YELLOW JACKET II

Cedarville University

Intelligent Ground Vehicle Competition

Design Report

May 26, 2006

Team Members:

Tyler Bixel
Joshua Blackburn
Joseph Eisentrager
Daniel Gallahgher
Timothy Linden
Luke Snider
Matt Zimmerman

Team Advisor:
Timothy R. Tuinstra

Table of Contents:

Team Organization

In order to facilitate the design of this year's *Yellow Jacket II* vehicle, the team members were organized into three teams to tackle the three major design areas. This organization is illustrated below in Figure 1.

Yellow Jacket II
Team Lead: Tim Linden, SR CPE
Advisor: Timothy Tuinstra, M.S.E.E.

Chassis Team

Lead: Tyler Bixel, SR EE

- Luke Snider, JR EE

Responsibilities:

- Chassis
- Drive Train
- Power

GPS Team

Lead: Dan Gallahger, SR EE

- Josh Blackburn, JR EE

Responsibilities:

- GPS integration
- Z-8 Programming

Software, Systems Integration Team

Lead: Tim Linden, SR CPE

- Joseph Eisentrager, JR EE
- Matt Zimmerman, SR CS

Responsibilities:

- Software Design
- Wireless Network

Figure 1 – *Yellow Jacket II* Team organization.

Chassis Team

Design Approach:

While the chassis design from the previous year's robot could have been used, a new chassis was designed for the purpose of making several improvements and changes to the chassis and overall project. One desired change was to make the robot drive from the near the center of the robot rather than the front so that more of the robot could be in the camera's vision. This would also prevent the back end from swinging around and hitting the obstacles after they had gone out of the camera's view. Also some shelving was desired for better organization and accessibility of the electronics and wiring. Furthermore, a new design would be easier to weatherproof than the previous design that had a unique shape to it. Finally, a new design would allow for easier incorporation of an adjustable height mast to improve the camera's versatility.

Robot Chassis

The new chassis in the shape of a box had dimensions of 22 inches high, 22 inches wide, 36 inches long and it was constructed using one inch square stock. To utilize the additional height of the chassis over last year's design, shelves to mount house electronics were placed near the top of the frame. The front shelf housed the Z8 board, the circuit board for switches, voltage regulators, and connectors, the two motor controllers, the wireless router, and the central power hub from the batteries. The rear shelf was a slide out drawer to allow easy access for the notebook computer. An adjustable height post extending from inside the chassis near the drive wheel axis held both the camera (the adjustable height provided more versatility for changing light and terrain conditions) and the GPS reception. To weatherproof the chassis, easily removable plastic panels were affixed on the sides, top, and rear. A 3D rendering of the chassis is shown in Figure 2 while a photograph of the new *Yellow Jacket II* vehicle is shown in Figure 3.

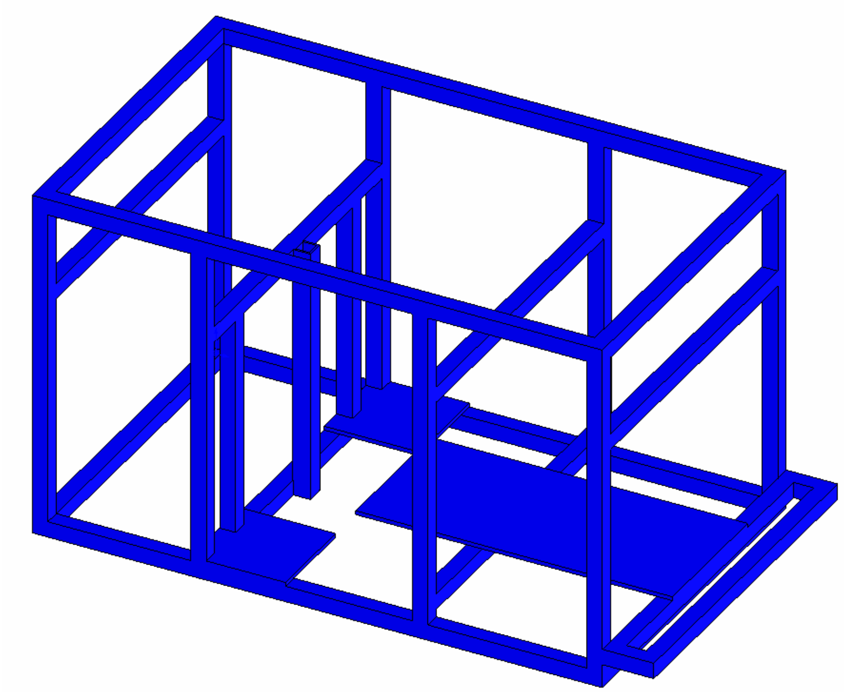

Figure 2 – 3D rendering of new *Yellow Jacket II* chassis.

Figure 3 – *Yellow Jacket II* vehicle.

Drive Train

The drive train consists of two custom fabricated 26-inch bicycle wheels mounted close to the center of the robot. With the rear drive wheels mounted closer to the center of the robot, turns of tighter radii could be made. Each wheel is mounted on a shaft with a gear sprocket which is connected to the motor via a chain so each wheel it is driven independently, allowing the robot to maneuver in tight spaces using differential steering. Two five-inch casters were used at the front corners of the robot instead of only one caster centered at the rear of the robot as in last year’s design. This provided more stability during movement for the robot and better distribution of the weight of the chassis.

Motors and Gearing

We chose ¼ horsepower right angle gear motors from Bodine Electric Company to propel our robot. This motor is shown in Figure 4. These motors have 24VDC windings and draw a maximum of 8.8A at max load. Since the maximum output speed of the gear motors is 125 revolutions per minute, we used a 2:1 gear reduction to couple the motors to the wheels so that the maximum speed of our robot is 4.83 miles per hour. This ratio meets the speed limit of 5 miles per hour established in the competition rules.

Figure 4 – Bodine ¼ horse power motor.

Motor Controllers

The motor controllers used have built-in inrush current limiting and are rated for 20 amps at 50VDC, providing plenty of margin for our 8.8 Amp motors. The motorcontroller that was used is shown in Figure 5. These controllers interface to the Z8 Microcontroller and provide pulse width modulated speed control of the motors.

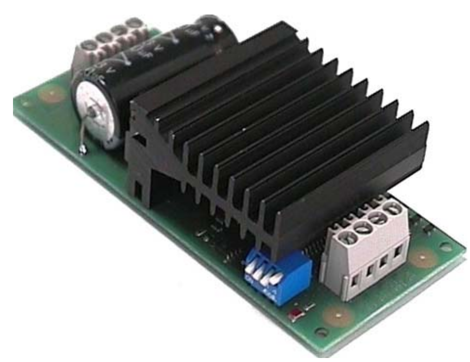

Figure 5 – Motor Controller.

Power and Electronics

Two compact 12VDC 12AH sealed lead acid rechargeable batteries from Power Sonic were chosen to power the robot. One of these is shown in Figure 6.

Figure 6 – Batteries used to power *Yellow Jacket II.*

An additional pair of batteries was purchased so that the robot could be operated and recharging of the batteries could occur simultaneously. These batteries were connected in series to provide 24VDC for the motors. A 12VDC line was pulled from one battery to power the wireless router. This 12 VDC line was also inputted into a LM317 voltage regulator chip along with three resistors for biasing, to provide a 9 volt potential for powering the Z8 board. The Z8 board had an additional voltage regulator that provided a 5 volt potential to power the Garmin GPS receiver. To enhance the safety of the robot and meet contest requirements, a manual emergency stop push button from Mouser Electronics that required a quarter twist for reactivation was connected in series with the 24 VDC line running to the motors at the rear of the robot. With this arrangement, the 24 VDC motors could be shut off independently of the electronics. Furthermore, this quality emergency stop button from Mouser was much more secure that the previous year's home-made stop button.

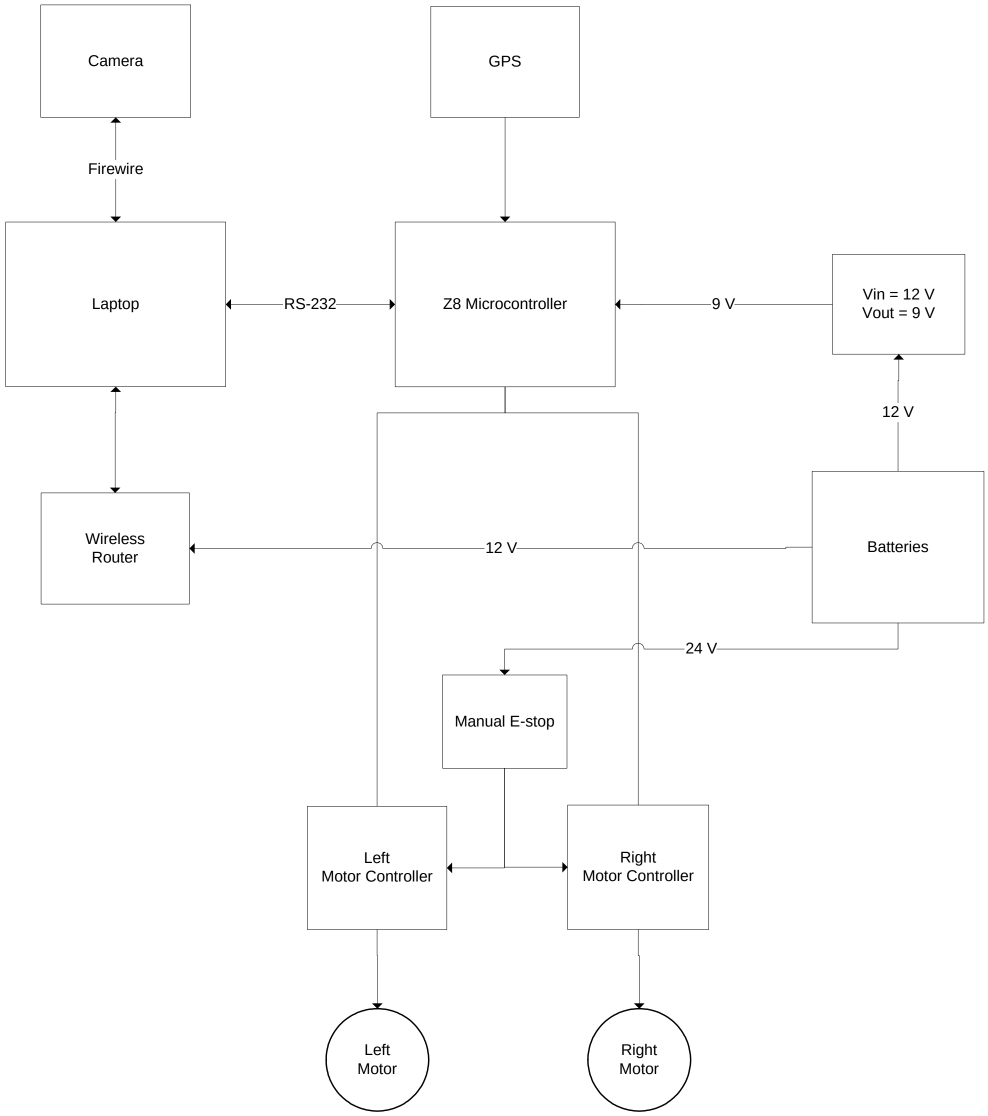

Figure 7 - Robot Electronic System Diagram.

GPS Team

A system block diagram for the *Yellow Jacket II* electronics system is shown in Figure 7. The Z8 microcontroller functions as a multiplexer for the GPS and motor controller signals. Using the Z8 as a multiplexer was necessary for two reasons. First, the computer only contains one serial port. The GPS and motor controllers required a total of three serial ports. Therefore, the computer can send different commands through the single connection with the Z8 and control all three devices. Second, the Z8 contained timers that were already set up for pulse width modulation (PWM) communication with the motor controllers. This allowed the computer to send a single command to the Z8 for each motor controller instead of being required to send a pulse every 20 ms. For ease of development, we choose to use the Z8 microcontroller development kit. This trainer board already included the CMOS to RS-232 conversion chips; however, it still gave us access to the timer pins needed for the motor controllers.

The Z8 connected to the computer and the GPS through the two included UARTs. An interrupt structure was established so that whenever the computer or GPS had or was sending information, the Z8 would run a particular interrupt service routine. Both connections were bidirectional such that commands could be sent from the computer, the GPS could be initialized, and GPS data could travel from the GPS to the Z8 to the computer. The motor controllers were controlled by PWM through the included timers on the Z8. In order to communicate through PWM, the computer sent the Z8 a one byte number. This number is scaled and offset by the Z8 such that the lowest value produces a 1.1 ms pulse and the highest value produces 1.9 ms pulse, which correspond to full speed backward and forward. A value of 7F produces a 1.5 ms pulse, which corresponds to the stopped position.

Software and Systems Integration Team

Processing Hardware

A standard 802.11b/g wireless router was installed on the robot to allow the JAUS communication protocol to be implemented. This wireless network also allows the robot to be remote controlled from a 802.11b enabled PDA. In this design the PDA is used to gather telemetry from the robot while in autonomous mode, allow for remote manual control when not in autonomous mode, and implement a wireless E-Stop. This is a great improvement over the previous method of manually controlling the robot via the keyboard on the laptop computer.

A Dell Latitude C800 laptop computer was is used to process the images acquired from the Firewire camera, interface with the robot and communicate with the wireless network. This laptop computer is running on a 1GHz Intel Pentium III processor with 512MB of memory. Although this laptop is not as powerful as current laptop computers it has proven to be sufficient for the algorithm which has been implemented. To improve the speed of the image processing several lookup tables were generated to significantly reduce the amount of floating point computations necessary to complete a given task.

The Linux operating system was chosen because of the open source nature of the software libraries that are used to access the hardware and the ability to customize the kernel to improve the performance of the system. Open source libraries were used to access the hardware interface to the Firewire camera and the display. Taking advantage of the library to access the camera it was possible to operate the camera in a DMA mode to allow the image processing algorithm to continue to process while a new image is being fetched. This method would allow the image processing software to potentially operate at the full camera speed if it were to be run on a newer laptop computer.

Image Processing

The image processing is the core of the design allowing the robot to make decisions of where it must go to properly avoid obstacles and stay within the course boundaries. The camera is the only sensor on the vehicle making this design unique compared to those that use sonars and laser range finders. A new algorithm was developed this year that is based upon edge detection which provides a new set of challenges that must be overcome to properly solve problem of autonomous navigation.

The algorithm for this year made use of edge detection to identify items of interest within the image data. Using the edge detection introduced the problem of detecting the traps that are present in the course as obstacles rather than as traversable terrain. This problem is solved by using color segmentation the foundational concept used in last year's Yellow Jacket entry. The basic concept for this method is to convert the image into the Hue, Saturation, and Intensity (HSI) color format. This allows the algorithm to ignore the intensity while retaining color information. By taking pixel samples of the tarp a color mask is created specifically matching the tarp allow any pixel that maps into this region of the H-S color plane to removed. This method does require that a predetermined color mask be set that determines any color set that is defined as traversable terrain.

Once the tarp and grass regions have been removed from the image an edge from the tarp would ideally no longer exist allowing edge detection to be used to find the path which the robot should follow to successfully navigate a given scene. The Sobel edge detection method was used to find the edges of lines and obstacles in the image. The result of the Sobel gradient operation is thresholded to give a resulting binary image that can be processed more easily.

This resulting binary image contains a significant amount of noise that must be removed in order to obtain an accurate heading. To eliminate as much noise as possible in the image regions of connected pixels are found. A pixel is determined to be connected to another if any of its eight neighbors is turned on. Using this method the number of connected pixels is found allowing the region to be removed from the image if it does not contain a large enough number pixels.

Once the noise has been removed from the image a radial scan can be performed. This gives the robot a perspective on all the possible new headings that it can decide to choose. This scan is converted into plot of the distance to the obstacle vs. the angle of the heading. From this data a heading can be determined from finding the centroid of this plot. The centroid is then converted into rectangular space and used to determine the speed of each wheel.

In order to successfully navigate the course the computer needs to ultimately determine what the speed of each wheel must be. The calculated centroid is used to determine the speed of each wheel. There are two parameters that mush be determined, the first of which is the speed of travel the second is the amount of turn. First speed is determined by the distance the centroid is from the robot while the second is determined from the ratio of the x distance from the center of the image and the y distance from the robot. This allows the robot to become more responsive to obstacles that are close while remaining less responsive to obstacles in the distance.

This new algorithm developed for this years entry of Yellow Jacket II is a significant improvement over the previous algorithm used in the Yellow Jacket entry. Currently the algorithm is running at 10 frames per second allowing the speed of the vehicle to be significantly increased compared to the previous entries 3 to 4 frames per second. This improved speed was achieved by using lookup tables for the RGB to HSI conversion as well as eliminating both the Hough transform and median filtering. The Hough transform was replaced with the edge detection and the median filtering was replaced by filtering edges based upon the size of the connected region. Figures 8 and 9 show examples of the image processing algorithm applied to images containing a “sand trap” and barrel obstacles respectively.

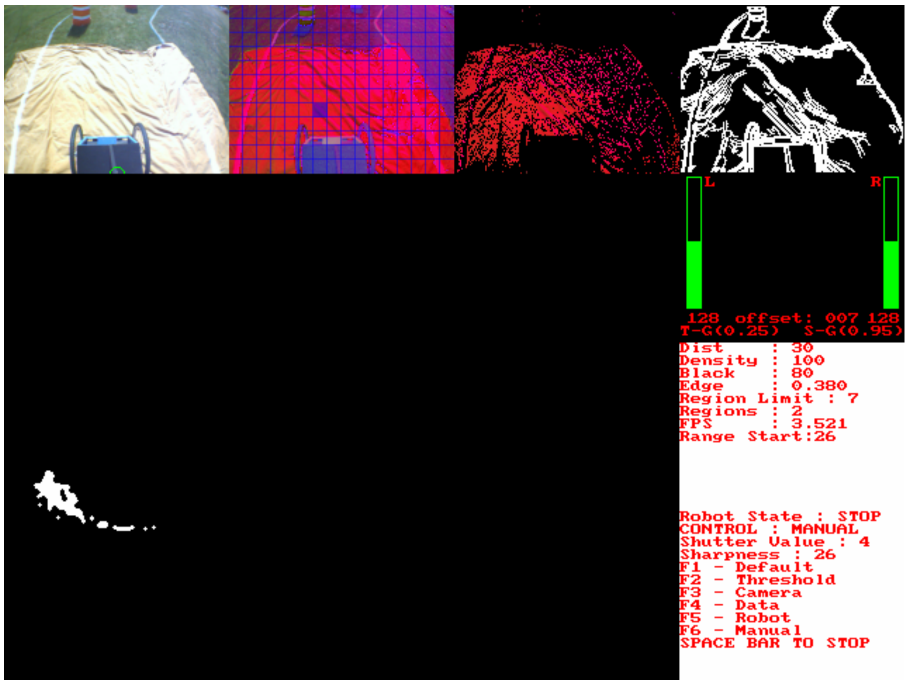

Figure 8 – Screen shots from the image processing module showing the original image, HSI data, edge detection as well as the settings for various user selecteable parameters of importance to the algorithm.

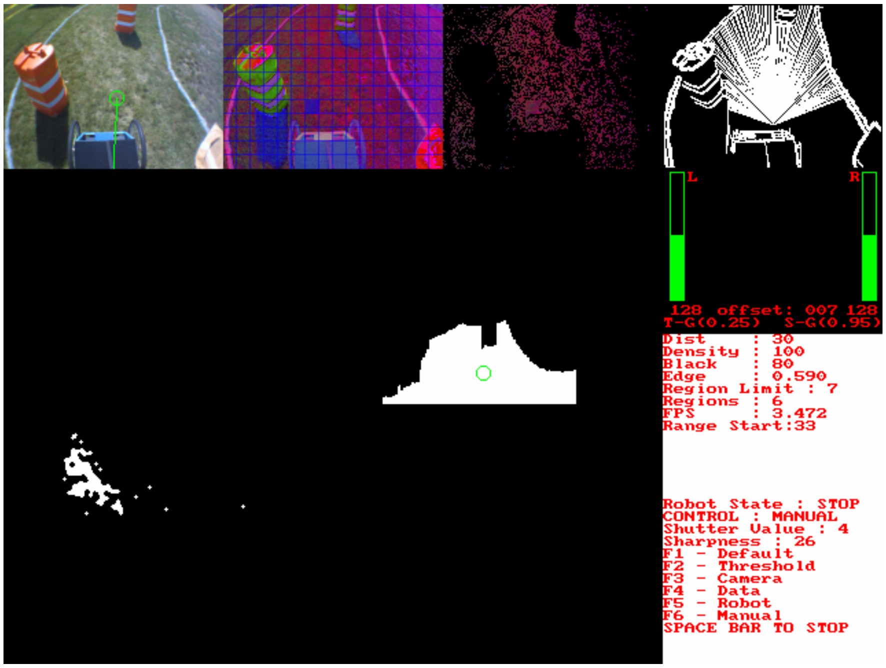

Figure 9 – A second screen shot from the image processing software.

Vehicle Cost

Item	**Unit Cost**	**Total Cost**
Gear-motors*	$450	$900
Motor Controllers	$130	$260
Chassis Materials	$75	$75
Bicycle Wheels	$100	$200
Casters	$15	$30
Fire-I Firewire Camera	$83	$83
Flex 10k FPGA	$150	$150
Electronics Interface Board	$40	$40
Emergency Stop Button	$29	$29
Laptop Computer	$900	$900
GPS Unit**	$200	$200
Ipaq	$300	$300
Wireless Access Point	$60	$60
Batteries	$23	$46
Misc. Parts		$250
Total Cost:		$3523

*Donated by Bodine

**Donated by Garmin

Conclusion

Cedarville engineering students have in many ways totally redesigned this year's *Yellow Jacket II* autonomous vehicle. The chassis has been completely redesigned allowing for better turn performance as well as additional room for electronics. Global positioning capability has been added for entry into the navigation challenge, something new for the Cedarville team. New concepts have also been developed including the ability to provide remote control as well as telemetry using a wireless network. These improvements should make *Yellow Jacket II* a competetive part of this year's IGVC competition.

Faculty Statement:

I Timothy R. Tuinstra certify that the level of effort expended by this seven member team of undergraduate students from Cedarville University has been significant and has been on the level of the effort required for a typical senior design competition.

Timothy R. Tuinstra, M.S.E.E.
Assistant Professor of Electrical Engineering
May 24, 2006