
| Encender | E |
| Apagar | A |
| Velocidad 150 | X |
| Velocidad 250 | Y |

Además de esto la APP está programada para cada 10 ms estar comprobando si el bluetooth no tiene algo de entrada, dado que configuramos que cuando el carro colisione ademas de apagarse envíe una “q” a la aplicación para que cambie de color el fondo y nos muestre la palabra colisión. Para esto fue necesario un temporizador para estar realizando ese cheque continuamente. Al encender el carro todo vuelve a la normalidad.

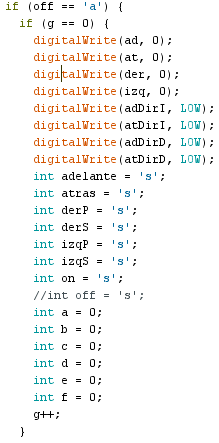
**Código arduino:**

En cuanto al código del arduino el mismo es de gran longitud dado que se tienen muchos estados distintos y para lograr manejar varios a la vez utilizamos muchas variables y con ellas evitar que entre a realice acciones que no debe realizar en el momento. Dado que son muchos casos vamos a explicar uno y con los demás pasa algo similar.

Por ejemplo: lo primero que se hace es hacer una selección entre los dos estados posibles y donde se habilita uno se deshabilita el otro.



Donde “s” es stop = detenerse. Luego si tenemos que está apagado en queremos que solo se apague una vez y no este repitiendo código es por eso que utilizamos una variable.



Y como vemos al salir la aumentamos esto para que no vuelta a ingresar ni aquí ni a encendido ya que encendido se encuentra en ‘s’. al momento que entre el carácter ‘e’ encendido se hará la misma selección del inicio pero con la diferencia que al salir del código de encendido este colocará g = 0; habilitando apagado por si en algún momento se volviera a presionar. Para encendido no se utiliza variable ya que si necesitamos que este en constante ejecución.

No agregamos el código de encendido en esta parte porque es muy extenso, serian todas las acciones que realiza el carro normalmente. A continuación el código completo donde se encuentran todos los casos.

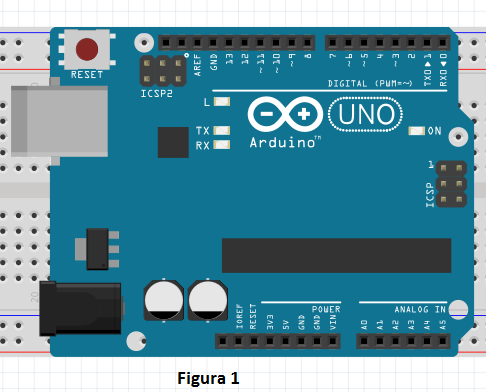
Entre las funcionalidades del carro tenemos:

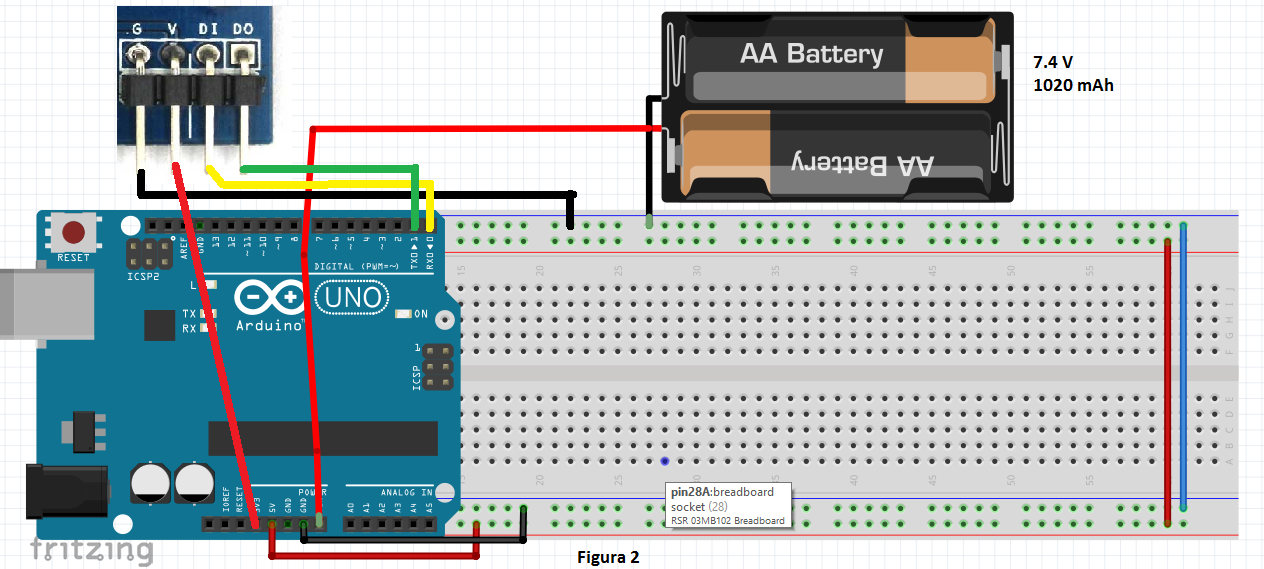
* Al chocar inmediatamente se apaga y notifica a la aplicación.
* Posee dos velocidades distinta, solo aplica para la tracción.
* Se puede encender y apagar el carro sin tener que desconectarlo.
* Luces de parque.
* Luces direccionales automáticas.
* La luz de reversa es el led verde que se encuentra sobre el carro.
* Luces frontales.

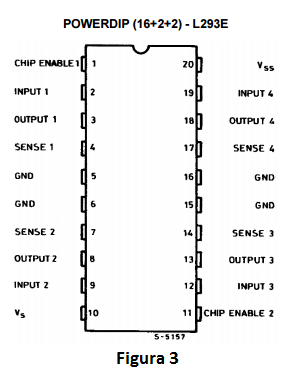
|  |  |
| --- | --- |
| // Carro Inalámbrico Bluetooth  int choque = 7;  int pulsador = 0;  int ad = 5;  int at = 6;  int der = 9;  int izq = 10;  int adDirD = 13;  int adDirI = 12;  int atDirD = 4;  int atDirI = 2;  int vel = 255; // Velocidad de los motores (0-255)  int estado = 'a'; // inicia en off  int adelante = 's';  int atras = 's';  int derP = 's';  int derS = 's';  int izqP = 's';  int izqS = 's';  int on = 's';  int off = 's';  int a = 0;  int b = 0;  int c = 0;  int d = 0;  int e = 0;  int f = 0;//ad  int g = 0;//atr  void setup() {  Serial.begin(9600); // inicia el puerto serial para comunicacion con el Bluetooth  pinMode(ad, OUTPUT);  pinMode(at, OUTPUT);  pinMode(der, OUTPUT);  pinMode(izq, OUTPUT);  pinMode(adDirD, OUTPUT);  pinMode(adDirI, OUTPUT);  pinMode(atDirD, OUTPUT);  pinMode(atDirI, OUTPUT);  pinMode(choque, INPUT);  }  void loop() {  if (Serial.available() > 0) { // lee el bluetooth y almacena en estado  estado = Serial.read();  }  pulsador = digitalRead(choque);  if (pulsador == HIGH) {  Serial.println("q");  estado = 'a';  }  if (estado == 'a') {  off = estado;  on = 's';  } else {  if (estado == 'e') {  on = estado;  off = 's';  estado = 's';  }  }  if (off == 'a') {  if (g == 0) {  digitalWrite(ad, 0);  digitalWrite(at, 0);  digitalWrite(der, 0);  digitalWrite(izq, 0);  digitalWrite(adDirI, LOW);  digitalWrite(atDirI, LOW);  digitalWrite(adDirD, LOW);  digitalWrite(atDirD, LOW);  int adelante = 's';  int atras = 's';  int derP = 's';  int derS = 's';  int izqP = 's';  int izqS = 's';  int on = 's';  //int off = 's';  int a = 0;  int b = 0;  int c = 0;  int d = 0;  int e = 0;  int f = 0;  g++;  }  } else {  if (on == 'e') {    //g=ad;s=stop; r=reversa;e=auto;a=autoOFF; i=izqPres;m=izqsuelto;m=derePresionado;d=derecha suelto.  if (estado == 'g') {  adelante = estado;  atras = 's';  } else {  if (estado == 'r') {  atras = estado;  adelante = 's';  }  }  if (estado == 's') {  digitalWrite(ad, 0);  digitalWrite(at, 0);  adelante = 's';  atras = 's';  f = 0;  e = 0;  //--------------------------------  digitalWrite(adDirI, HIGH);  digitalWrite(atDirI, HIGH);  digitalWrite(adDirD, HIGH);  digitalWrite(atDirD, HIGH);  delay(100);  digitalWrite(adDirI, LOW);  digitalWrite(atDirI, LOW);  digitalWrite(adDirD, LOW);  digitalWrite(atDirD, LOW);  delay(100);  digitalWrite(adDirI, HIGH);  digitalWrite(atDirI, HIGH);  digitalWrite(adDirD, HIGH);  digitalWrite(atDirD, HIGH);  delay(100);  digitalWrite(adDirI, LOW);  digitalWrite(atDirI, LOW);  digitalWrite(adDirD, LOW);  digitalWrite(atDirD, LOW);  delay(100);  } | if (adelante == 'g') {  if (e == 0) {  giro(adelante);  e++;  }  } else {  if (atras == 'r') {  if (f == 0) {  giro(atras);  f++;  }  }  }  if (estado == 'i') {  izqP = estado;  } else {  if (estado == 'm') {  izqS = estado;  } else {  if (estado == 'n') {  derP = estado;  } else {  if (estado == 'd') {  derS = estado;  }  }  }  }    if (a == 0) {  if (izqP == 'i') {  digitalWrite(adDirI, HIGH);  digitalWrite(atDirI, HIGH);  delay(100);  digitalWrite(adDirI, LOW);  digitalWrite(atDirI, LOW);  delay(100);  digitalWrite(adDirI, HIGH);  digitalWrite(atDirI, HIGH);  delay(100);  digitalWrite(adDirI, LOW);  digitalWrite(atDirI, LOW);  digitalWrite(der, 0);  digitalWrite(izq, vel);  delay(170);  izqS = 0;  a++;  b = 0;  }  } else {  if (b == 0) {  if (izqS == 'm') {  digitalWrite(izq, 0);  digitalWrite(der, vel);  delay(30);  b++;  a = 0;  izqP = 0;  }  }  }  if (c == 0) {  if (derP == 'n') {  digitalWrite(adDirD, HIGH);  digitalWrite(atDirD, HIGH);  delay(100);  digitalWrite(adDirD, LOW);  digitalWrite(atDirD, LOW);  delay(100);  digitalWrite(adDirD, HIGH);  digitalWrite(atDirD, HIGH);  delay(100);  digitalWrite(adDirD, LOW);  digitalWrite(atDirD, LOW);  digitalWrite(izq, 0);  digitalWrite(der, vel);  delay(170);  derS = 0;  c++;  d = 0;  }  } else {  if (d == 0) {  if (derS == 'd') {  digitalWrite(der, 0);  digitalWrite(izq, vel);  delay(30);  d++;  c = 0;  derP = 0;  }  }  }  digitalWrite(der, 0);  digitalWrite(izq, 0);  g = 0;  off = 's';  }  }  }  void giro(int dir) {  if (dir == 'g') {  digitalWrite(at, 0);  digitalWrite(ad, vel);  f = 0;  } else {  if (dir == 'r') {  digitalWrite(ad, 0);  digitalWrite(at, vel);  e = 0;  }  }  } |

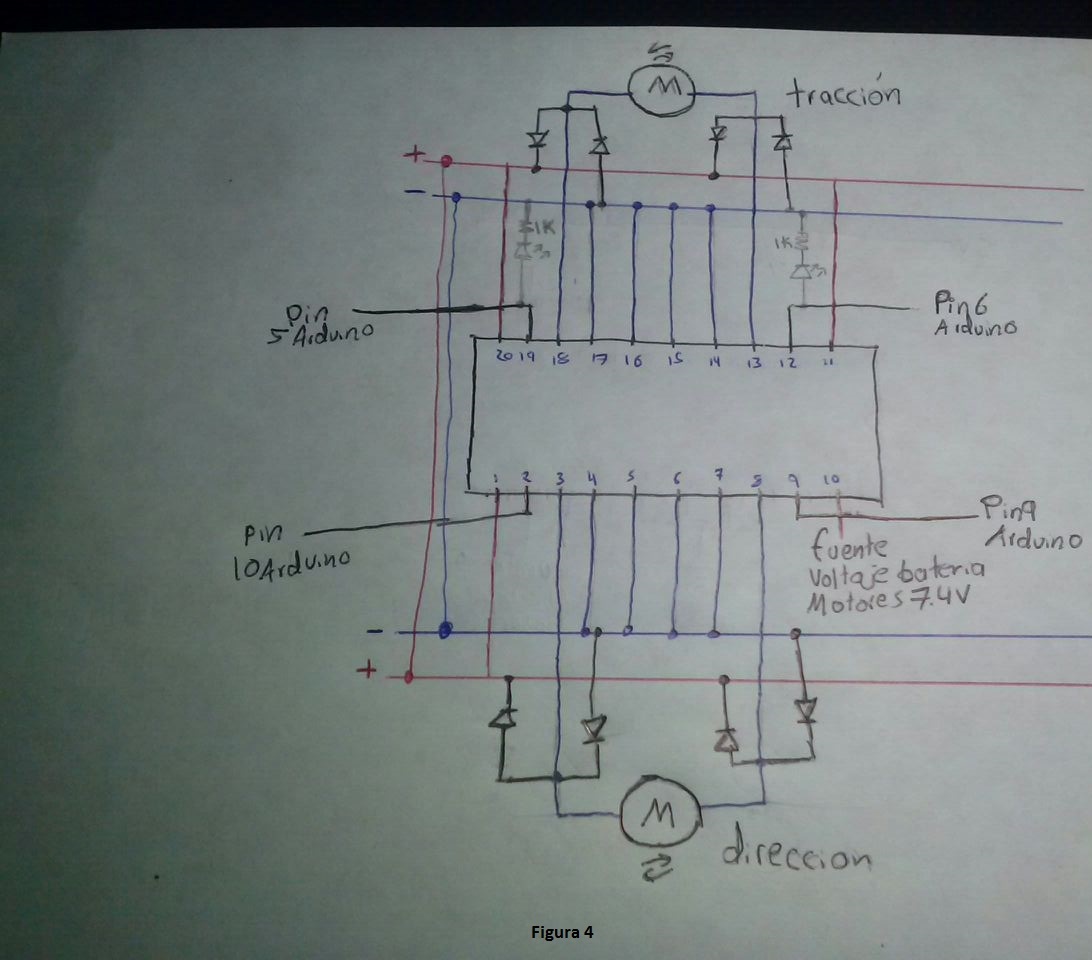
En términos generales al unir hardware con software lo que nos costó más fue encontrar el equilibrio perfecto para la direccion ya que al no utilizar servomotor se debe tener un valor muy preciso para que el carro no quede desalineado. Al final nos ayudamos con una liga para mantener la posición siempre alineada.

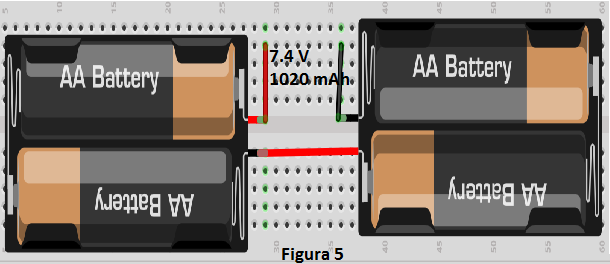
**Diagrama:**

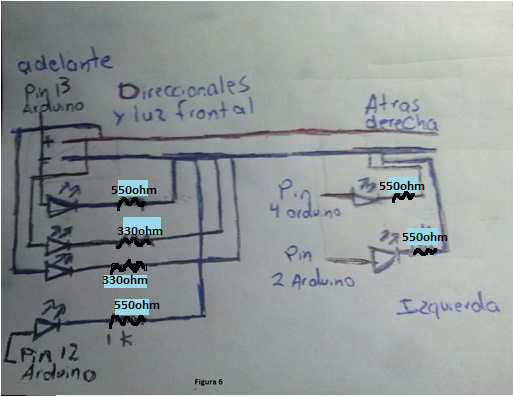


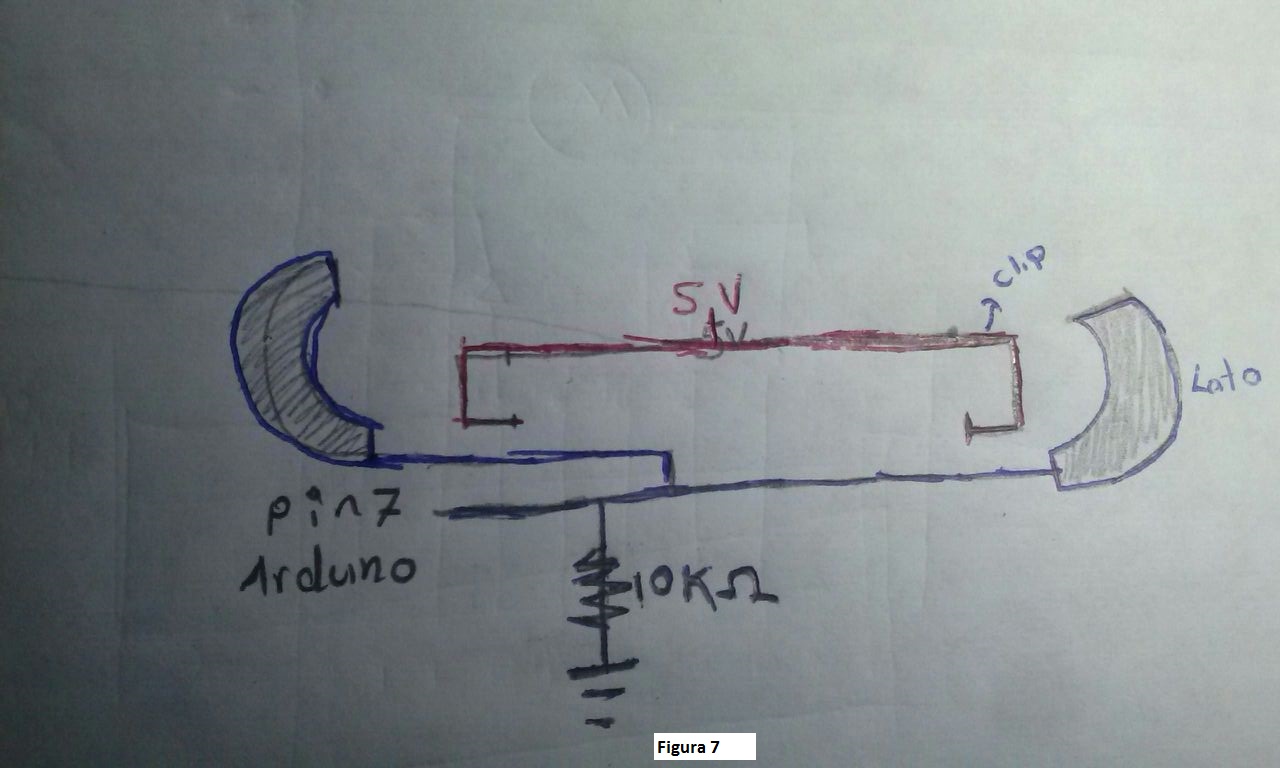


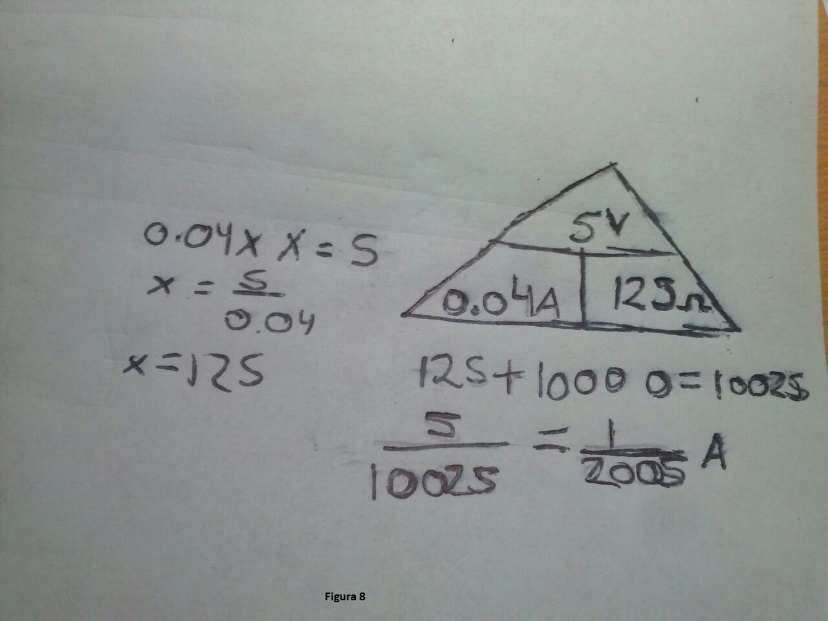




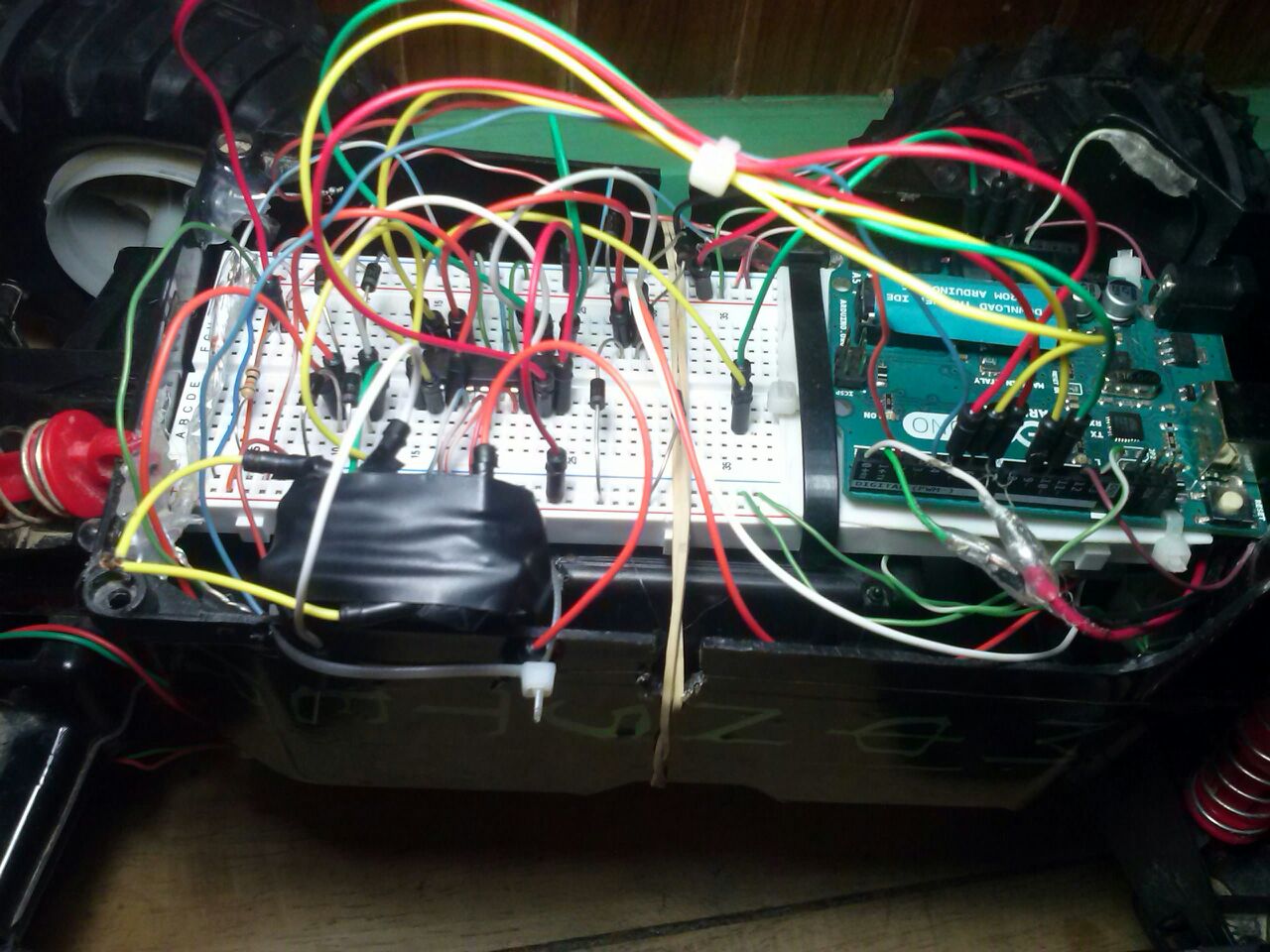




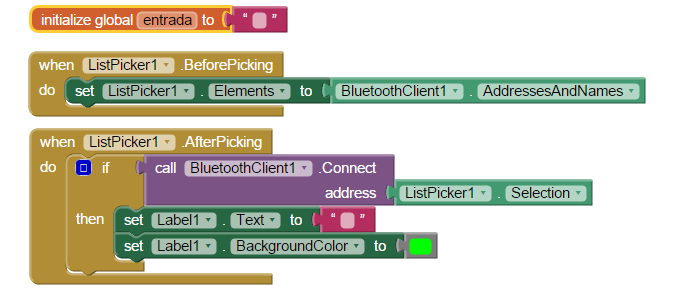


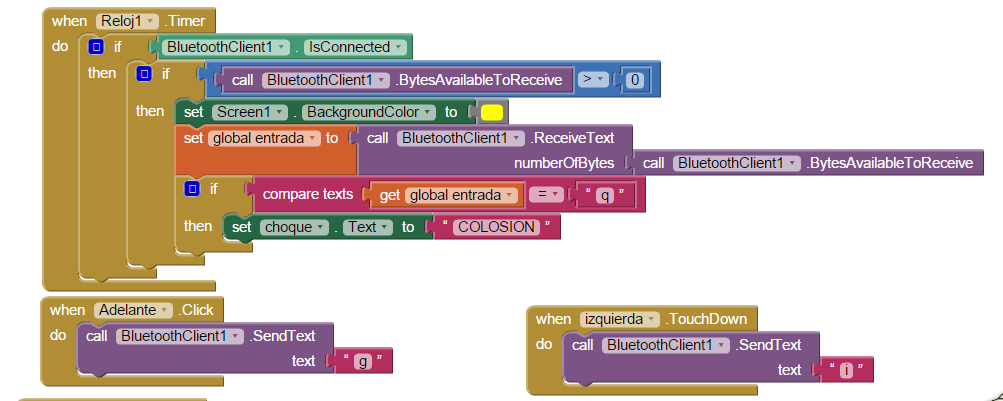


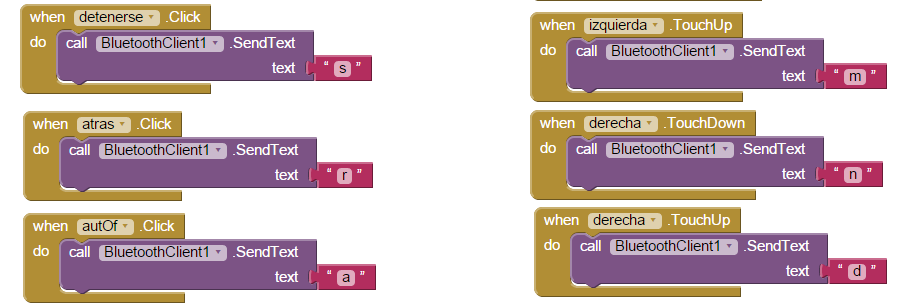


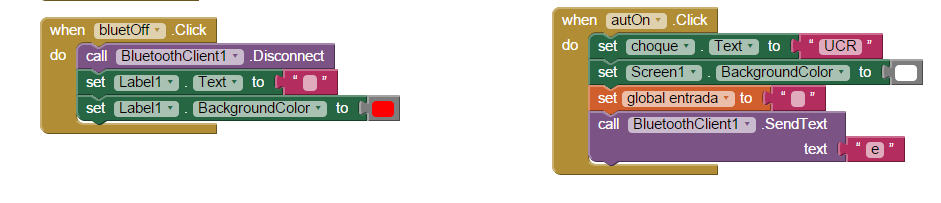
****

**Diagramas “UCR MonsterCar”:**

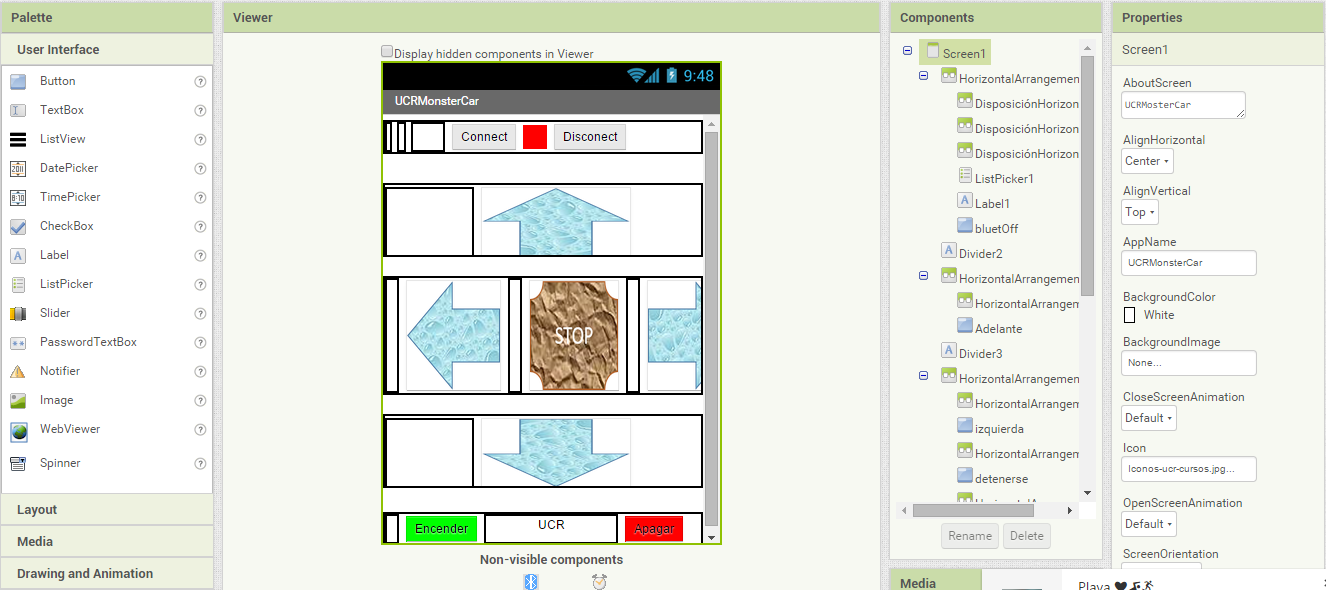












**Anomalías:**

En cuanto a las anomalías y problemas presentes no se nos presentó ninguno, el tema de la resistencia de 2ohm que nos está faltando es aún necesaria pero es lo único que nos puede traer problemas, en cuanto a todo lo demás en proyecto salió según lo esperado.

**Conclusiones:**

**Fernando Pereira Leiva - B04751:**

Durante este proyecto fue notable como fuimos acumulando conocimiento conforme íbamos revisando cada página web y empezábamos a relacionar algunos temas vistos en clase con la necesidad de aplicarlos en algún punto del circuito del carro. Es interesante cuando ramas distintas se mezclas y conforman cosas nuevas.

A pesar de que nos tomó mucho tiempo conformar todas las partes y tenerlo listo, esto nos ayudó a mejorar nuestras habilidades para trabajar en grupo y mejorar la forma de compartir conocimiento.

**Johan Durán Cerdas - B42319:**

Desde ya hace mucho tiempo, para ser claro antes de interesarme por los sistemas computacionales ya tenía un interés por encaminarme por el camino de la electrónica y la robótica sin embargo eso no se dio pero por dicha al entrar en este carrera nos topamos con el curso de diseños de circuitos digitales, que sin duda alguna nos da las bases para poder crear cosas impresionantes, ya que cuando mezclamos un área como los sistemas con otra área como la electrónica podemos crear aparatos mucho más inteligentes a uno que no mezcle ramas como las dichas.

Con la realización de este carro el conocimiento que me quedó más claro es la capacidad que tenemos para crear cosas solos o en grupo, y de las grandísimas ventajas que tenemos los ciudadanos del siglo XXI con el fácil acceso al internet y con la infinidad de información que se encuentra ahí presente. Como ya lo dije anteriormente este curso nos da las bases pero las podemos completar dándole uso a la inmensa base de datos que es internet.

Otro aspecto que me pareció sumamente interesante es el hecho de que si quitamos el arduino y el puente, podríamos decir que todo lo demás es material reciclado y me parece increíble que con aparatos que ya íbamos a desechar podamos crear cosas nuevas, en nuestro caso no tiene utilidad más que distracción pero que si vemos desde otro punto podría llevar a cosas de gran utilidad para nosotros.

**Juan Diego Ramírez Carranza - B25391:**

Con este proyecto y este curso en general me enseño que como futuros profesionales en computación e informática nos es imprescindible tener conocimiento en muchos ámbitos como en este caso en el diseño de circuitos y electrónica en general.

A pesar de que puede ser que nunca más tenga relación alguna con la electrónica, si llegara a pasar al menos tendría la ventaja de saber sobre lo que me hablan y esto me haría multidisciplinario y con la capacidad de compartir conversaciones laborales que se generan en ambientes donde se mezclan varias ramas distintas de la ingeniería actual.

**Bibliografía:**

<http://www.st.com/web/en/resource/technical/document/datasheet/CD00000058.pdf>

<https://es.wikipedia.org/wiki/Circuito_digital>

**Fuentes de ayuda:**

<http://www.iesromerovargas.es/android/>

<http://42bots.com/tutorials/hc-06-bluetooth-module-datasheet-and-configuration-with-arduino/>

<https://www.youtube.com/watch?v=ZLjlmHW0QjU&index=3&list=PLnwu2s7SIakT3OW0kOQHMuyJYm9aorwc4>

<https://www.youtube.com/watch?v=MiWLzbpfqr0&list=PLnwu2s7SIakT3OW0kOQHMuyJYm9aorwc4&index=2>

<https://www.youtube.com/watch?v=Ei43v93NEMQ&list=PLnwu2s7SIakT3OW0kOQHMuyJYm9aorwc4&index=5>

<https://www.youtube.com/watch?v=EIMjJtHkTUc>

<https://www.youtube.com/watch?v=ZlecE1CgIho>

<https://www.youtube.com/watch?v=f87lnDg5P30>

<https://www.youtube.com/watch?v=eexdP215ocs>

<https://www.youtube.com/watch?v=yMSgkotNQYo>