

Instituto Tecnológico de Costa Rica

Ingeniería en Computadores

Taller de Programación

III Proyecto Programado:
FORMULA E II

Profesor: Jeff Schmidt Peralta

Grupo 1

Estudiantes:

José Julián Camacho Hernández; 2019201459

José Antonio Espinoza Chaves; 2019083698

I Semestre 2019

Tabla de Contenidos:

Introducción.....	3
Descripción del Problema.....	4
Análisis de Resultados.....	5
Diagrama de la Arquitectura.....	5
Diagrama de Módulos.....	5
Plan de Pruebas.....	8
Reglas del Grupo, Roles.....	8
Fechas de entregables.....	8
Dificultades Encontradas.....	9
Bitácora de Actividades.....	11
Estadísticas de Tiempos.....	11
Conclusiones.....	13
Recomendaciones.....	14
Literatura o Fuentes Consultadas.....	14

Introducción

La telemetría es una tecnología que ha venido mostrando una evolución a través de los últimos años. En el automovilismo, sobre todo en la Fórmula 1 y actualmente en la Fórmula E, se utiliza con el fin de recolectar datos sobre el rendimiento del vehículo. Da la posibilidad de obtener información importante del carro y efectuar diferentes acciones dependiendo de estos mismos. Es una tecnología de gran importancia y que incorpora diversos conceptos importantes sobre el desarrollo de computadores. En el proyecto se demuestra que es posible aplicarse inclusive a un carro de juguete, para así lograr controlarlo y obtener ciertos de datos mediante la telemetría. En el proyecto se verá la telemetría aplicada para la obtención del nivel de batería el nivel de luz en el entorno, etc.

Descripción del Problema:

El proyecto consiste en la aplicación de la telemetría a un carro de juguete a control remoto. De tal manera que reciba comandos por medio de botones en una ventana de interfaz (Test Drive), con el objetivo de controlarlo, mediante un servidor WiFi. Se debe utilizar el carro que se usó en el segundo proyecto del curso, el cual debía tener un circuito funcional donde el carro avanzara, girara, frenara, cambiara luces, hiciera movimientos especiales y recibiera un dato de luz en el ambiente; todo esto controlado por un Arduino WiFi (NodeMCU). Se debe utilizar una interfaz donde el usuario pueda ser capaz de interactuar con el carro, de tal forma que mediante el software sea posible controlar el hardware. Para conseguir esto, se deben utilizar canales de conexión entre ambas partes, denominados sockets.

Además se debe incluir una interfaz que contenga diferentes ventanas con funciones específicas cada una. Debe presentar una ventana de inicio la cual debe mostrarse al usuario una vez ejecute el programa. Aquí se presentarán toda la información importante de la escudería, el estado del carro, índice ganador de la escudería, una ventana para editar el logo y los patrocinadores. Debe incluir una ventana about con el nombre y foto de los integrantes y la versión de Python utilizada. También contará con una ventana que muestra una tabla de posiciones de competidores que está ordenada según la eficiencia y porcentaje de carreras ganadas.

Análisis de Resultados:

Diagrama de la Arquitectura:

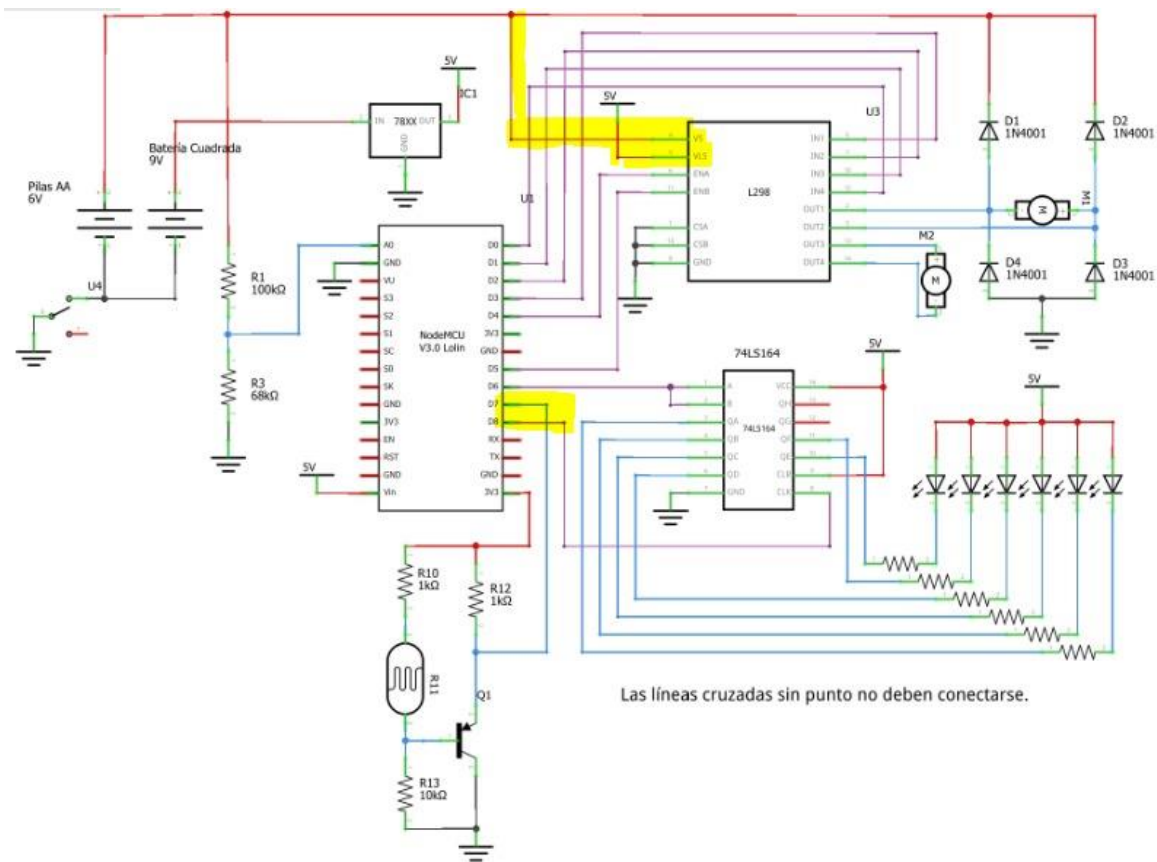
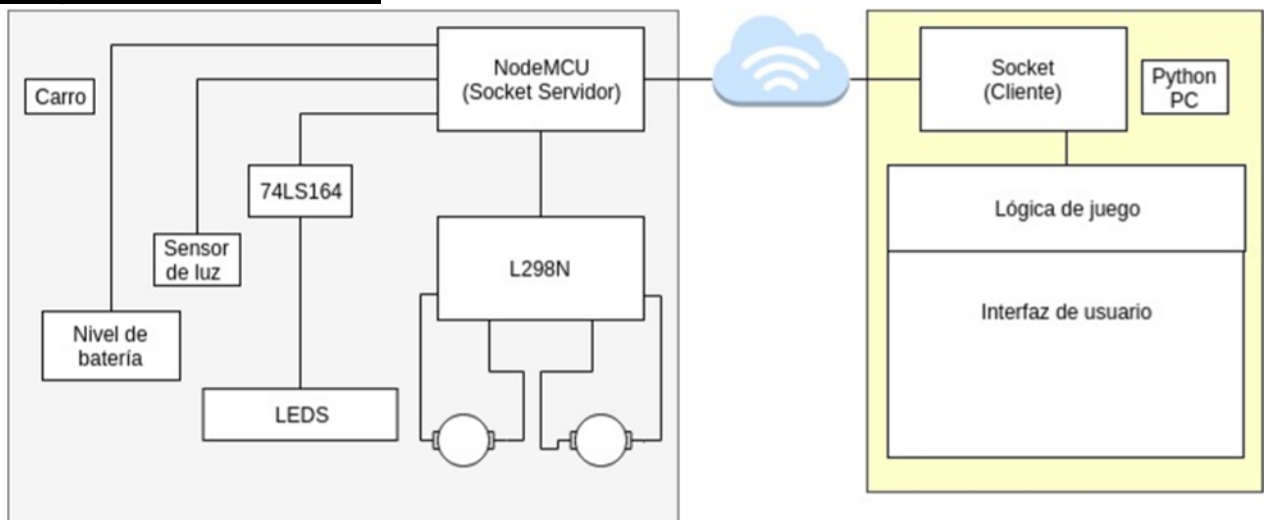
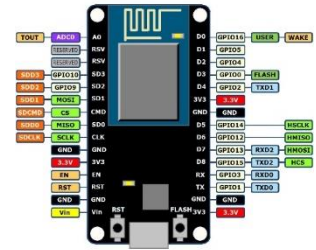


Diagrama de Módulos:

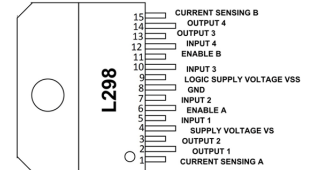


El carro debe tener los siguientes módulos:

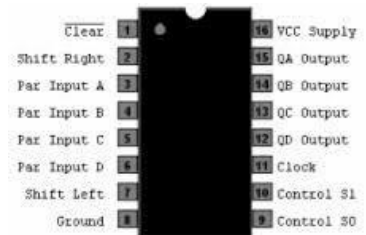
NodeMCU: contiene un microcontrolador y WiFi, recibe y genera las señales para los otros componentes. Debe conectarse por medio de un socket con el cliente en el software



Circuito integrado L298N: controla los motores que el NodeMCU no puede alimentar, además de cambiarle la polaridad a los motores para cambiar de giro.



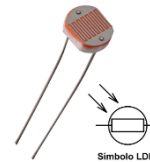
Registro de corrimiento 74LS164: permite controlar las luces del carro.



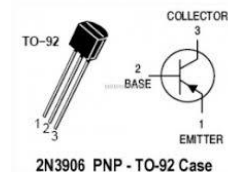
Dos motores: uno para la tracción del carro y otro para la dirección.

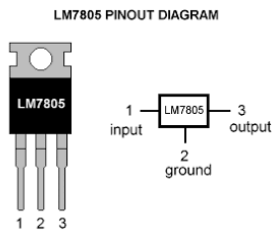
Seis leds: dos para las luces frontales, dos para las traseras y una para cada direccional derecha e izquierda.

Sensor de luz: genera un valor dependiendo de la cantidad de luz que perciba.



Transistor 3906: funciona como interruptor.





LM7805: regulador de tensión adecuada para los componentes.

En Python debe existir un socket del cliente para conectarse con el NodeMCU, la lógica de programación de funciones y la interfaz para que el usuario pueda controlarlo.

Diagrama de Arquitectura:



Plan de Pruebas:

Se realizarán pruebas por cada componente: pruebas para el funcionamiento de la fotorresistencia, de los motores con voltaje realizando diversos patrones, y las luces led tanto frontales como traseras y direccionales. Además las pruebas del método de ordenamiento de los competidores, pruebas sobre la implementación de programación orientada a objetos para creación de clases que facilitan la creación y modificación de los competidores y sus autos.

Reglas de Grupo, Roles:

Todo el proyecto se realizará de forma conjunta entre los dos integrantes del grupo. Lo primordial en el grupo es el trabajo en equipo y la responsabilidad. De esa manera se trabajará de una manera óptima con el objetivo de concluir el proyecto de la mejor forma y en el menor tiempo posible.

Fecha de Entregables:

13/6/2019: Entrega del pdf y código de manera digital, incluyendo la dirección del video.

14/6/2019: Entrega y Defensa del carro en el CIC.

Dificultades Encontradas:

Se encontraron problemas durante la creación del test-drive, como por ejemplo que al poner un string que mande un valor de pwm negativo (para que vaya en reversa) al momento de leer el mensaje el nodeMCU no lo ejecutaba. Para esto no se pudo encontrar ninguna solución. Otra dificultad encontrada fue que las luces no respondían al momento que girar esto pues se definió que al girar prendiera solo una de las direccionales dependiendo del caso y las otras las apagara mientras avanzaran. El problema es que si el auto estaba quieto si funcionaba; de lo contrario, o sea al estar en movimiento el motor de tracción, las luces se prendían por, lo que se cree que son señales basura del NodeMCU. A este problema no se le encontró solución. Otro problema encontrado era que a la hora de poner los comandos a ser mandados para cada función.

En el proceso de programación y desarrollo de la interfaz gráfica para los pilotos y automóviles de las diversas escuderías, se presentaron algunas dificultades que se intentaron solucionar de la mejor manera posible. La primera etapa fue plantear la forma en que se iban a trabajar las diferentes ventanas y funcionalidades de la interfaz. Fue necesario la utilización de diagramas para comprender la relación entre todas las ventanas, esto con el fin de trabajarlas funcionalmente.

Posteriormente, se presentaron dificultades al trabajar con clases, por lo que se tomó la decisión de desarrollar las ventanas con funciones, dado a la facilidad que brinda la librería TKinter. Además, se tomó la decisión de utilizar archivos de texto tipo “.txt”, ya que estos son fáciles de utilizar y son de uso común. Tanto escribir en ellos, como leerlos fue simple por medio del método “readlines”, en el que cada línea del archivo de texto se muestra como un elemento de una lista de Python. Sin embargo, esto significó otra dificultad, ya que al leer un entero de estas listas, se reflejaba como un string del entero finalizado con “ \n”, por lo que para trabajarlos como enteros, eran necesarias funciones para convertirlos.

Sin embargo, la mayor dificultad que se encontró fue en cuanto al ordenamiento de datos. Esta lógica fue particularmente compleja, ya que se necesitaba leer todos los archivos de texto de los pilotos y

automóviles, manejar grandes matrices con sus datos, recorrerlas, y poder imprimir el contenido de los archivos de texto en pantalla, una vez ordenados.

Finalmente, se presentaron inconvenientes con la selección de pilotos y autos para ir al Test Drive, así como para hacer una ventana por cada uno de ellos para ver sus datos completos. En el proceso también surgieron errores y problemas menores que fueron solucionados durante el mismo desarrollo.

Bitácora de Actividades y Tiempos:

Día	Actividad	Horas
Actividades Reunidos en Grupo		
24/5/2019	Se arreglaron conexiones del circuito y se volvió a montar la protoboard en el nuevo carro. Se soldaron cables necesarios y leds que estaban despegadas.	3
27/5/2019	Se trabajó la parte estética de la interfaz gráfica, como las fotos a poner, la comunicación entre ventanas por botones	4
31/5/2019	Se terminó la parte de las escuderías y la lista de posiciones, así como lo de añadir nuevos competidores y nuevos carros. Se empezó a trabajar la estética de la ventana del test drive, lo que son botones e imágenes.	4
3/6/2019	Se empezó a trabajar la parte lógica y de comunicación del test drive mediante el comando <code>nodeMCU.send ()</code> y con esto se logró hacer los botones para avanzar, otro para girar a la izquierda o derecha.	5

	No se logró hacer que hiciera para atrás.	
7/6/2019	Se hicieron todas las pruebas de movimiento, celebración, movimiento especial así como empezar a trabajar con el nivel de batería y el sensor	3
9/6/2019	Se empezó con la documentación y se siguió trabajando con los valores de batería y luz que estaban dando mal.	3
13/6/2019	Se revisaron los ultimo detalles del proyecto antes de entrega, se terminaron funciones incompletas, se terminó este pdf y se terminó el video explicativo del proyecto.	4
TOTAL		26 HORAS

Conclusiones:

- Una vez finalizado el proyecto, al ver y evaluar en retrospectiva todo el proceso que conllevó este trabajo, se pueden destacar varios aspectos. En primer lugar y como punto principal, se evidenció un crecimiento personal en diversos ámbitos. Por ejemplo, el proyecto logró su objetivo, ya que el conocimiento en programación aumentó considerablemente. Desde aprender más sobre la utilización de TKinter y archivos de texto, hasta la capacidad lógica para el manejo de interfaz mediante varias ventanas y el ordenamiento de información en las mismas. Otro aspecto fundamental que se vio estimulado en el proceso, fue la capacidad de organización del tiempo y trabajo en equipo. Estos resultaron de gran importancia para llevar a cabo el proyecto de manera óptima, ya que sin una correcta organización o con trabajo totalmente individualista, es prácticamente imposible conseguir el objetivo del proyecto. Por otro lado, el proyecto caló muy profundamente en mis aspiraciones personales. A pesar de que el proceso fue muy estresante e incluso frustrante en algunos momentos, la investigación y el acercamiento que se tuvo con el mundo del automovilismo, me generó un gran interés en cuanto a desarrollar dispositivos para carros, ya que es un mundo muy apasionante y en constante crecimiento junto con la tecnología de computadores.
- Se puede concluir que el proyecto fue un gran reto para nosotros, esto por el hecho de ser nuevo grupo de trabajo y no tener carro armado como otras parejas de trabajo. También me pareció muy interesante el tema, puesto que emociona el hecho de saber que mediante una interfaz gráfica hecha por nosotros se puede controlar un carro a control remoto armado por nosotros igualmente. Al terminar este proyecto personalmente reforcé mis conocimientos en TKinter e incluso conocí más sobre la manera de conectar con otros objetos por medio de un arduino. El proyecto me gustó bastante puesto que me gusta mucho la parte de electrónica y montar circuitos. Proyectos como este son un muy buen ejemplo de apenas una pizca de lo que se va a ver en un futuro en la carrera.

Recomendaciones:

Una vez terminado en proyecto se puede dar varias recomendaciones a la hora de efectuar un proyecto de este tipo. Entre estas recomendaciones está que si se está trabajando en parejas que se dividan el trabajo, esto es muy importante puesto que había dos grandes retos (la parte de objetos y la parte de telemetría), pero si se dividen las responsabilidades el proyecto sale con menor dificultad.

Otra recomendación sería investigar bastante acerca de la programación orientada a objetos y programación utilizando txt, esto puesto que gran parte del proyecto se basa en la manipulación y creación de este tipo de códigos. Por último, otra recomendación muy importante es hacer consultas a otros compañeros, tutores e incluso estudiantes más avanzados en la carrera puesto que es muy probable que sepan más que usted en temas donde se presenten dudas.

Referencias:

Gamboa, S. (2019). FormulaE_CE_TEC-. Recuperado de https://github.com/santigr17/FormulaE_CE_TEC-

Miller, B. (2014). Programación orientada a objetos en Python: Definición de clases. Recuperado de <https://interactivepython.org/runestone/static/pythoned/Introduction/ProgramacionOrientadaAObjetosEnPythonDefinicionDeClases.html>

Turkel, W. y Crymble, A. (2012). Trabajar con archivos de texto en Python. Recuperado de <https://programminghistorian.org/es/lecciones/trabajar-con-archivos-de-texto>