Mr. 2-Bots

Michael J. Knapp

EEL5666C, Intelligent Machine Design Lab
Department of Electrical and Computer Engineering
University of Florida

Instructors: Dr. A. A. Arroyo
Department of Electrical and Computer Engineering
University of Florida, Gainesville, FL 32611

Dr. E. M. Schwartz
Department of Electrical and Computer Engineering
University of Florida, Gainesville, FL 32611

Date of Talk: 1 August 2005

Table of Contents

Abtract:

The retirement of one the University of Florida's favorite cheer leaders, Mr. 2-Bits, has led to an autonomous replacement, aptly named Mr. 2-Bots. The motivation in design of Mr. 2-Bots is having something that would both please the crowd as well as bring attention to the University of Florida's engineering program. A robust sensor suite was chosen, most notably a Global Positioning System receiver, to allow for the system to later be moved to a different platform, such as a blimp. In his current form Mr. 2-Bots navigates a series of GPS coordinates, called waypoints, and at each successive waypoint plays the next portion of Mr. 2-Bit's cheer, "2-Bits," "4-Bits," "6-Bits," "a dollar," "all for the Gators, stand up and holler!" using a modified digital sound recorder.

Executive Summary:

Mr. 2-Bots is an autonomous robot that navigates a set of successive GPS waypoints, stopping at each and playing a portion of the Mr. 2-Bit's cheer. Mr. 2-Bots consists of a platform, sensors and behaviors. His physical platform is a Traxxas Rustler. The motor controller and steering servo that came with the vehicle were both used, after having found that both are interfaced using a pulse width modulation signal with a 20ms period and 7.5% duty cycle as a center point. Since the motor controller only offers fine motor control in reverse, a relay was used to switch the direction of the motor and the controller is simply always operated in reverse mode.

Mr. 2-Bot's first behavior is collision detection. He handles this via two forward mounted SPDT switches. When either of these switches is triggered he reverses turning the opposite direction to attempt to escape the object. His second behavior is obstacle avoidance. He handles this by using a Devantech SRF-04 Ultrasonic Range Finder. When he detects an object approaching, he changes course to avoid it. Mr. 2-Bot's third behavior is GPS navigation. For this, he uses a Lassen iQ GPS Receiver. His current coordinate and previous coordinate are used to calculate his current heading and that is compared to the heading to the target waypoint. He then proportionally changes course based on the difference in headings. The final behavior is to play part of Mr. 2-Bit's cheer when a waypoint is reached. Here he uses a modified Olympus VN-180 digital voice recorder. It was found that the voice recorder grounded the button contact when a button was pressed and left it floating otherwise. This is simulated in software by leaving the I/O pin as an input and then setting the pin to an output of logic zero for 1 ms.

Introduction:

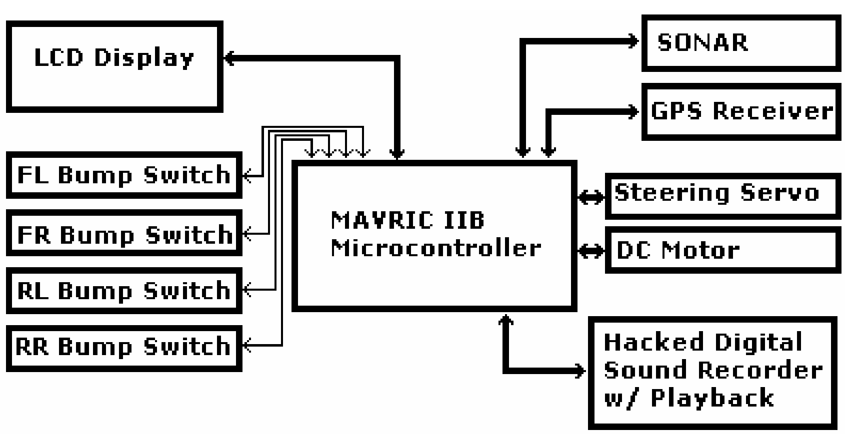
This paper covers the development process of the autonomous vehicle named Mr. 2-Bots. The name was chosen as a tribute to someone who is both a Gator favorite in The Swamp as well as the reason for the design of this vehicle, that someone is Mr. 2-Bits. The problem arose in 2001 when Mr. 2-Bits officially retired. Since then he has continued to make special guest appearances at home football games, but no one knows how long that will last. That is why an heir apparent was developed for someone who has led Gator cheers since 1947.

First, the overall layout of the integrated modules that comprise Mr. 2-Bots is discussed. Second, the physical design of Mr. 2-Bots is covered. Third the control of the physical components is reviewed. Fourth all of the sensors necessary for Mr. 2-Bots to garner information about his physical environment are examined and finally, how the physical components of Mr. 2-Bots coalesce into his designed behaviors is analyzed. From there both the experiments performed as well as the conclusions reached are listed.

System Design:

(a) Integrated System:

Mr. 2-Bots is system comprised of an AtMEGA128 based BD Micro MAVRIC-IIB microcontroller board along with ten external sensors and actuators. Mr. 2-Bots is actuated by a steering servo and driven by a DC motor. Mr. 2-Bots is able to get information about his physical environment through four SPDT switches, one SONAR sensor and one GPS receiver as well as communicate to the user through an onboard LCD and finally play the appropriate cheer via a hacked digital sound recorder and playback module.



Block Diagram showing Mr. 2-Bot's Design

Using the Bump Switches, GPS Receiver and SONAR, Mr. 2-Bots is able to successfully navigate to successive GPS coordinates by driving itself with the DC motor and steering with the servo. Mr. 2-Bots is able to play the prerecorded cheers via the hacked digital sound recorder as well as use the LCD display to send the user feedback.

(b) Mobile Platform:

Mr. 2-Bots is based off of the chassis of a Traxxas Rustler 1/10th scale R/C racing car. This platform is ideal for Mr. 2-Bots because was designed to allow for easy mounting of the modular components that are necessary on an R/C car, so replacing these with modules for the autonomous vehicle was not a difficult undertaking. The motor, motor control and steering were left unchanged from their original forms. Each used a standard pulse width modulation signal of period 20ms and center duty cycle of 7.5% to interface.

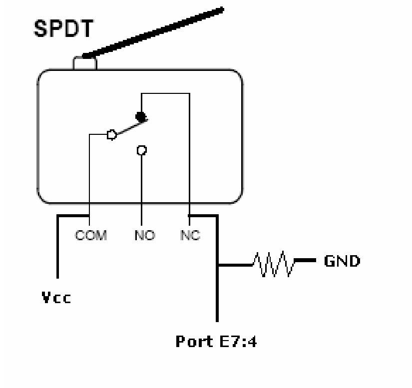
(c) Actuation:

The actuation of Mr. 2-Bots is through two modules. The first is a servo mounted to the steering rack of the chassis. Since Mr. 2-Bots steers like a normal automobile, a servo was needed to turn the rack left and right as well as hold it center. The second module is a small DC motor mounted to the rear differential of the chassis. This drives the rear axle of Mr. 2-Bots, which is controlled by a Traxxas XL-1 motor controller.

(d) Sensors:

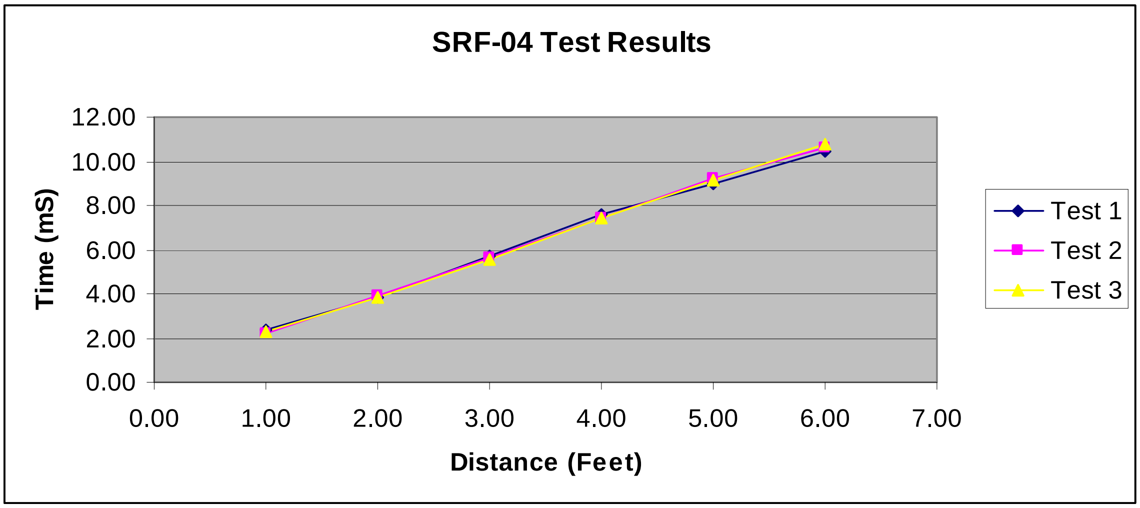
(i) The first sensor Mr. 2-Bots employs is a set of four single pole double throw bump switches, which are made by Omron with a part number of D2F-L, at each of his four corners. They are each wired to be active low. The pole of each is then connected to ATMega128's external interrupt 3:0 pins. In software, the interrupt is set to fire continuously on active low and the interrupt handler sets a flag for the specific switch and disables the specific switch's interrupt so it will only fire once and is reset when the flag's status is checked. In this way, no physical contact is ever unaccounted for if polling where to have been used and having the interrupt fire continuously on low ensures if Mr.

2-Bots were to be trapped against an object, it would register until he were freed from contact with the object. One learned lesson was to be sure to secure the front switches as an accident with a curb at 1.5m/s does in fact destroy the switches, which happened a few times. Also when said accident occurs just before demo day, there is no time to order replacements since. The diagram below shows the actual wiring of the switches with Vcc as the logic high output from a I/O pin.

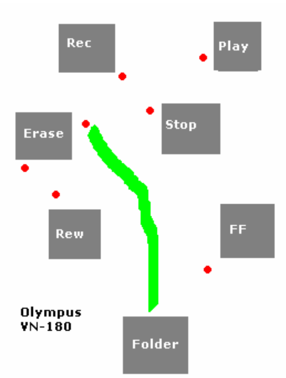


Bump Switch wiring diagram

(ii) The second sensor is a Devantech SRF-04 Ultrasonic Range Finder. This sensor works by taking an input pulse of 10uS and then later returning a pulse of varying width based on the distance of the objects ahead with a maximum pulse of 38 mS if no object is detected. For this sensor I wired the return pulse pin to the ATMega128's input capture pin. This allows me to have an interrupt fire exactly when the return pulse ends as well as not waste any clock cycles on polling the return pulse pin. Mr. 2-Bots uses hard coded times to determine how close he is to objects. Based on the results of an experiment performed, the following are the times used for the given distances are given below. The important point to note is that the time appears to grow linearly with distance at a slope of approximately 1.5 as well as having the results be fairly repeatable. This allows for the actual distance to be determined at any range.



(iii) The third sensor is a modified Olympus VN-180 digital voice recorder. Since Mr. 2-Bots must be able to playback at least five different sound clips, it was necessary to find something with said capability. The most economical solution was to modify the digital voice recorder already in possession. Upon opening the VN-180, it was found that a button press is registered by contact with ground while the button is depressed and the button is left floating while the button is not depressed. This proved extremely simple to implement in software as the ATMega128 I/O pin associated with the button is left as an input to simulate floating and changed to output a logic zero for a few microseconds to simulate the actual button press. Wiring the VN-180 also proved easier than expected as it was found that each button had a solder pad near it. This bypassed the need to remove part of the contact to solder to. The wiring diagram shows the solder pad in red and is associated with the button nearest it, except the "Folder" button whose solder pad is near the "Erase" button.



(iv) The final and special sensor is a GPS receiver. After researching different GPS receivers, it was settled on the 12 Channel Lassen IQ GPS Receiver with DGPS as the best choice. This also requires an antenna, so the Ultra-Compact Embedded Antenna for Lassen IQ was decided upon for its size and direct mating to the Lassen IQ receiver. To mount the receiver an SMT mating header was required. Since Mr. 2-Bots runs both the microcontroller board as well as the sensors at +5v DC, communication is done at RS-232 levels. The Lassen IQ receiver was specifically chosen for its ability to return position and velocity data as well as do as much at set intervals as well as its implementation of the proprietary Trimble protocol, TSIP. This will allow Mr. 2-Bots to always have a stream of current velocity and position data allowing frequent course correction. The receiver also returns the velocity in ENU (East, North, Up) format in meters per second allowing heading to be easily calculated. Using velocity in this format along with the GPS coordinate, Mr. 2-Bots can always know his position and heading relative to the target GPS coordinate. Mr. 2-Bots uses the velocity data as well to keep his speed at roughly 1.5 meters per second.

(v) Mr. 2-Bots also has a Hitachi controller based LCD display for communicating information to the user.

(e) Behaviors:

(i) The primary behavior is collision recovery. Here Mr. 2-Bots must immediately respond to a collision with an object and successfully recover. This behavior takes precedence over all other behaviors. Once collision recovery is successful, then normal behavior resumes.

(ii) The second behavior is preemptive object avoidance. Here Mr. 2-Bots uses information about objects in his path to adjust his course to avoid the object while still maintaining an overall course towards the next waypoint.

(iii) The third behavior is navigation is Mr. 2-Bot's most complicated behavior. Here Mr. 2-Bots will determines his current coordinate and orientation to his next coordinate. This is done by keeping a record of previous coordinates.

(iv) The final behavior is detection of successfully reaching a waypoint and playing the appropriate portion of the cheer, then setting the next waypoint as his new destination. Also, if Mr. 2-Bits does not have any waypoints programmed, he will set 5m north, south, east, west and current position as the first through fifth waypoints respectively.

Experimental Layout:

Once all of the modules had been integrated, testing was performed by video recording Mr. 2-Bots perform his programmed behaviors. The video is available on the accompanying CD. As can be seen from the video, Mr. 2-Bots does make a rough square centered on his starting position, with the first point being north. This is video evidence of him completing his task.

Conclusion:

In conclusion, it was found that the most difficult task was to navigate small distances. Since the GPS receiver only provides a fix at most once per second and the instructors wanted Mr. 2-Bots to travel at a reasonable velocity for humans, navigation had to be done on very few position updates. However, Mr. 2-Bots is able to navigate larger distances quite well. In the end, this is his biggest limitation. With respect to all other portions of the project, Mr. 2-Bots is technically complete.

Future work on Mr. 2-Bots includes first implementing a Kalman filter for GPS navigation. Currently his heading navigation is crude and would be much improved with a compass, but using a Kalman filter for navigation should help him to navigate much more precisely. Second, adding orange and blue LEDs to Mr. 2-Bots would improve his visual appeal. Third, I would like to have the SONAR distance thresholds be variable based on speed. Fourth, I would implement a floating point coprocessor to properly deal with the floating point data returned by the GPS receiver. Finally, my parser for the stream of GPS data is not perfect and a packet could be lost if the data portion of the packet contains two juxtaposed sets of ETX and DLE bytes. It is necessary to parse based on the beginning of the packet being a lone DLE byte and not the end of the packet as is currently implemented.

If the project were to be redone, a different platform would be used. Since the R/C platform was designed for pavement or low cut grass, Mr. 2-Bots is unable to navigate rougher terrain. Specifically, I would use a Traxxas E-Maxx for its size and high torque motors. I also may attempt to write the code in assembly language. The AVR-GCC compiler used is not trusted since code size was noticed to sometimes change by large amounts when one extra line of code was added to say turn on an LED for debugging.

Documentation:

[1] PonyProg Tutorial. 2003. Len Payne. 26 May. 2005 <http://www.techonthenet.com/oracle/index.htm>.

[2] The GNU C Library. 2001. Free Software Foundation. 26 May. 2005 <http://www.gnu.org/software/libc/manual/html_node/>.

[3] Lassen iQ GPS Receiver System Designer Reference Manual. February 2005. Trimble Navigation Limited. 31 July 2005. <http://www.sparkfun.com/datasheets/GPS/Lassen%20iQ_Reference%20Manual.pdf>.

[4] GPS Coordinate Converter. 2005. Jeff Boulter. 31 July 2005. <http://boulter.com/gps/>