Autonomous Prime

University of Michigan-Dearborn

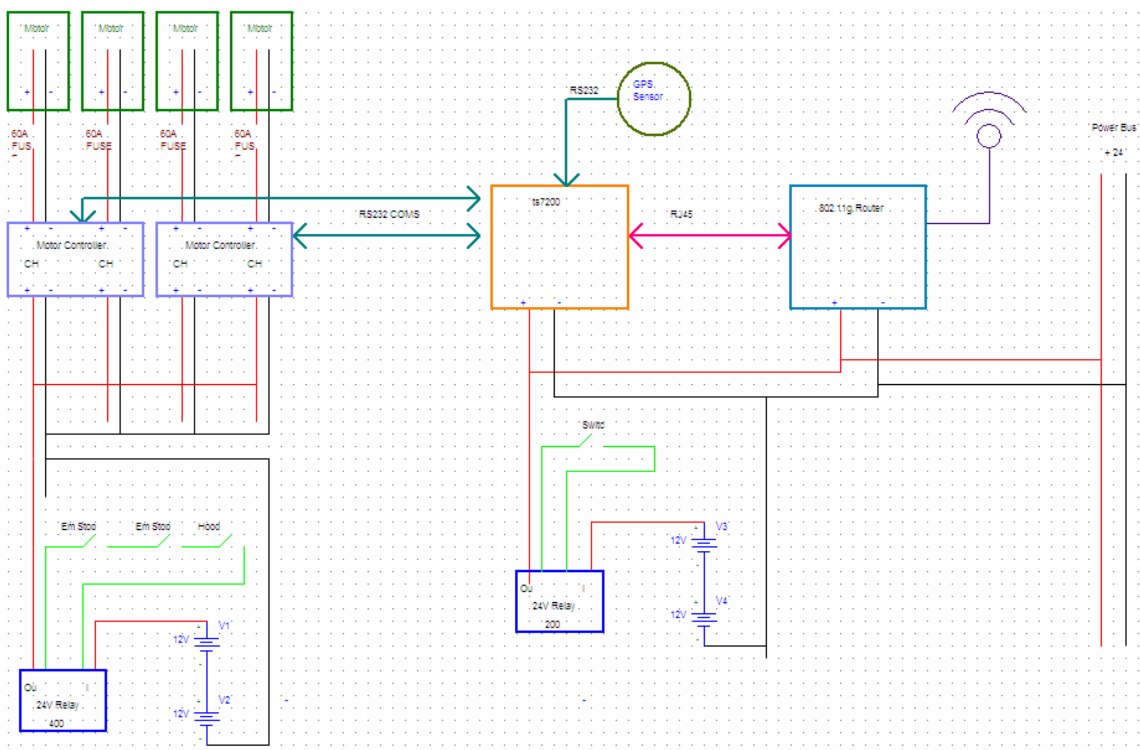
Anthony Lucente, Jonathan Hyland, Michael Kinnell,
Richard Chase

Department of Electrical and Computer Engineering
University of Michigan-Dearborn

June 10-12th, 2006

Introduction

Autonomous Prime was designed to be a multi-functional vehicle that can operate in almost any terrain. The vehicle platform itself was single-handedly designed by a UMD student to complete his master's work, which was then to be integrated and tested with other team members' autonomous sensors. The final objective was to create a safe and robust robotic platform for almost any application that can easily incorporate into a wide array of electronics and sensors. Autonomous Prime's design is geared for the 2006 IGVC competition, but is certainly not limited to this specific function. The platform is compatible with almost any new hardware, software, and variation in functionality. Since the concept of the architecture is simple and effective, the concept can be scaled to a small servo powered vehicle, a larger military vehicle, or anything in between.



-- System Diagram --

Design Innovations

The most important innovation is Autonomous Prime's simple, yet versatile architecture. Since the vehicle is controlled by UDP packets, there are many devices that can operate it: a handheld 802.11 remote, a laptop or PDA with wireless capability, or even a computer hardwired into the network, located directly on the robot. With a 4 wheel drive system where each motor is a geared 1HP DC brush motor, the robot can go through almost any terrain with no issue.

A second innovation is the design of the chassis itself. Since the robot was 3-D modeled and water-jetted, the robot can be easily re-fabricated. All the mounting points for electronics and mechanics are cut on the water jet, meaning that there is virtually no drilling or modifying needed after the robot is constructed. Tabs and slots were created with appropriate offsetting, allowing for extremely easy welding. For example, a piece of aluminum being welded perpendicular into a ½ inch piece of aluminum is only giving a ¼ inch tab. This leaves a ¼ hole that the welder can back fill which creates an extremely strong weld.

Thirdly, the power of the robot is easily changeable and long lasting. Both 24V systems can be changed to paralleled 12V systems to allow for extremely long battery life of over 20 hours without using a gas generator or charger. This estimate is for continuous use, so if the robot was not in continuous use, the power would last for days on end. Since both the electronics and motors run on separate systems, there is virtually no noise and back feeding in the electronics from the motors, which eliminates many problems that can occur.

Furthermore, the motor controllers are extremely efficient when it comes to power. They can be configured to ramp up speed on the motors to eliminate high start currents. Without a ramping of the motor speed, startup currents can reach up to 300A. By changing the ramping speed, the max startup current is around 50A, and only requires around 10A to keep running. Power distribution to the motors can also be controlled. If the motor on channel 1 runs a little slower than the motor on channel 2, the motor controller can be programmed to send a little more power to the first motor so that the motors on both channels will spin in sync.

The software itself is also innovative. The main program is written in such a way that if a critical system should fail, the robot will safe itself. For example, if the robot's E-stop was hit and the motor controllers turned off, the code will know that the motor controllers are not present. The robot will not operate again until both motor controllers are online. If one of the motor controllers is online and the other is not, the robot will continue to stand still until it sees both motor controllers. Also, if a ½ second goes by and the robot has not received a UDP packet, the robot will go into safe mode and stop. This prevents the robot from running away in an autonomous and remote control situation. If the robot is 1 mile away, and your transmitter loses power, then the robot will simple halt until it gets a command, instead of carrying out its last command.

Components of the robot are also easily replaceable should they fail. The motors, motor controllers, batteries, and electronics can be removed with either a set of wrenches or Allen keys. Even though items can be removed with a wrench or Allen key, items are secure enough to withstand heavy usage without any shifting whatsoever. The plate holding the electronics is secured down with pushpins. A simple pull of the pin will either lock the panel in place or release it to give access to the electronics underneath. Even the hood itself can be replaced by popping out four pins.

Mechanical Design

The mechanical design of the robot is extremely sturdy. The hull is made of 3/8 aluminum and the hood is made of ¼ inch aluminum. The hull can take an extremely large beating and much wear and tear before any significant damage occurs. The motors are mounted underneath the hull with 6 bolts, allowing for near perfect alignment on 3 axes, and the batteries are placed symmetrically inside the robot to construct a near 50/50 weight distribution. This near perfect distribution helps when turning the robot.

The robot is steered using skid steering as opposed to Ackerman steering. An advantage to skid steering is the physical construction of the robot. Ackerman steering, which is like the steering on a car, requires a lot more mechanical parts. If the system is not designed correctly, the car will not steer properly. Skid steering is easier to design, but there are some drawbacks. The robot can not move full speed and turn at the same time, so the motors are not being used to their full potential. However, the robot can change its direction without changing its position. Since the weight distribution is 50/50, the robot is able to stay in one spot and turn without much change in its x and y position.

The batteries are secured in position by a 1 inch high frame and bungee straps. The charger, motor controllers, and 802.11 routers are mounted to the frame with either high grade Velcro or bolts. An electronics plate can be placed on top of the batteries and secured with 4 quick release push pins. The plate can also be interchanged with a chair to allow someone to ride on it. However, the electronics and switches must be mounted inside the hull before this can take place. The main purpose of the electronics plate is to allow for easy access to switches and for other devices that may need debugging.

The hood is closed using a latch, and gas struts inside the robot will automatically lift the robot when the latch is released. Chains placed near the gas struts insure that the struts are not fully extended, which will cause damage to them in the long run.

Electrical Design

For the robot to be robust and easily expandable/interchangeable, there are 2 separate electrical systems on the robot, each controlled by an individual relay. The first system is the electrical system that drives the motors; the second is the system that drives the electronics. Both of these systems are 24V systems with similar connectors to allow for interchangeability in between the two banks of batteries. If there is not enough power in one bank of batteries to run the motors, then they can be easily switched with the bank of batteries running the electronics. The electronics generally require less power than the motors, meaning that the batteries are normally well charged or have enough power to run the motors in an emergency situation. The battery banks consist of two 12V batteries in series, which can easily be changed to two paralleled 12V batteries so that electronics and motors can be run off 12V instead of 24V. The motors can also be run off of 12V to obtain a much longer battery life.

The motor circuit runs two dual motor controllers which in turn each drive two motors. This means that each motor can be driven independently, allowing for precision controlling when encoders are used. The two motor controllers are connected to the batteries through a 400A relay which is controlled by 3 emergency stop switches, and the motors are connected to the motor controllers through a 60A circuit breaker. Because the circuit breaker is 60A, thermal, manual resetting breaker, the current can exceed 60A for a reasonable amount of time. This allows the motors to pull large amount of current if it gets stuck or meets some resistance, but the circuit will break if the current is too high for too long.

The electronics circuit runs any electronics that the team may need through a 200A relay controlled by a switch. The system is also 24V, and was set this way to allow for easy incorporation of any new electronics. Items that use lower voltages can simply be controlled by a small regulator running from the 24V. The relay is wired up with a spider's web of small Anderson type connectors to allow for quick connect/disconnect from the circuit.

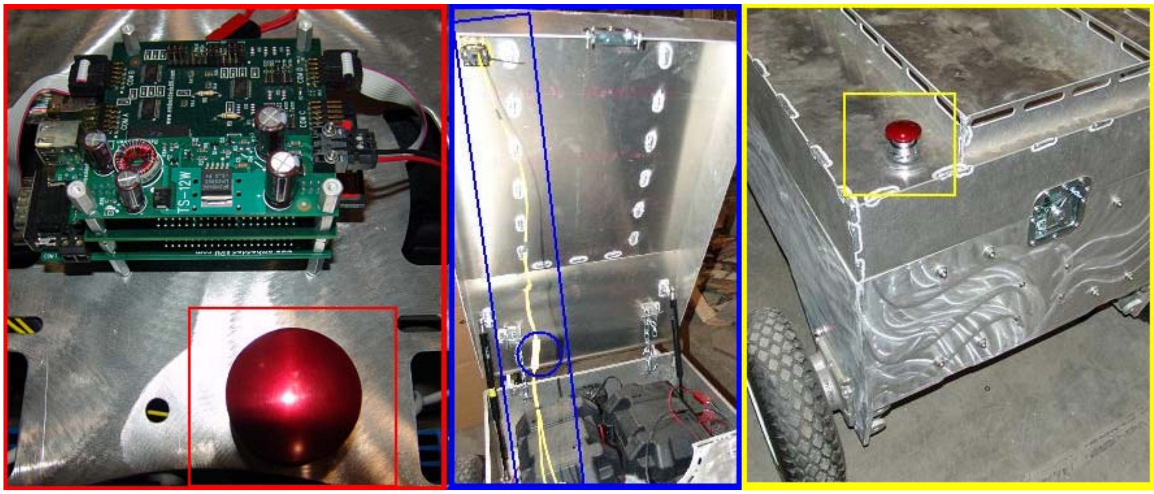
Embedded Circuit

The main motherboard controls all essential operating functions of the robot. Both motor controllers and a GPS unit are connected to the motherboard through serial ports. The motherboard runs a full Linux OS and accepts remote commands through UDP packets over the network. Communication with the motor controllers is continuous, and any interruption in communication will halt the robot for safety. The motor controllers are on a separate power circuit than the electronics, so if the safety switch is engaged, this will not effect the electronics. The motherboard is programmed to realize that the motor controllers are not online, and will take actions accordingly.

The GPS data is in NMEA format, and is taken into the motherboard through a serial port. The data is then placed into a C++ class, sent over the network and is able to be received by any other device on the laptop. If no GPS is available, or the GPS receiver can not see a satellite, then the motherboard will transmit all zero values in place of junk or old data.

Emergency Stopping

There are 3 ways to stop the vehicle in an emergency: 2 emergency stop switches, and a hood quick release/safety pin. All 3 are wired in series to the 400A motor relay, and all 3 switches must be closed in order for the relay to operate. When the switches are activated, all power to both motor controllers will be cut. This will not affect the motherboard, but when the motors are turned on, the motherboard will take a second or two to reacquire a communication with the motor controllers. To open either of the 2 emergency stop mushroom switches, the switches must be pressed down. To close the circuit, the switches must be pulled up. The hood release is simply a quick release Anderson that disconnects when the hood extends all the way. An Anderson connector is wired to the relay, so that any of these switches can be bypassed. For example, the mushroom switch on the hood can be connected directly into the relay, bypassing the mushroom switch on the electronics plate.



-- Emergency Stop --

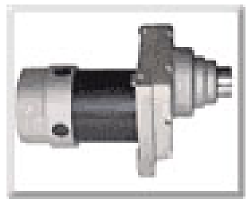
Motors

The motors are 24V DC brush motors, which can be run on 36V for more demanding applications. They are 1HP motors, with a gear ratio of 19.963:1. With such a large horsepower and gear ratio, the robot wheels have an extreme amount of torque, capable of getting the robot trough most any obstacle. Running with minimal resistance, the robot will sustain top speeds using 10-15 amps. The initial startup current can be near 260A, but the motors' speed can be ramped using the motor controllers, which will cut down on this current spike significantly.

The motors are fitted with an Anderson connector that connects to a 60A circuit breaker, which then connects to its respective motor controller. The breakers will sustain 60A for a long period of time. For convenience, the breakers can be tripped or reset just by flipping a switch.

The motors are mounted to the chassis in 6 different locations on the robot. This allows for the motors to be aligned as best as possible with the robot. Note, when installing the motors, it is best to only thread the bolts in a few turns to allow for some play when trying to fit the rest of the bolts in. It is important that all bolts are in so that the motor stays in its designed orientation.

The motor mount is also reinforced with additional welding. Side panels are put on each side of the motor and welded to the bottom of the hull for added durability. Even though the motors are surrounded by aluminum, the motor can be replaced simply by removing the 6 bolts holding it in.



-- 24V DC brush motors --

Motor Controllers

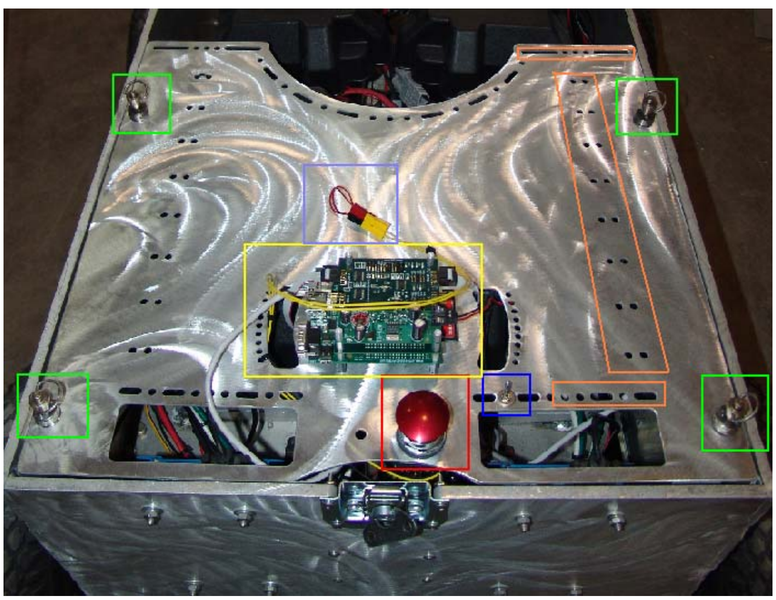
Both channels of the motor controllers are connected to the each motor with an Anderson connector. Along with the motor wires coming out of the back of the motor controller, there is also a wire that will turn the motor on or off, as well as an emergency stop wire. These 2 wires are not being used in the current configuration, but can be used in later projects. The motor controllers are mounted to the rear portion of the hull. There is also an optional mount for the TS7200 motherboard in between the two motors.

The motor controllers are setup to communicate through RS232 7E1. If a message is not received by the motor controllers at least once per second, the watchdog timer in the motor controller will trigger the motor controller to safe itself. Although the RS232 communication portion of the controller can not be changed, there are several ways to communicate with it, either by PWM or a simple 0-5V signal. In its current configuration, the motor controllers communicate through RS232 with the main processor on the robot. Most every other option on the controller is configurable: speed ramping, encoder usage, power distribution, etc. For specifics on how to communicate with the motor controller, please refer to the motor controller manual. How the motherboard communicates with the controllers is discussed in the software section.

-- Motor Controller --

Electronics Plate

The electronics plate is placed on 4 tabs above the rear bank of batteries. It is designed to be easily interchangeable as well as easy to modify. It is made of 1/8 inch aluminum so that holes can be easily drilled in it. There are also many holes for zip tying down cables, and it is held in by 4 quick release pins for easy removal and access to the electronics underneath. The motherboard of the robot can be mounted on the inside of the hull, the electronics panel removed, and a seat can be mounted with the quick release tabs for riding.



-- Electronics Plate --

Hood

The hood is designed of 1/8 inch aluminum and mounted with gas struts to open automatically. The hood is secured in the back with a latch, and once the latched is released, the hood will open. The hood can also be taken off by removing the two pins in the front, as well as popping off the gas struts. The hood pivots on the front end of the robot on two quick release pins. It is sturdy enough to hold a large amount of weight, such as a person. The top contains a riser that fences in an area so that items may be strapped down on it for transport. One emergency stop is located on the rear portion of the hood, and the hood is thin enough to drill to add more items to.

Image Processing

Images are acquired for a Videre Design STH-DCSG-C by using MatLab's Image Acquisition toolbox. These Images are then processed in MatLab to determine the relative orientation of the lanes in the course, as well as the location of all obstacles. The controls algorithm uses this information to plan the route though the course. The image-processing hardware is composed of a TS6000 singe board computer running Linux and the Videre Design STH-DCSG-C. The camera is mounted in a way that allows for height and tilt adjustments to maximize coverage of the vehicles image plan. The image processing software module can be broken down into two sub modules lane finder and obstacle detection.

Lane Detection

The lane finder has the purpose of estimating vehicle orientation in the course with respect to the heading direction of the vehicle, as well as the offset between the lane center and the center point of the vehicle's wheelbase. The first step in this process is to acquire the image from the camera, which is done through the Image Acquisition toolbox. Next the image is applied to a filter with products intensity image. This is done to decrease processing time. Finally, the image is passed through the Hough Transform which will detect the lines. For line detection, Hough Transform is the best way to go because it is not affected by noise or lighting conditions. This information is then passed to the control module which determines the best direction for the vehicle to move towards.

Obstacle Detection

The second sub module in the image-processing unit is to detect obstacles and hazards that may obstruct Autonomous Prime path. Obstacle detection is done though the digital camcorder. Once the lane markings are detected, the vehicle's path will be delimited within their boundaries. Thus, obstacle detection is performed only within a Region of Interest (ROI), delimited by the lanes and extending up to a critical distance, (approximately 10 feet). Since all obstacles are to be avoided, identification of the obstacles is not needed; only their detection is important. Obstacle detection uses intensity and color information of al pixels within the ROI to plan the path of the vehicle. The algorithm used is as follows:

1. A small region of the image immediately in front of the vehicle is chosen as a representative of an obstacle free region.
2. The color image in this region is extracted and the covariance of the color information is obtained.
3. An approximation to the eigenvector corresponding to the smallest Eigen value of the covariance (least dispersion measure) is computed.
4. This eigenvector is used as weights to be used to convert the color image to modified intensity image.

By our choice of weights, the intensity of the part of the image that corresponds to the chosen region would be nearly constant while parts of the image that are different would have intensity different from that of the region. If the intensity at any part of the image differs from that of the chosen region by more than three standard deviations of the intensity in the region (this value is easily computed from the covariance matrix), then it is assumed that the difference is due to obstacles. Obstacles are typically detected when they are 10 feet from the vehicle by the vision system.

Direction Control

The lateral control of the vehicle is governed by the location of the lanes and the location of the obstacles. First, a heading direction is determined based on the lanes that are detected. The control is based on a target point determined by where the vehicle should be at a look-ahead distance D as shown in figure 4. Based on straightforward calculations it can be shown that a smooth control can be obtained if the input is obtained by multiplying f, the offset of the desired trajectory at the look-ahead distance D, by a gain. The vision system first computes the offset f using the lanes. If there are any obstacles in front of the vehicle, they act as a repulsive force on the target point and moving it laterally. The magnitude of the repulsive force is made inversely proportional to the distance of the obstacle from the car, and its distance from the desired direction. The image processing program also provides a visual display of the lanes and the target point.

Navigation

Navigation data is acquired for a Garmin 16a GPS and compass. The software Module for Navigation control interfaces though a serial port (RS232) with Matlab. Matlab decodes the GPS signal and applies a Kalman filter to improve accuracy. The compass is then used to determine the heading of the vehicle. The obstacle detection and avoidance system use the same image processing software as stated above. The data obtained from GPS unit is used to send steering control signals to the main microprocessor to navigate from one waypoint to other. When an obstacle is detected, the obstacle avoidance system will take over the control until the obstacle is avoided.

Item	Quantity	Price	Description	Price
Electronics				

Optima Batteries	4	$150.00	2 sets of Optimas Yellow Tops to power motors and electronics separately	$600.00
Micro ITX PC	1	199.00	Micro ITX single board computer	199.00
TS-7200	1	$165.00	TS-7200 Single Board Computer with 32 MB RAM, 16 MB Flash	$165.00
KIT-7200	1	$180.00	Kit with 256 MB CF, Power Supply, Cables, USB-CF Adapter,gcc,etc (TS-7200 not included)	$180.00
TS-SER4	1	$89.00	TS-SER4 PC/104 Peripheral Board provides 4 Serial COM ports	$89.00
TS-12W	1	$32.00	TS-12W 5V Switching Power Supply	$32.00
PS-18VDC-110V	1	$16.00	18VDC wall mounted power supply,110 VAC Input	$16.00
Motors/Electronics				
AX2850	2	$620.00	2 Channel +/- Speed controller up to 40V 120 Amps per channel w/ Opt Encoder inputs	$1,240.00
NPC-T64	4	$286.00	1HP Inline heavy duty 24V motor Geared 20:1, 230 output rpm, 13 lbs.	$1,144.00
Optical Encoders	4	$119.00	High Resolution Optical Encoders for Accurate Speed Control.	$476.00
NPC-PT5306	4	$73.00	14" Solid Rubber Tires	$292.00
NPC-PH448	4	$20.00	Motor to wheel adaptor hub	$80.00
802.11b/g Network				
Access Point	2	$250.00	Outdoor Access Point Capable of Bridging, Point to Point, Point to Mulipoint,etc	$500.00
Omni Antenna's	2	$69.38	2.4GHz OMNI-DIRECTIONAL ANTENNA	$138.76
			TOTAL COST	**5151.76**

Total Man Hours:	**900 Hours**

Faculty Advisor Certification

I hereby certify that the engineering design in the vehicle by the current student team has been significant and equivalent to what might be awarded credit in a senior design course.

N. Natarajan
Associate Professor,
Department of Electrical and Computer Engineering
University of Michigan-Dearborn