**INSTITUTO TECNOLÓGICO DE COSTA RICA.**

**SEGUNDO PROYECTO PROGRAMADO:**

**FÓRMULA E CE TEC.**

**CURSO:**

**TALLER DE PROGRAMACIÓN.**

**PROFESOR:**

**Msc. PEDRO GUITIÉRREZ.**

**ALUMNOS:**

**MARCO PICADO M. 2018310184.**

**SIMÓN FALLAS V. 2019324313.**

**BRANDON MUÑOZ C. 2018087204.**

**FECHA DE ENTREGA:**

**JUEVES 30 DE MAYO**

**PRIMER SEMESTRE 2019**

Contenido

[INTRODUCCION: 2](#_Toc9866502)

[RECOMENDACIONES: 3](#_Toc9866503)

[Desarrollo: 3](#_Toc9866504)

[Aprovechamiento: 4](#_Toc9866505)

[ANÁLSIS DE RESULTADOS: 5](#_Toc9866506)

[Diagrama de módulos: 5](#_Toc9866507)

[Plan de pruebas: 6](#_Toc9866508)

[Reglas del grupo: 6](#_Toc9866509)

[Actividades: 7](#_Toc9866510)

[Roles: 8](#_Toc9866511)

[Bibliografía consultada: 8](#_Toc9866512)

[Fechas entregables: 9](#_Toc9866513)

[ANEXOS: 9](#_Toc9866514)

# INTRODUCCION:

En el siguiente proyecto se espera implementar la construcción de un automóvil autónomo alimentado con baterías y capaz de recibir señales desde Python mediante código Arduino.

Como se sabe ya, a manera de antecedente, se tiene que los autos de fórmula 1 han ido evolucionando rápidamente de la mano del avance científico contemporáneo. Siendo el área de la informática uno de los grandes contribuyentes que impulsaron el desarrollo en estas competiciones.

Dentro de los avances dados por la informática se tiene el de la recopilación automática de datos del vehículo. Esto desde una perspectiva general se compone de sensores que recogen mediciones y las envían a los servidores de cada equipo para ser analizada por los ingenieros en tiempo real.

Naturalmente, este procedimiento permite que se lleve un mejor control del auto, un seguimiento más cercano e incluso maniobrar los vehículos desde el centro de operaciones, una herramienta muy útil en caso de accidente. Además, este gran paso en las competiciones contemporáneas permite aparte de lo anterior, obtener un avance enorme en el tema de la prevención de accidentes en general y la seguridad de las y los usuarios.

Sea en una competición o en la carretera, hoy en día, los automóviles cuentan con sistemas de computación capaces de activar o desactivar bolsas de aire mediante sensores de choque, medir la temperatura del motor, monitorear sistemas de frenos ABS, apagar el motor en casos específicos y un número elevado de utilidades que en una primera instancia se implementaron en competiciones y luego en el mercado mundial de vehículos.

Dado lo anterior es posible hacerse a la idea de que un automóvil entonces lleva más que mecánica, mucha computación y por ende ingeniería en computadores. Entonces, con el presente proyecto se planea tener una inmersión dentro del sector de computadores relacionado a los automóviles.

Lo anterior se logrará mediante la modificación de un vehículo a control remoto de juguete, al que se le hará una importante modificación, aprovechando algunos de sus componentes como los motores, la carcasa, socket de baterías entre otros.

Así, con todo preparado se construirá un nuevo hardware compuesto por un circuito que será comandado por un NodeMCU para la comunicación usuario máquina, un circuito integrado L298N que administrará el envío de energía a los motores y un registro de corrimiento HD74LS164P, el cual nos ayudará a controlar las luces del auto.

Para llegar a este punto primero se someterá cada componente a pruebas en protoboard que servirán para implementar su funcionamiento por medio de la programación Arduino. Una vez que todo esté probado se procederá a soldarlo en la tarjeta pre perforada, colocarla en el auto, colocar las luces y conectar las baterías y motores. Todo esto se ha descrito de una manera rápida; para un mejor seguimiento e información acerca de este proyecto se recomienda consultar la bitácora de trabajo, en la que se brinda información de manera detallada acerca de todo lo que ha sido el desarrollo de este proyecto.

# RECOMENDACIONES:

Aquí se presentan algunas recomendaciones a seguir para el mayor aprovechamiento del vehículo y las pruebas, además de algunas otras para el desarrollo del vehículo:

## Desarrollo:

1. Se recomienda el uso de baterías alcalinas por cuestión de consumo y durabilidad.
2. Se recomienda la adquisición de un auto que tenga suficiente espacio para trabajar cómodamente.
3. En caso de necesitar hacer cortes se recomienda el uso de un bisturí.
4. Para el momento de soldar se recomienda comprar una cera para cautín, ya que con eso se evita que el estaño se pegue a la punta y eso nos puede dañar algún componente.
5. Se recomienda a la hora de soldar tener puesta una máscara que nos aísle de los efectos dañinos del humo de soldadura o plástico, también uso de guantes para evitar quemaduras y se incita a que se adopte como práctica para resguardo de la salud.
6. Para el plan de pruebas es ideal tener dos o más protoboards pegadas ya que así se puede proceder con más comodidad (ver anexo 2).
7. Para pegar y aislar componentes como LED’s se puede optar por silicón caliente, ya que no afecta los componentes, y es barato.
8. Se recomienda también, a la medida de los recursos del desarrollador, adquirir un estaño de buena calidad, ya que al final de cuentas es más cómodo para trabajar, y se evita el daño de algún componente.
9. Los cables a utilizar, han de ser de varios hilos, preferiblemente de cobre para que no se quiebren.
10. Se recomienda utilizar algún IDLE de terceros para una mayor comodidad a la hora de programar como el autocompletado de palabras entre otras funciones.

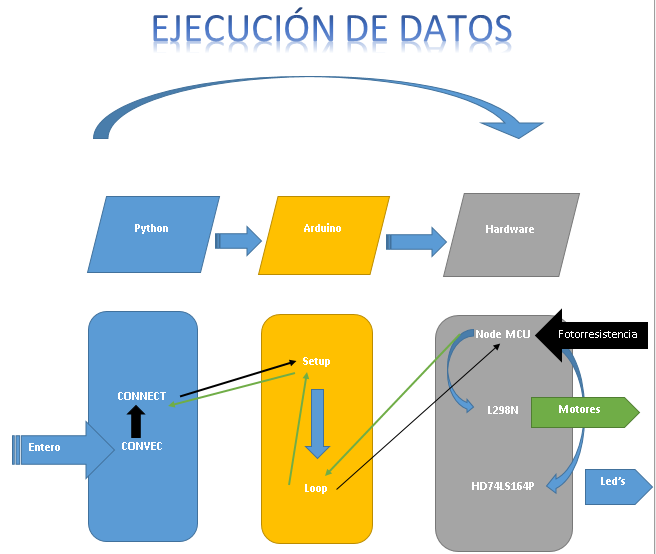
## Aprovechamiento:

1. Se recomienda mantener las baterías desconectadas hasta el momento de las pruebas.
2. Se recomienda colocar el interruptor de auto en un lugar que no interfiera con nada y que sea de fácil acceso.
3. Es necesario consultar el archivo Readme.txt que se encuentra en el paquete enviado junto con esta documentación.
4. Se recomienda almacenar el vehículo en un lugar que no tenga inconvenientes con la humedad.
5. Se recomienda manejar el vehículo con cuidado ya que algunos componentes se encuentran expuestos y podrían dañarse con algún golpe.

# ANÁLSIS DE RESULTADOS:

En esta parte se puede leer aspectos generales del grupo, su manera de trabajar, reglas del grupo, actividades, diagrama de módulos y plan de pruebas.

## Diagrama de módulos:



## Plan de pruebas:

El plan de pruebas consiste en ir probando poco a poco cada uno de los componentes del circuito con su respectivo código de Arduino. Cabe destacar que esto se hará con protoboards (ver recomendación 6) y que las órdenes serán dadas desde Python para ir corriendo con todo el proyecto en armonía.

Para comenzar se probó el nodeMCU con su programación que permite conectarlo con Python a través de Arduino y así se corrobora el análisis de envió y recibo de señales digitales y analógicas. Un string de un número servirá para accionar nuestros componentes y manipularlos a conveniencia.

Posterior a esto se probaron los LED’s mediante el registro de corrimiento HD74LS164P, donde diferentes valores de un byte encenderán ciertas luces, se tiene que un string de ceros enciende todas las luces y un string de unos las apaga. Entre estos dos hay 254 posibilidades más de configurar las luces por lo que se opta por programar un convertidor de decimal a binario y así facilitar la introducción de parámetros.

Ya con esto aprobado se procede a realizar las pruebas del motor de tracción, el cual será manipulado por un circuito integrado L298N que suministrará de energía a los motores con el fin de darles la potencia deseada para los diferentes escenarios a enfrentar y, además, de una vez se prueba la acción de cambio de polaridad para la reversa y el giro a los lados. También, cabe recordar que en todo momento se debe tener la cantidad de energía correcta, hecho que se verifica con el multímetro.

Ya con las pruebas hechas se inicia el ensamblaje del circuito, manejando una buena distribución del mismo dentro de la tarjeta pre perforada.

Se recomienda ver los anexos de esta documentación para hacerse una idea de cómo se implementó el plan de pruebas.

## Reglas del grupo:

1. Llegar temprano a las reuniones.
2. Cumplir con las actividades dentro del tiempo determinado.
3. NO tener dudas, sino despejarlas con los compañeros, tutor o profesor.
4. Mantener el respeto en todo momento.
5. Mantener el orden y aseo al momento de trabajar.
6. Responder a las llamadas grupales a la hora de trabajar remotamente.
7. Los costos se dividirán de manera equitativa.
8. Si se va a faltar a reunión se debe avisar a los compañeros al menos un día antes.

## Actividades:

**Actividad uno:**

Investigar el cliente Python. Tiempo requerido: 3 horas.

**Actividad dos:**

Visualización de tutoriales Arduino. Tiempo estimado: 7 horas.

**Actividad tres:**

Compra de componentes, carro, baterías y bitácora. Tiempo estimado 2 horas.

**Actividad cuatro:**

Iniciar documentación y bitácora. Tiempo estimado 15 días, ya que va a medida que se avance con el proyecto.

**Actividad cinco:**

Implementación del código Arduino y cliente Python. Tiempo estimado 3 días.

**Actividad seis:**

Desarrollo de GitHub. Tiempo estimado 20 minutos.

**Actividad siete:**

Gestión de pruebas en protoboard (plan de pruebas). Tiempo estimado 3 días.

**Actividad ocho:**

Soldadura y ensamblaje del circuito. Tiempo estimado 4 días.

**Actividad nueve:**

Modificación de carro para que aloje el circuito. Tiempo estimado 2.5 horas.

**Actividad diez:**

Soldar luces y fotorresistencias a los cables dispuestos para ellas y pegarlas en sus lugares. Tiempo estimado 3 horas.

**Actividad once:**

Empacar el carro, revisar y finalizar documentación y bitácora. Tiempo estimado 3 horas.

**Actividad doce:**

Repaso de posibles faltantes y quehaceres. Tiempo estimado: una noche.

**Actividad trece:**

Añadir auto documentación a código Python. Tiempo estimado: una hora.

## Roles:

**Investigación:** Todos participan de manera individual y en grupos.

**Soldadura**: Todos participan.

**Implementación de código:** Todos participan.

**Bitácora:** Brandon Muñoz

**Documentación:** Marco Picado M

**Asignación de tareas:** Simón Fallas

**Modificación de carcasa de carro:** Marco y Brandon.

**Plan de pruebas:** Todos participan.

Los roles antes mencionados se acomodan para trabajarlo los días:

Jueves y Martes de 12:00 pm a 4:00 pm, Sábados de 8:00 am a 12:00pm o más tarde, Lunes de 12:00 pm a 5:00 pm. Todo esto para las actividades en grupo; las individuales como investigación, documentación, bitácora y testeo de código se llevan a cabo en los demás ratos libres de los integrantes y respetando el espacio de las demás materias.

## Bibliografía consultada:

[1] del Valle Hernández, L. (2019). NodeMCU y el IoT tutorial paso a paso desde cero. Recuperado de https://programarfacil.com/podcast/nodemcu-tutorial-paso-apaso/

[2]L298N datasheet. (2019). Recuperado de https://www.sparkfun.com/datasheets/Robotics/L298\_H\_Bridge.pdf

[3]  HD74LS164P datasheet (2019). Recuperado de: <http://www.alldatasheet.com/view.jsp?Searchword=Hd74ls164p&gclid=Cj0KCQjwla7nBRDxARIsADll0kA3qCxFM47O_X-65lcP0z52aLQzhSlkti8SAKSO2SMBkTcnOJrF8BcaAtzAEALw_wcB>

[4] Electronix, (2018). Curso de arduino desde cero. Disponible en: <https://www.youtube.com/channel/UCUJ6BMwZFHTnBozdGONtIhA>

[5] Rodríguez, J. (2007). Gestión de proyectos informáticos: Métodos, herramientas y casos. Rescatado de: <https://books.google.co.cr/books?id=I22YPj6iBisC&printsec=frontcover&dq=gestion+de+proyectos+inform%C3%A1ticos&hl=es&sa=X&ved=0ahUKEwj33KfJ2LziAhUCeawKHdX4B1MQ6AEIJzAA#v=onepage&q=gestion%20de%20proyectos%20inform%C3%A1ticos&f=false>

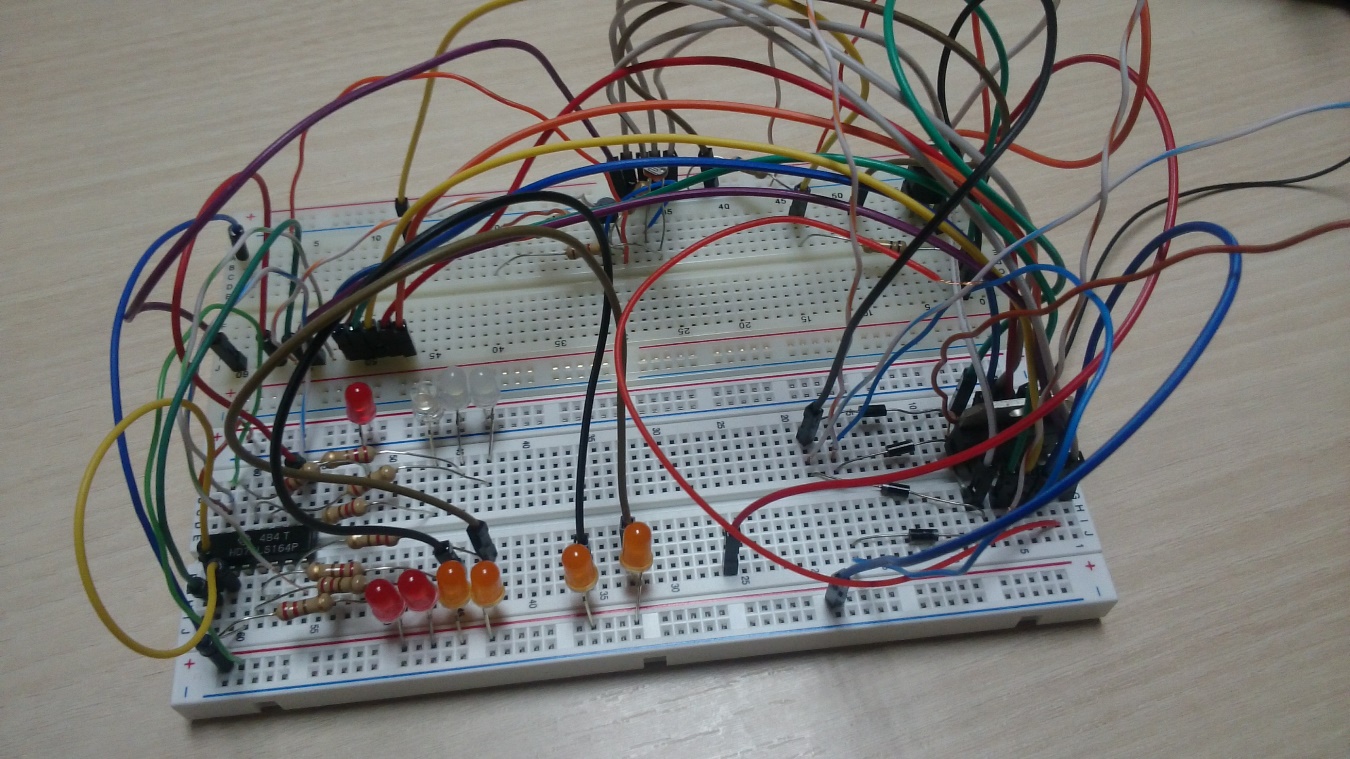
## Fechas entregables:

Jueves 30 de mayo: hardware.

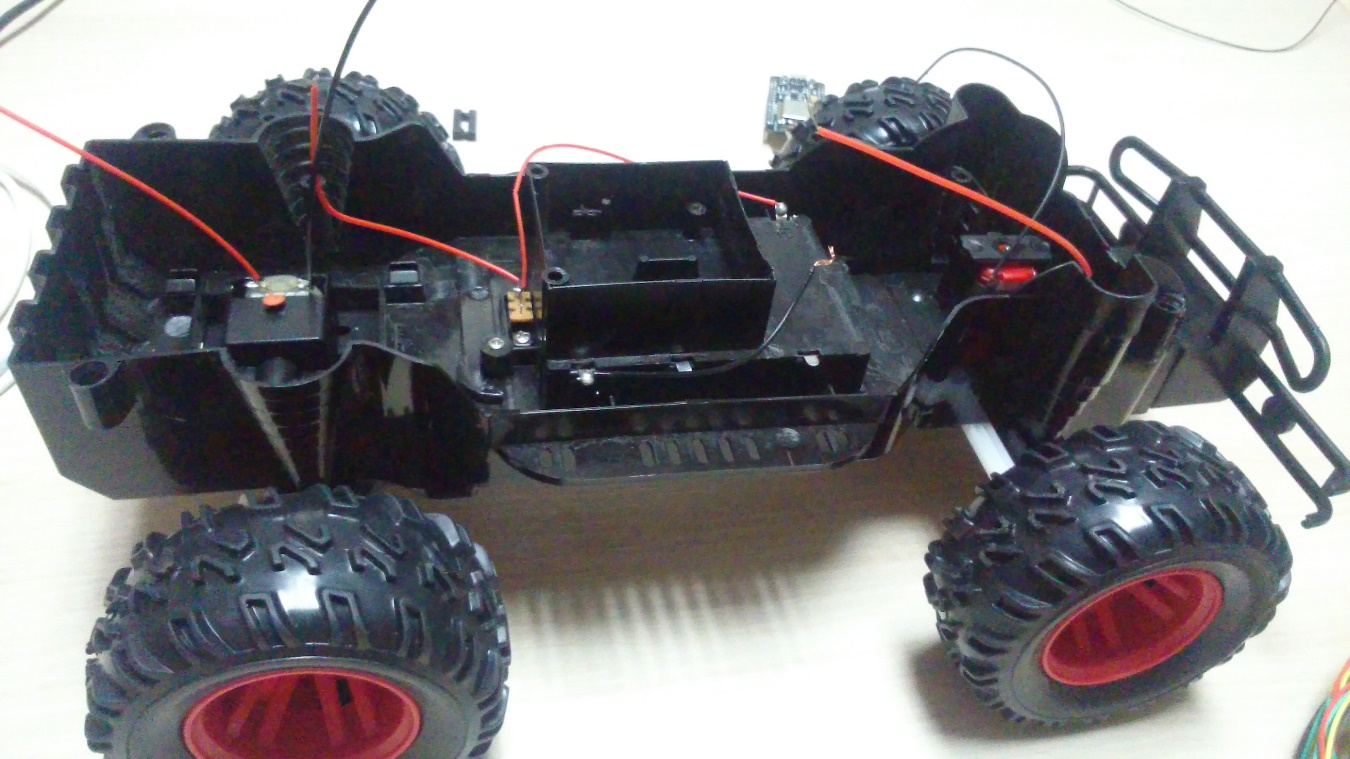
# ANEXOS:



Anexo 1: Modificaciones de carcasa.



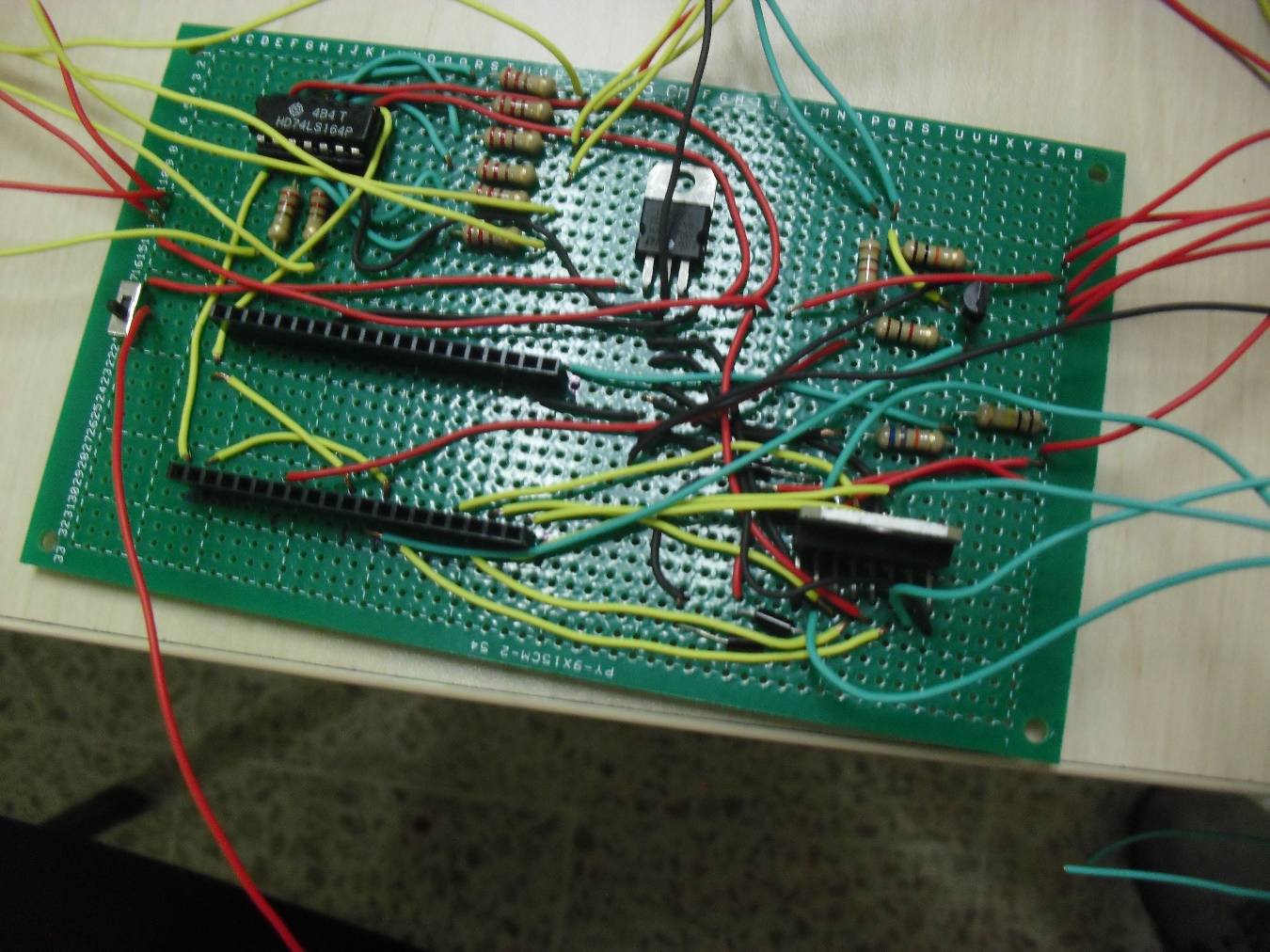
Anexo 2: Plan de pruebas.



Anexo 3: modificación de base del carro.



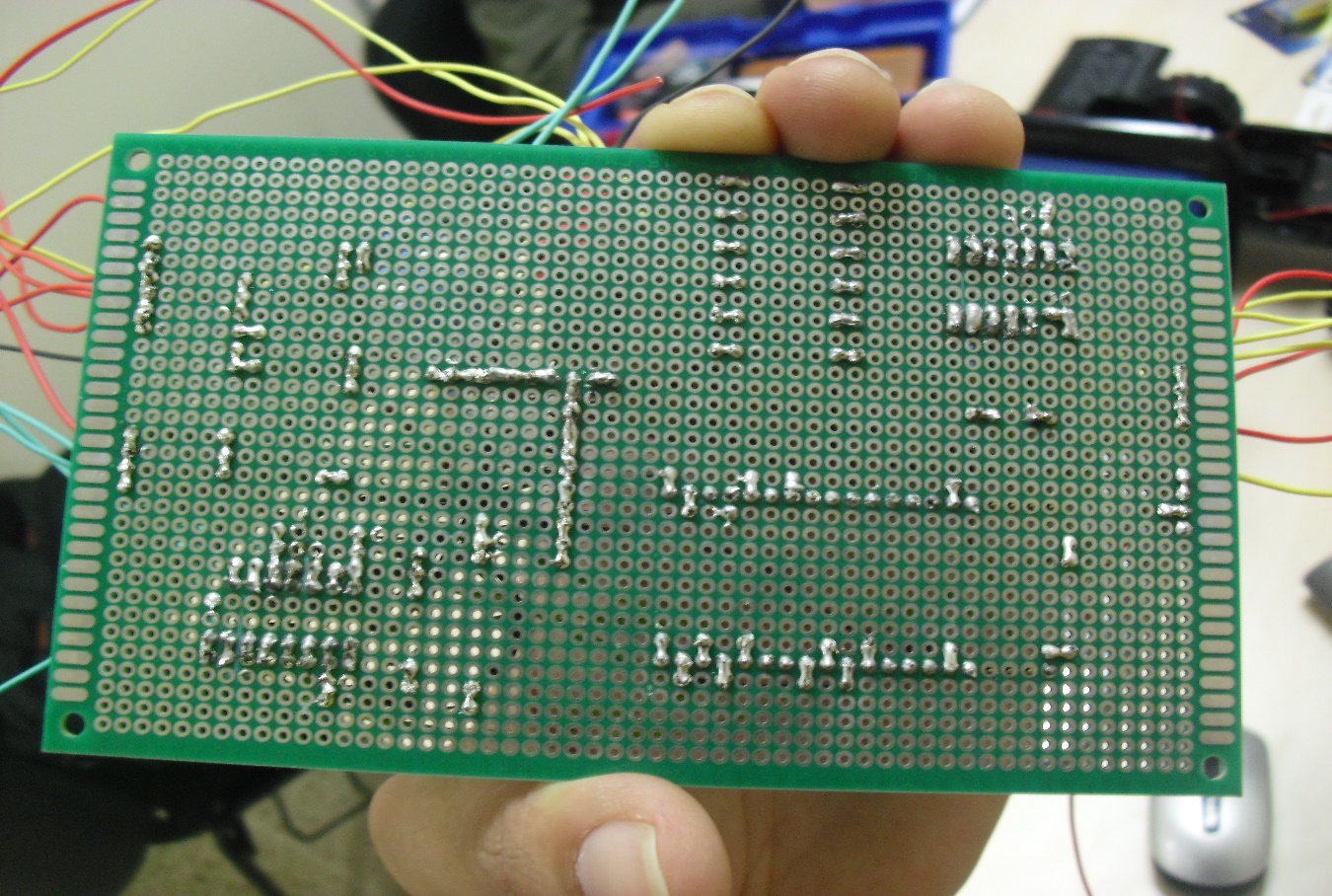
Anexo 4: Soldadura, todos participan.



Anexo 5: Circuito soldado parcialmente.



Anexo 6: Jornada de trabajo ordinaria.



Anexo 7: Distribución de soldadura en tarjeta pre perforada.



Anexo 8: Implementación de orificios para LED’s traseros.