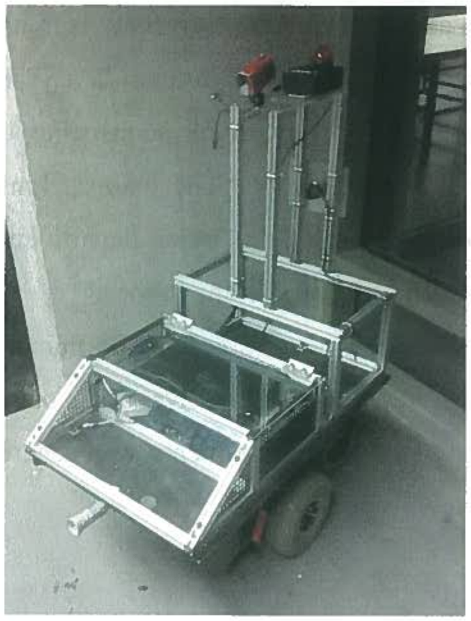
JOHNS CONNOR



2012 Intelligent Ground Vehicle Competition

Department of Electrical and Computer Engineering Design Team
The Johns Hopkins University
Baltimore, Maryland

Advanced Team Members

Aidan Fowler (Junior)
Adam Gross (Junior) – Team Leader
Alex Hernandez (Junior)
Nathan Ching Lin (Junior)
Michael McEleney (Junior)
Daniel Naito (Junior)
Tom Prats (Junior)

Underclassmen Team Members

Fawaz Ahmed (Senior)
Ryan Cropp (Freshman)
Bill Kim (Freshman)
Timothy Ng (Freshman)
Mitchell Sacks (Sophomore)
Alexander Simonelli (Freshman)

I, Jin U. Kang, certify that the design and engineering of the Johns Connor vehicle by the registered student team listed above has been significant and equivalent to what might be awarded credit in a senior design course.

[illegible]

Jin U. Kang, Ph.D., Team Advisor
Professor and Chair
Department of Electrical and Computer Engineering
The Johns Hopkins University

THE JOHNS HOPKINS UNIVERSITY BALTIMORE 1876 VERITAS VOS LIBERABIT

Introduction

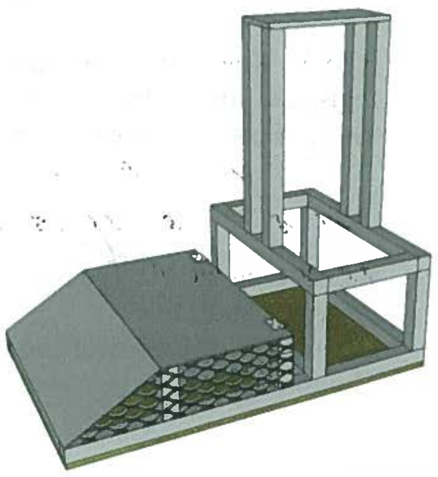
The IGVC design team at Johns Hopkins University is in its third year, making its second appearance at the annual competition. The design team is a year-long for-credit course in the Electrical and Computer Engineering Department, consisting of electrical engineering and computer engineering undergraduate students. The team is limited to a budget of $500 per semester and may not include members from other departments. This year's team put in approximately 700 man-hours while working on the robot.

Improvements from the previous year's design are numerous, with the most consistent aspect being the chassis. This year, a new GPS receiver and a new motor controller made it onto the robot, and all of the wiring between components was secured properly. In addition, the decision controls were completely re-written.

Hardware

Component List

- Jazzy 1113 ATS motorized wheelchair
- Roboteq AX3500 motor controller
- Sony Mobile HD Snap Camera MHS-CM1
- FV-M8 32 channel San Jose Navigation GPS receiver
- SparkFun GPS Evaulation Board
- Honeywell HMC6352 digital compass
- MaxBotix LV-MaxSonar-EZ0 High Performance Sonar Range Finder
- MaxBotix XL-MaxSonar-WR1 Weather Resistant Sonar Range Finder
- National Instruments USB 6211 DAQ
- BHPC series Beehive Marker Lamp (safety light)
- Current Logic 24V to 12V step-down
- Current Logic 24V to 9V step-down
- Seed Studio Works 315 MHz remote relay switch
- Dell Vostro 1520 laptop computer
- DC laptop power adapter
- National Instruments LabVIEW software

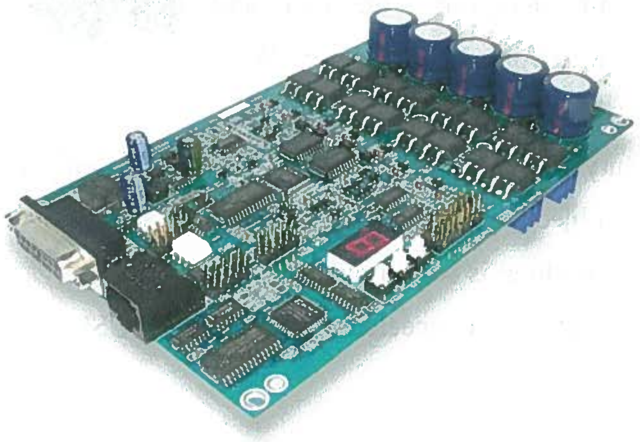


Johns Connor is built upon the base of a motorized wheelchair, the Jazzy 1113 ATS. The seat was removed and replaced with an aluminum frame mounted on a plywood board. The aluminum frame was designed with Google SketchUp and constructed with pre-cut struts of 80/20 10 Series aluminum.

The major components are described in detail below in the following sections.

Motor Controller

The Johns Connor's motor controller, purchased this year, is a Roboteq AX350. It replaces the stock motor controller for the Jazzy wheelchair, which was previously controlled by a tapping into the joystick. Additionally, in order to control the former motor controller, exact voltages needed to be passed out through the DAQ, which then correlated to a voltage on the motors. This was prone to slight fluctuations, due to elements that would affect resistances in the wire, and gave a much greater chance of a misinterpreted voltage, however slight.



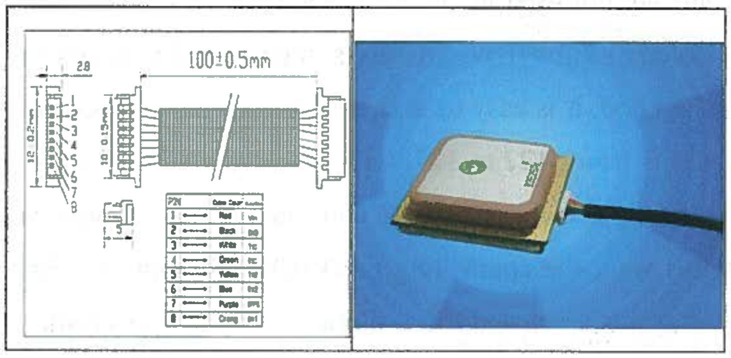
With the robot's current motor controller, LabVIEW passes out hexadecimal strings through a serial port, meaning that it operates with binary bits, a much more dependable method. The motor controller patches out signals for both the right and left motor controller, and there are 255 options for each. There is a 25-pin serial connection, which means 8 pins for each of the channels. The additional pins are not used in any significant way by Johns Connor. The motor controller program, implemented in LabVIEW, interprets integers and converts them to voltages on the motors. Though this method, it is easy to anticipate the speed at which Johns Connor will move. The range of integers is from -127 to 127, with negative values causing the motors to spin backwards. This allows simple and convenient conversion to an approximate percentage of a maximum speed. In this way, the controlling LabVIEW program can simply send integers into the function developed at this level, and cause motion, without ever needing to know how to convert values.

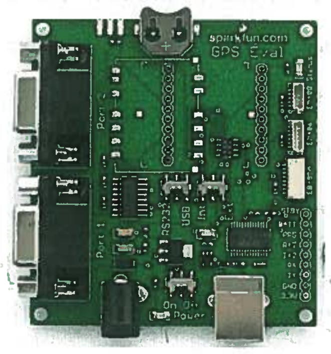
The motor controller VI functions as the innermost black box, and is where all data computation finally ends, and what all the other functions (LabVIEW sub-VIs) are driving into. Additionally, the motor controller sub-VI also serves to inform the safety light as to whether or not it is in autonomous mode. If it isn't in autonomous mode, which in terms of Johns programming is equivalent to the motor controller not being on, then the safety light will be solid. However, once the motor controller turns on, the light will begin to blink. Additionally, the emergency stop buttons (wireless and otherwise) turn off the motor controller, and kill power to the motors (leaving the safety light and other features on, but making sure the robot is incapable of moving.)

Sonar

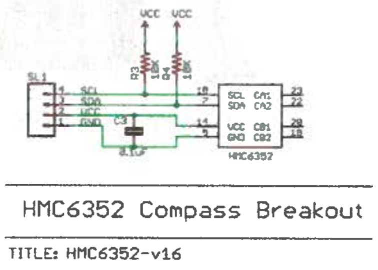
The baseline object avoidance technique is to use sonar range finders to detect when the robot is too close to something. Three range finders are mounted underneath the plywood base on the front side of the robot, one in the center and one on each corner. The center range finder is a MaxBotix XL-MaxSonar-WR1 and the two on the corners are MaxBotix LV-MaxSonar-EZ0 range finders. The range of both range finders is greater than twenty feet. The output voltage is proportional to the proximity of the detected object; a low voltage means something is very close and a high voltage means nothing is within range of the sonar. The sonar output voltages go into the DAQ, which is connected to the laptop via USB.

GPS and Digital Compass





Johns Connor uses the FV-M8 32 channel San Jose Navigation GPS receiver, which supports WAAS. It is accurate to 3.3 meters and is used at a baud rate of 38400 bps. The GPS is connected to the GPS Evaluation Board from SparkFun Electronics. This evaluation board is specifically designed to work with our GPS. It interfaces to the computer through a USB port. The board converts serial data to NMEA sentences.



Johns Connors uses a HMC6352 digital compass. It is supplied with 3.3 V and has a .5 degree heading resolution. The compass is connected to the computer through an Arduino Uno microcontroller.

LabVIEW is used to receive and process the data from the GPS and compass. The Arduino is programmed to output the compass values as a degree measurement. The GPS coordinates come into LabVIEW as NMEA sentences. The software processes the NMEA sentences to collect the current latitude and longitude. The current latitude and longitude and desired latitude and longitude that are pre determined are used to calculate the distance to the desired waypoint and the angle between the current and desired locations. This angle is compared with the current angle Johns Connors is facing to determine the difference between the current heading and the necessary heading. The necessary change determined by the difference of these angles and the distance between the current and desired location are used in our decision control.

Vision

The main component is the Sony MHS-CM1 video camera that has a video resolution of 1440 x 1080 pixels. LabVIEW is used to take in streaming video from the camera and convert them into frames of 640 x 480 pixel images. The goal for vision is to detect a clear grassy path that corresponds to the direction of the GPS. Signal processing involves using a color threshold function that highlights the color of grass and blacks out everything else. This is done by applying a threshold to the three planes of a color image in RGB mode and changing the pixel values of red, green, and blue. Pictures of grass were taken using the camera, and they were used to determine the pixel values of grass. The resulting image after color threshold function is a

binary 8-bit image. There is often noise in the resulting binary image, so Fast Fourier Filter in low-pass mode is used to truncate rough edges.

Power

The original wheelchair runs on two 12V batteries for a total 24V system. Since our components require a variety of voltages, that 24V was stepped down to lower voltages. We purchased two voltage converters to get 12V and 9V, and built the third 5V converter using simple circuit components.

The motor controller uses the full 24V to control the wheelchair motors. It is regulated internally by the digital signals as described above. The sonar range finders use 5V from the custom circuit. The wireless emergency stop receiver and safety light use 12V for power.

To prevent the laptop's and video camera's batteries from draining (the laptop itself powers the DAQ, Arduino, GPS, and digital compass), they too draw power from the wheelchair's batteries. A cigarette lighter car charger is used for the laptop, drawing 12V from the voltage converter, and then converting to the laptop's voltage internally. The video camera is connected directly to the 9V voltage converter. These are meant to be backup trickle chargers only, so there is a separate power switch for charging these two devices, in order to prevent the wheelchair batteries from draining too quickly.

As per competition rules, Johns Connor contains two emergency stop buttons, one on the back of the robot and one as a wireless remote. Both cut power to the 24V line going into the motor controller, which stops the motors and therefore stops the robot's movement.

Decision Control

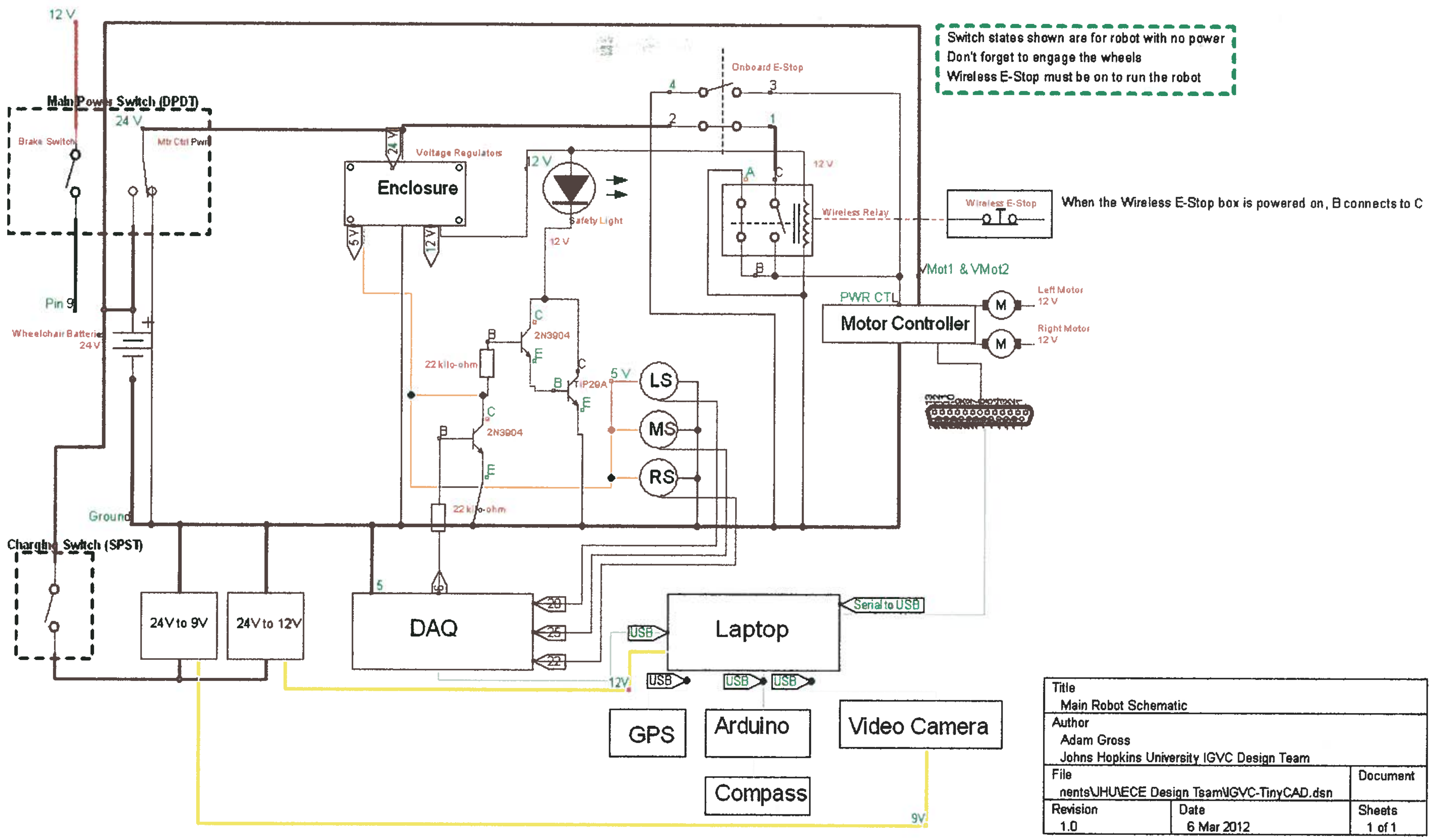
There are three major components to the decision making for Johns Connor. They are centered around the three main clusters of input information: the sonar range finders, the video camera, and the GPS and digital compass. The three work in tandem to aid the robot in making the best possible decision at any given phase of operation. The methods employed in those decisions, all implemented in LabVIEW, are described in the following paragraphs.

The basis for all decision making is computing an angle from the compass and GPS. By using its current GPS location and the destination one, the program is able to compute an objective angle that it should be traveling along the map at (i.e. 0 degrees is east, 90 north, 180 west, 270 south). Then, by using the compass, a difference is computed between the angle the robot should be moving in, and the angle it actually is moving in. That angle is then used to compute a rate of turning constant that is applied to a constant to determine an integer value for each of the motors. Johns Connor will try to turn in the direction that would get it to the appropriate angle fastest, and the rate at which it turns varies based on how far it is from the correct angle. If there is no turning required, then Johns Connor will go straight.

The sonar range finders, while the simplest form of input, and perhaps the most dependable form, are, in an ideal case never used. Their functionality serves only as a backup, a method for avoiding objects that are immediately in front of the robot. There is a distance threshold before their decision making algorithm takes over the other one, and it does override all other decision making algorithms. If an object is about to be hit, avoiding it becomes the top priority for the robot. Decision making for the sonar input is divided into five possible states, and depending on which one the robot perceives, a different decision will be made. The states are divided based on which thresholds have been crossed. The left, right, and center alone, the right and center and the left and center. The condition where all three are eclipsed mirrors what will happen when only the center has had its threshold passed. Then, based on these states, a decision is made. For example, if the left's threshold is crossed, it will turn slightly to the right, but if the left and center are both crossed, it will turn a bit harder to the right, since it knows that it has to turn further to avoid the objects. The same is mirrored for the right side. The only special case is the middle or all three case, where the turning is determined by which direction the robot is supposed to turn (as determined by the GPS and compass), then a harder turn is computed. In this way, the sonar range finders function as a last line of defense to protect Johns Connor from losing time.

The vision algorithm however is much more involved, but it ultimately handles most of the object and line avoidance, and is vital for Johns Connor's effective operation. The vision algorithm is as follows: take the current image from the camera, and perform on it the necessary operations to perform a green threshold on it. Green is selected to create a safe zone in the image, the safe zone being the grass that is safe to traverse. The vision then overlays an imaginary line on top of the input image, calculated by the current desired angle change (as produced by the GPS and compass starting at the bottom center of the robot). That line is then used to determine whether or not the robot can safely proceed along that path, by checking to see if there is enough green space along that line. If it ensures that there is, it will proceed along that path, and if it does not, then it will attempt to use the same algorithm for both five degrees to the left and five degrees to the right. Once it finds the closest possible angle to the desired angle, Johns Connor will then begin to turn towards that angle. In this manner, the robot will avoid all non-grass objects, including the white lines at the edges of the course.

In using all three of these devices together, Johns is able to make intelligent decisions about where it is safe to travel, while keeping the end goal in mind for all the decisions it makes. By using the GPS and digital compass to govern the commands of the camera and sonar range finders, it can be ensured that it will avoid objects in a manner that keeps it moving towards the waypoints, while still actively avoiding contact with any of the objects in its path.



Title Main Robot Schematic		
Author Adam Gross Johns Hopkins University IGVC Design Team		
File nents\JHU\ECE Design Team\IGVC-TinyCAD.dsn		Document
Revision 1.0	Date 6 Mar 2012	Sheets 1 of 1