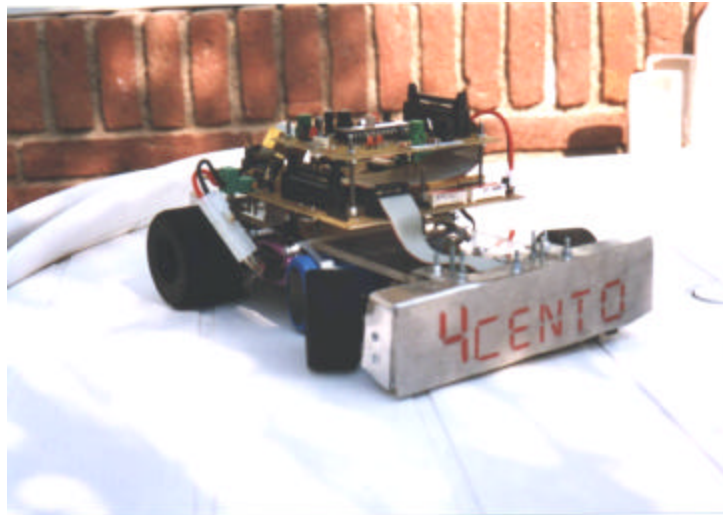


Micro-robot velocista Cuattrocento

Roger Blanch Machordom, Miquel Albiol Dalmau

Roger Blanch es estudiante de ingeniería Industrial en la UPC (roger.blanch@upcnet.es)

Miquel Albiol es estudiante de ingeniería en Telecomunicaciones en la UPC (miquel.albiol@upcnet.es)



Resumen

Cuattrocento es un robot velocista diseñado y construido por un estudiante de Telecomunicaciones y otro de Industriales. Su diseño está pensado para superar un circuito cerrado con curvas en ambos sentidos de radio relativamente grande (>75 cm). Supone nuestra primera experiencia en solitario, tanto para el mecánico como para el electrónico.

fototransistor dispuestos de tal manera que el transistor conduce más o menos corriente en función de la luz reflejada frente al sensor. Los alimentamos en continua, lo que nos obliga a protegerlos de la luz solar. Los emisores van a masa y los colectores a través de una resistencia a alimentación (configuración inversora). La tensión de colector va a un AO unipolar en modo comparador lo que nos permite ajustar al mínimo el consumo de los CNY70 ajustando la tensión de comparación.

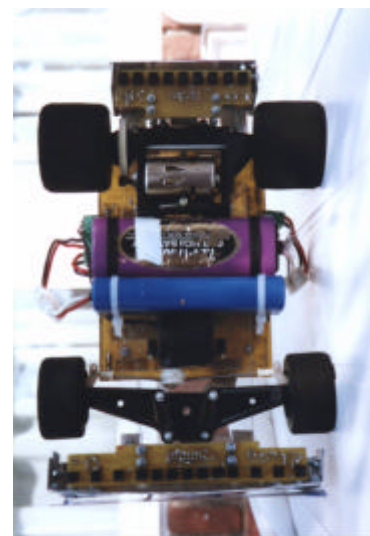
1. Introducción

Los principales elementos de Cuattrocento son dos arrays de sensores de línea, uno delante y otro detrás, un microcontrolador, un servo de dirección y un motor potente.

2. Elementos Hardware

El microcontrolador escogido para el control es un PIC16F877 de Microchip. Las principales ventajas de él son la posibilidad de grabarlo a través de un puerto de PC (puerto paralelo en nuestro caso) y la cantidad de pins de entrada / salida de que dispone. Consta de 8k de memoria de programa, 256 bytes de EEPROM y 368 bytes de RAM. Dispone de dos pins con funciones de PWM, que nos permiten controlar el motor y el servo de dirección.

Para detectar el contraste blanco-negro del circuito de velocistas hemos utilizado sensores CNY70 (13 delante y 9 detrás), los cuales consisten en un fotodiodo y un



La alimentación del motor y el servo está separada del resto mediante un par de optoacopladores para proteger la lógica y los sensores de posibles sobretensiones producidas por los bobinados, lo que supone el uso de dos baterías. Ambos, motor y servo de dirección, se controlan mediante PWM. El motor se conecta entre drenador de un transistor

POWER MOS-FET (BUK950855 de Phillips) y alimentación. El PWM del servo de dirección se aplica mediante otro MOSFET (BUZ11A) mediante una resistencia de pull-up.

En la placa principal del robot se han dispuesto arrays de leds indicativos de la posición de la línea en el eje delantero y trasero.

Finalmente la placa que contiene el microcontrolador está separada para poder grabarlo en el mismo robot. Incluye también cuatro botones (reset, start, y dos más para calibrar dirección y motor) y cuatro jumpers de configuración.

Los inconvenientes principales del hardware escogido son los sensores y el motor. Los sensores porque consumen mucho (0.5 A los 22 sensores dispuestos en el robot) y el motor porque es innecesariamente potente, lo que nos hace perder par-motor al atacarlo con un PWM de como mucho un 10%-13% de ciclo de trabajo.

El micro-robot está ligeramente sobredimensionado tras las modificaciones de las normas, especialmente que el circuito sea perfectamente ovalado, ya que el robot estaba inicialmente diseñado para sortear curvas continuadamente.

3. Elementos software

El algoritmo de control se ha programado en lenguaje C sobre la plataforma de trabajo MPLAB de Microchip, y compilado para el PIC16F877 con el compilador de MS-DOS HT-PIC que se integra en MPLAB.

El bucle principal de programa consiste en muestrear los sensores, obtener el error y actuar sobre los registros que controlan de forma transparente las señales de PWM. Si no dispusiéramos de las aplicaciones de PWM y tuviéramos que generar manualmente los PWM's, perderíamos mucho tiempo de procesador con interrupciones de reloj y muestrearíamos con mucha menor frecuencia.

El control consiste en el algoritmo PID para la dirección y un algoritmo que acelera el motor cuando este se encuentra en recta (sensores delanteros y traseros alineados, más o menos). La acción proporcional nos devuelve a la línea, la acción integral es útil para mejorar el comportamiento en recta (integral del error tiende a cero) y la derivativa mejora la entrada en curva (variación del error brusca).

Finalmente, hemos introducido algoritmos que nos permiten retornar a la línea si la perdemos, o no hacer caso de la línea exterior si nos saliésemos demasiado.

4. Elementos mecánicos

El diseño mecánico está basado en la estructura de un coche de radio-control. Se han aprovechado el sistema de ruedas, tanto las tractoras como las directrices, así como las respectivas banquetas que sujetan los ejes traseros y delanteros. Hacia falta rediseñar el chasis para que cupieran los demás elementos electrónicos, las placas, sobretudo las de los sensores, las baterías y cableados. También había que tener en cuenta las dimensiones permitidas. Todo el dimensionado está echo de tal manera que se aprovechan al máximo los recursos del robot, sobretudo en el sentido longitudinal, que al maximizar-lo obteníamos mejor respuesta de los sensores delanteros en cuanto a tiempo, lo que nos permitía una anticipación en la detección de un inicio de curva.

El chasis es de metacrilato transparente de un grosor de 5mm. Este material es suficientemente rígido para aguantar el peso de los componentes y permite un tratamiento de precisión en cuanto a agujeros y recortes. A su vez es bastante ligero, cosa determinante para la velocidad del robot

En cuanto al motor, como hemos dicho, esta un poco sobredimensionado, ya que el robot estaba inicialmente concebido con otras especificaciones. Debido a esto, procuramos reducir la velocidad de giro mediante un dispositivo piñón-corona, que reducía a 1/8 la velocidad. De hecho hacia falta mayor reducción, pero no ha sido posible reducir más. Esto nos ha dado bastantes problemas para controlar el robot.

Para la dirección usamos un servo convencional, unido a las ruedas directrices con dos barras de una longitud determinada cada una. El ajustar bien la dirección y la velocidad de cambio de dirección han sido nuestros principales focos de atención en el diseño del robot.

5. Conclusiones

Al empezar a hacer el robot sabíamos de antemano que hacia falta muchas horas de dedicación para lograr un resultado que mínimamente cumpliera con nuestros objetivos. Las horas han sido las que preveíamos y más. Ha hecho falta también ayuda de compañeros y también un poco de suerte para lograr un resultado satisfactorio.

El hacer un robot necesita de conocimientos en muchos ámbitos, no solo mecánicos y electrónicos. Hace falta conocimiento informático para la programación, que en definitiva va a ser el cerebro que ríga el comportamiento de este robot. También hace falta un poco de "intuición" sobre como se va a comportar el robot para tener-lo en cuenta para todo en general.

Al final hemos obtenido un resultado bastante satisfactorio del cual nos sentimos orgullosos y con ganas de mejorar para la próxima vez.

6. Agradecimientos

Agradecemos a amigos y compañeros de la universidad por compartir sus experiencias y conocimientos (y materiales) con nosotros. Gracias a ellos hemos ganado tiempo adelantándonos a problemas con los que nos hubiéramos encontrado.

Referencias

Como ya habrá observado no hay referencias en el texto, esto es debido a que nuestras fuentes de información fueron básicamente orales, juntamente con los datasheets y application notes de los componentes escogidos.

