

Autor: Mauricio Romero Jofré
Fecha: 29 de octubre de 2017

Computador

La propuesta de computador del vehículo consiste en una Raspberry Pi. Ésta tiene las siguientes características [1], [2]:

- 26 pines digitales de propósito general
- Puerto USB.
- Puerto de audio.
- Puerto HDMI.
- Puerto Ethernet

La alimentación necesaria para la Raspberry es de 5 V con 2 A +o 700 mA (dependiendo del modelo). Se puede alimentar a través de un cable MicroUSB.

Dado el enfoque que ha tenido el desarrollo del proyecto lo principal es utilizar los pines de propósito general para tener comunicación con los sensores y pantalla para mostrar el estado del auto y sensores. Dentro de las posibilidades de programación de la Raspberry para crear el sistema operativo del automóvil está Linux(Raspbian)+Python o Linux(Raspbian)+ROS (Robot Operative System) [3]. En ambas plataformas es posible diseñar los sistemas de control necesarios para el auto.

Uno de los principales problemas entre la comunicación de sensores con la Raspberry es que las entradas y salidas que se pueden realizar con ella son sólo digitales, lo que produce un problema de adaptación con los sensores. Además, dado que los pines libres son limitados, eso produce una restricción respecto al número de sensores a conectar (tanto presentes como futuros). Existen diferentes casos.

Si los sensores son analógicos se deben utilizar ADCs de manera que exista compatibilidad entre los tipos de señales que puede recibir la Raspberry. En el caso de sensores digitales, hay que conocer el protocolo de comunicación que utilicen. Esto puede ser un gran problema, puesto que si se conecta cada sensor a cada pin de la Raspberry, se debe configurar por el lado de la Raspberry que distintos pines sean leídos de manera distinta. Esto puede acarrear un gran desorden si se conectan/desconectan sensores durante las pruebas. Una forma de combatir la situación es implementar una arquitectura de bus en el automóvil, de manera que sensores y actuadores compartan el esquema comunicativo entre sí. La complejidad de implementación está en acoplar a cada sensor y actuador electrónica que realice la interfaz entre el dispositivo y el bus. Se pueden utilizar Arduinos Nano como controladores del bus, dado su bajo precio y tamaño. Una virtud del esquema de bus es que se pueden acoplar más sensores que el pinaje de fábrica de la Raspberry.

Diferencial Electrónico

También me dediqué a buscar en la literatura formas de implementación de diferenciales electrónicos. Existen numerosos trabajos en la página de la IEEE, que dan cuenta que es un problema que está en constante desarrollo [4] [5] [6] [7] [8]. Existen modelos complicados, en donde se estiman fuerzas sobre ruedas, relaciones geométricas, distintos ángulos de giro, momento de inercia, etc. Además, en algunas publicaciones se refieren a como estimar parámetros necesarios para calcular posteriormente la velocidad en cada rueda. Otros buscan una implementación rápida del diferencial basada en la

programación de FPGAs. Lo principal que se puede concluir es que en [9] determinan una sencilla ecuación para determinar la velocidad de las ruedas de tracción a través de dimensiones del automovil, ángulo de giro del volante y velocidad angular de las ruedas. La piedra angular del modelo es estimar de manera precisa la velocidad angular de las ruedas y el ángulo de giro del volante, por lo que realizar un buen trabajo en la parte del computador se verá reflejado en el desempeño del diferencial.

Por último, se debe pensar en implementar un sistema que identifique cuando se está verdaderamente en una curva y cuando se está resbalando producto de las condiciones del suelo. La falencia de este sistema puede significar poner en riesgo la integridad de los usuarios y mundo alrededor.

Referencias

- [1] Raspberry Pi
<https://www.raspberrypi.org/documentation/usage/gpio-plus-and-raspi2/>
- [2] Raspberri Pi
<https://www.raspberrypi.org/documentation/setup/>
- [3] ROS
<http://wiki.ros.org/ROSberryPi/Installing%20ROS%20Kinetic%20on%20the%20Raspberry%20Pi>
- [4] Y. E. Zhao, J. W. Zhang, X. Q. Guan. Modeling and simulation of electronic differential system for an electric vehicle with two motor-wheel drive.
- [5] Xiaodong Wu, Li Yang, Min Xu. Speed following control for differential steering of 4WID electric vehicle.
- [6] Yee-Pien Yang, Xian-Yee Xing. Design of electric differential system for an electric vehicle with dual wheel motors.
- [7] Xiaodong Wu, Min Xu, Lei Wang. Differential speed steering control for four-wheel independent driving electric vehicle.
- [8] Merve Yildirim, Eyyüp Öksüztepe, Burak Tanyeri, Husan Kürüm. Design of electronic differential system for an electric vehicle with in-wheel motor.
- [9] Yiming Cui, Yi Shen, Xin Zhang, Yan Wang, Miao Zhang. Electric differential control for electric vehicles based on EMD method.