Woody the Firefighting Robot

By
Timothy Martin

University of Florida
Department of Electrical and Computer Engineering
EEL 5666
Intelligent Machines Design Laboratory

TAs : Mike Pridgen
Thomas Vermeer
Instructors: Dr. A. Antonio Arroyo
Dr. Eric M. Schwartz

Table of Contents

Abstract

Woody the firefighting robot is designed with entertainment in mind. Woody is not so much useful as he is fun to watch. Woody's task is to simply put out a flame, but he does this and so much more. Woody uses a special IR turret to perform advanced obstacle avoidance; the drive motors utilize a PD controller for rpm control; and Woody uses a separate audio board for adding funny move quotes and music. Together these systems make for an interesting and entertaining robot performance.

Executive Summary

Woody has proven a unique and successful robot. Unlike a most IMDL robots Woody uses more precise actuation to perform his task of extinguishing the oil lamp. Woody was designed using stepper motors and non-hobby dc servo motors for accurate and precise control of sensors and mobility. Woody also utilizes a Nintendo Wii Remote which is not a typical sensor used in the class. Last Woody has the ability to play audio files, which enhance his performance

Woody's main sensory is mounted on stepper motors. These motors are designed to be moved to exact positions. This precise movement enables Woody's scanning IR sensor turret, and the Wii Remote turret to work.

Woody's drive system is also quite advanced. The DC motors have home made encoders converting the motors to servo motors. The encoders are used for a PD feedback RPM controller. This controller makes Woody very smooth and precise in his movements. Woody needs precise control because his extinguishing ability is limited. If the extinguishing speaker is off by a few degrees the oil lamp flame will not go out.

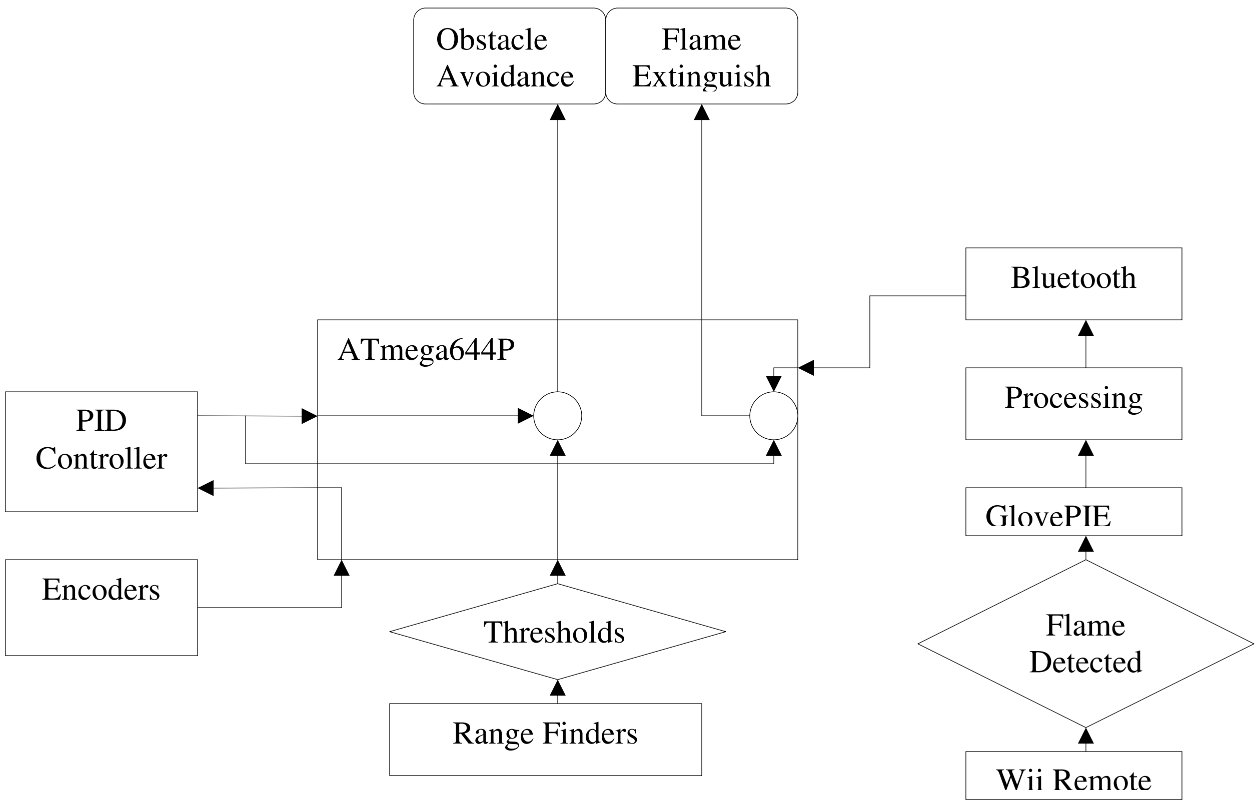
Woody's movement must be precise in order to put out the flame; likewise his sensory must be equally precise and accurate. The Nintendo Wii Remote fits the bill. The Wii Remote is an interesting and powerful sensor with only a $40 price tag. The Wii Remote has an IR camera with a resolution of 1024 by 768 pixels. This camera is used to track up to four IR hotspots. Woody uses the Remote to track the flame of the Oil lamp.

Woody's most entertaining aspect is his ability to play audio. Woody has an audio board that can play any wave file converted to the correct format. This board is used to play move quotes and music. These audio clips greatly increase Woody's appeal to onlookers. Everyone seems to enjoy the quote from Dirty Harry "Go ahead, make my day", just before Woody fires.

Introduction

This report will cover the intricacies of Woody including but not limited to the platform, sensory, and behaviors. This report will show how Woody accomplishes his task of seeking out an oil lamp flame, and then extinguishing it.

Integrated system

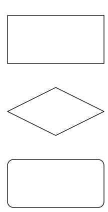


Key

System =

Event =

Process =



Mobile platform

Frame description:

Woody's frame is circular in shape and 16" in diameter. The round shape prevents woody from snagging on any obstacles that are not avoided. The frame has 2 slits for the drive wheels to allow it to sit close to the floor. This feature is crucial for the range-finding sensors. All of these sensors are mounted below the frame. The reason is simply because some obstacles are quite short, yet not traversable.

Figure 1 Undercarriage

Drive train:

Woody uses two dc servo motors for a differential drive system (figure 1). The frame is also supported by a ball transfer caster displayed at the top of figure 1. The servo motors attach to the drive shafts with universal joints to nullify misalignment from mounting (figure 2). The drive shafts of the motors are mounted with two ball bearings. There is one bearing on either side of the drive wheels. The bearings eliminate bending moments to the servo motors.

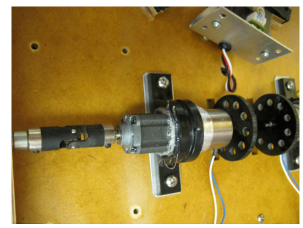
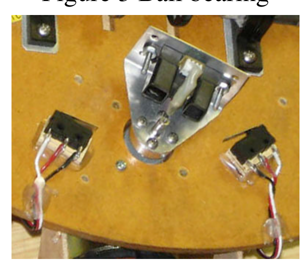


Figure 2 motor

Figure 3 Ball bearing

Sharp IR turret

The Sharp IR turret, Figure 4, is one of the more unique mechanical systems on woody. This turret strafes back and forth between two limit switches while collecting range data. This sensor turret would be equivalent to an array of 56 or so individual Sharp IRs, because one pass of the turret equates to 56 data points with exact angle displacements. The turret is the main sensor used for obstacle avoidance. This turret can also identify objects by detecting their edges; however this method has proven a tad inaccurate.



Wii remote turret

The Wii remote turret, Figure 5, is similar to the Sharp IR turret because they both use a stepper motor for positioning, but the Wii remote turret is able to continuously rotate. This is possible because the Wii remote communicates over Bluetooth, and therefore has no wires to inhibit constant rotation. This turret spins the remote around while the robot avoids obstacles.

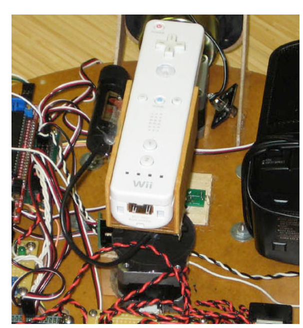


Figure 5 Wii turret

Actuation

DC Servo Motors

Woody's drive system utilizes 2 surplus Namiki dc gearmotors, Figure 6, that I converted to servo motors. The gear ratio is 80:1, the rpm is 144 at 15VDC, the current draw is ~ 500 mA at stall and 100 mA no load. The motors were converted to servo motors by adding codewheels to the backshaft of the motors. The codewheels, Figure 7, have 10 holes, which equates to 800 pulses per revolution of the output shaft when using a rising edge interrupt, and 1600 with a change interrupt. The codewheels are coupled with photo interrupters to create incremental encoders for under $4. The commercial equivalent to this costs $30 with a resolution of 16000 pulses per revolution, which in this application is just excessive.

Figure 6 Namiki motors

The Namiki motors are driven using a single SN754410 1 amp dual H-bridge motor driver, Figure 8. The motors are controlled using PWM on the ATmega644P. The PWM frequency is 64 kHz to eliminate audible switching of the H-bridge transistors. The Atmega644P uses a PD controller with feedback from the homemade encoders to control the motors.

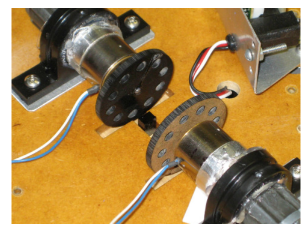


Figure 7 Codewheels

Stepper Motors

Stepper motor theory is extensive and reaches beyond the scope of this project. For this reason this report will focus specifically on the methods used.

The stepper motors used in Woody are two phase bipolar. This means that to drive these motors, a dual H-bridge is required. The steppers have 4 wires, two per phase. The stepping methods used on this robot are half and full stepping, with a chopper.

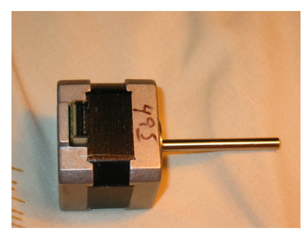


Figure 8 Stepper Motor

Half and full stepping refers to the resolution of the stepper. In this project the steppers used were Nema 17, two phase, bipolar steppers (Figure 8). Their resolution is 200 steps per revolution, and their current rating is 40 mA per phase (80 mA total). Full stepping refers to the method of stepping that uses the true resolution of the stepper. This is achieved by pulsing the four leads in a four part sequence. In this form of stepping both phases of the stepper are always energized. Half stepping is an eight step process which alternates between energizing both and a single phase of the stepper. This method achieves double the resolution with less torque.

The most important part of modern stepper motor control though lies in the chopper circuit. It is worth noting that the specs listed above did not list a voltage rating, this is because the voltage input to a stepper is not to important when a chopper is used. The chopper is a circuit that monitors and adjusts the instantaneous current through the stepper motor. The point of the chopper is to make a stepper faster and more efficient. The stepper is an inductive load, and consequently there are transients in between steps. When higher voltages are used these transients begin to decrease. The chopper circuit allows for high voltages by limiting the current through the stepper motor. This is necessary because V=IZ. If the voltage increases and the impedance stays the same the current must increase unless it goes through a chopper.

The stepper motors on woody perform high speed, and accurate positioning for the Wii remote and the GP2Y0A02YK long range Sharp IR. The motors can achieve higher speed and accuracy than hobby servos, and they can rotate continuously. The steppers do not need full encoders like servo motors because their positioning system is open loop. [12]

Fire extinguishing speaker

Woody uses a 4Ω 5W speaker to extinguish flames. A tin can with a hole is attached to the front of the speaker to funnel air flow, Figure 9. This allows the speaker to generate a powerful enough blast of air to extinguish a flame up to about 12 inches away. The speaker is driven by a single NPN Darlington transistor and an adjustable voltage divider circuit. The robot is often shop tested with a 24V supply compared to the ~15V battery supply voltage

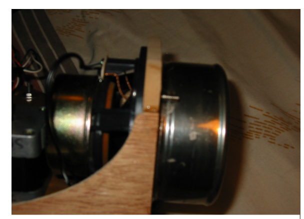


Figure 9 speaker mount

General purpose speaker and wave shield

The general purpose speaker, Figure 10, is used in conjunction with the wave shield attached to an arduino, a microcontroller board based on the ATmega328. The wave shield uses SPI (Serial Peripheral Interface) for communicating with an SD card that has WAV audio files saved in a FAT16 format. The shield can play up to 512 files of any length [11].

Figure 10 General purpose speaker with built in audio amplifier

Sensors

Range finders

Sharp IR GP2Y0A02YK

Figure 11 [7]

The Sharp IR GP2Y0A02YK (Figure 11) sensor outputs an analog voltage that corresponds to a distance measurement. However it is fairly inaccurate. The tolerance is ±10 cm [1]. For this reason in software this sensor is interpreted digitally by a threshold. This means an obstacle is either in the way or it isn't. The real benefit of this sensor is its thin beam. When the sensor is panned back and forth an angular position of an obstacle is easily obtained.

Sharp IR GP2D12

Figure 12 [8]

The Sharp IR GP2D12 is the same sensor as the GP2Y0A02YK except for a different lens [1]. Two of these rangefinders are facing sideways near the rear of the robot. These sensors are used to indicate whether an obstacle will be encountered while turning. If an obstacle is in the way the robot will drive in reverse to avoid it.

Parallax Ping Sonar

Figure 13 [9]

The Parallax sonars are very accurate sensors. The distance measurements appear to be within the accuracy of about an inch. These sensors are used when precise distance measurements are useful, and when a wide angle sensor is beneficial. The front sonar is used for positioning Woody about 7cm from the oil lamp. The wide angle of the sensor allows for accurate measurements, even when the robot is not directly pointed at the oil lamp. The rear sonar mainly takes advantage of the wide angle of the sensor because this sensor is non-scanning.

Motion control

Limit Switches

Figure 14 [10]

The limit switches on Woody serve are used to find the extremes of the IR turrets angular range. This is useful for centering of the turret, and they act as a failsafe if the stepper loses its position.

Encoders

Figure 15

The encoders are very simple yet powerful sensors on Woody. These sensors drive external interrupts on Woody's MCU, in order to calculate motor RPM. This data is then in turn used for the on board PD controller. These sensors were home made yet they work excellently. There is also another single pulse reference encoder placed on the Wii remote turret. This encoder is only for centering.

Flame detection

Nintendo Wii Remote

Figure 16 [13]

The Nintendo Wii Remote is the most import sensor on Woody. The Wii Remote has a IR camera on board that keeps track of up to 4 blobs. This feature is used to determine whether there is a flame visible and where it is exactly in reference to Woody. A program called GlovePIE is used to receive the data from the Wii remote. This program has its own scripting language, and it is intended mostly for mapping out computer game controls. GlovePIE converts the Wii Remote's IR blob coordinates to a virtual joystick. Once this is done, a Java based IDE called processing converts the joystick values of from -1 to 1 to 0 to 200. This data is sent over Bluetooth to Woody's MCU.

Behaviors

Initializations

Stepper motor centering

Each time Woody is booted up he first initializes his scanning sensors. The IR turret measures the steps in between its limit switches then centers itself in between the two extremes. The Wii Remote turret spins around until its reference detects center. If either of these initializations did not occur Woody would not work.

PD controller initialization

Once the steppers are centered the PD controllers are initialized. This function is actually more of a refresh than an initialization. The controller needs to be reset each time there is a direction change of the robot

Low level

PD controllers

The PD controllers use feedback from the wheel encoders in order to maintain nearly constant rpm regardless of terrain. This allows Woody move in a relatively straight line. The original controller was a meant to be PID, but the integral term caused the system to go unbalanced when disturbances were incurred.

High Level

Obstacle avoidance

Woody has four dedicated obstacle avoidance sensors. One long range Sharp IR, two short range Sharp IRs, and a Parallax Ping sonar. The long range Sharp IR is the main avoidance sensor. It is mounted to the IR turret. The turret detects obstacles over a ~100 degree span. The rear IR determines if any obstacles will be encountered during turns. Woody does not turn in place so his tail is swung back and forth. The rear sonar prevents Woody from backing into obstacles.

Obstacle Avoidance Table

Obstacle locations	Front left	Front right	Front	Front left and back right	Front right and back left	Back
Behavioral Reaction	Move right	Move left	Move in the previous direction	Move backwards	Move backwards	Move in the opposite of the previous direction

Flame detection

When Woody is in flame detection mode, he tracks the flame with the Wii remote turret while avoiding obstacles. This behavior is a simple demonstration of Woody performing many behaviors at once. After 20 seconds, if Woody can see the flame, he goes into flame extinguishing mode.

Flame extinguishing

When Woody enters the flame extinguishing mode, he first centers the Wii turret, then he spins until the flame is in view. Once the flame is in view Woody uses the front sonar to measure the distance to the flame. Between the Wii Remote and the sonar Woody is able to center himself in front of the flame. Once Woody is positioned in front of the flame he fires. If the flame goes away he reverts back to flame detection. If the flame is not out, Woody will attempt to shoot it again.

Sound generation

Throughout Woody's performance he plays various audio clips. These clips are purely for entertainment value. The clips are played by Arduino wave shield mentioned in the actuator section. The clips are triggered by certain events that occur

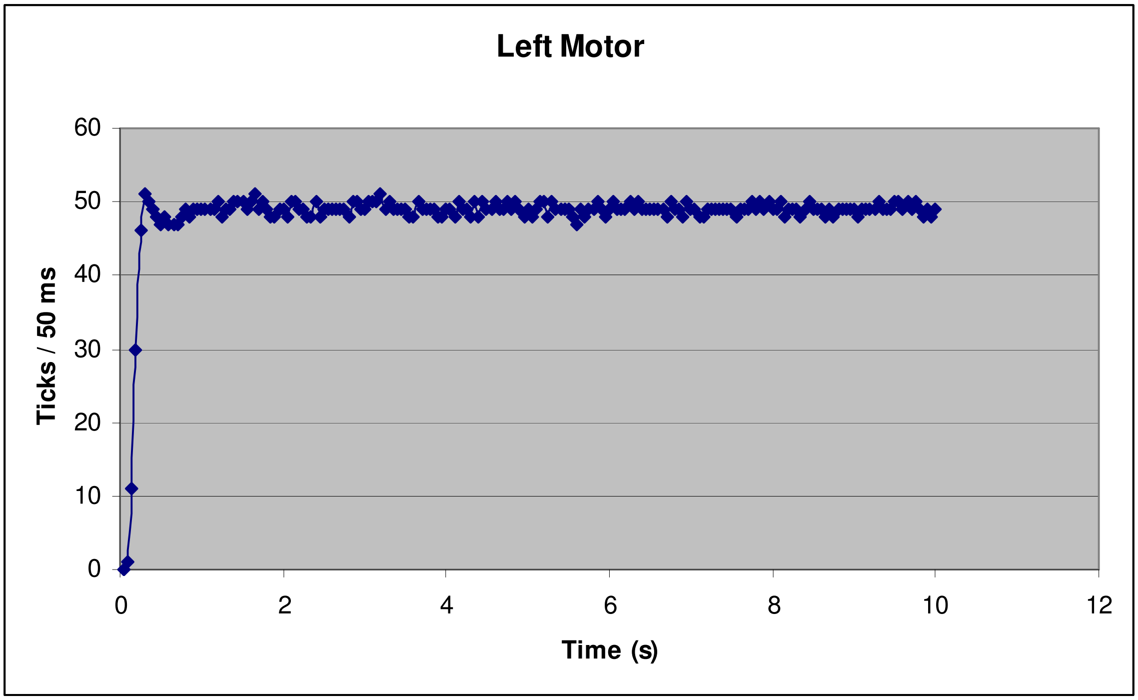
Event	Audio clip	Clip origin
Flame seen for the first time	"What are you doing in my swamp?"	Shrek
Flame seen after 20 seconds (going into extinguish mode)	"There is no escape. Don't make me destroy you"	Star Wars
Woody is in firing position	"Go ahead make my day"	Dirty Harry
Woody misses	"Just a bit outside"	Major League
Flame is out	Gators fight song	obvious

Experimental Layout and Results

PD controller

The PD controller was by far the most heavily tested part of Woody. The PD controller was in charge of keeping the rpm on the drive motors consistent no matter the terrain. In order to test the rpm controllers I had to have access to a data log rpm values. This was achieved by setting aside 400 bytes of SRAM for storing a 200 byte array for each motor. The refresh rate of the controllers was 20Hz which corresponded to a 10s sample time with 200 samples. The data was collected by sending the array values over the built in UART to Microsoft Hyper Terminal. Once the data was acquired in Hyper Terminal, I copied it into excel for tuning.

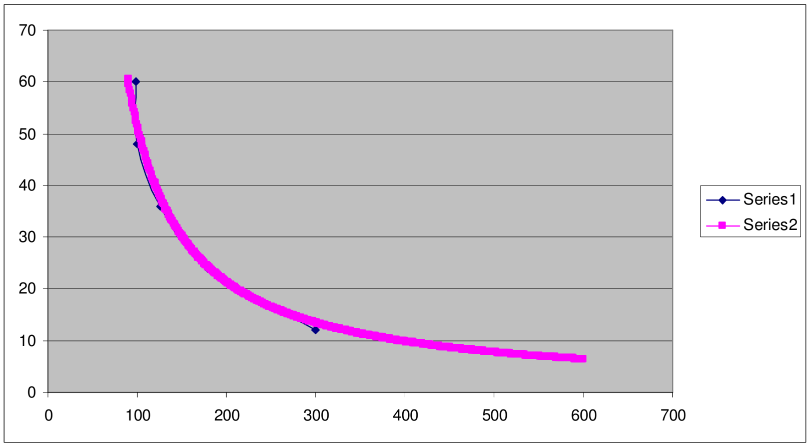
The Initial plan was to use a PID controller, but I discovered through testing that this sort of controller would go unstable if large disturbances were incurred. The integrator term caused the motors to oscillate violently when a motor's speed dropped for an instant. All of the initial tests were takes with the robot's wheels freely spinning. I was able to achieve great results. The following diagram shows that the left motor would achieve the desired speed in .3 seconds, with no overshoot, and would not vary its speed more than a couple percent.



I recently began testing with the robot moving. During these tests I determined it would be beneficial to slightly alter my P and D gains.

Sharp IR voltage to distance testing

At one point I was pursuing an alternative method to seeking out the flame. This method was edge finding, and ranging using the Sharp IR turret at the front of Woody. I was able to come up with a close fitting equation to compare voltage to distance, but unfortunately the Sharp IR rangefinder proved too inaccurate. The pink line is produced using the equation I developed, and the blue line is the raw data from the Sharp IR. As you can see there is very little of the blue line left, due to how well the equation conforms.



Equation

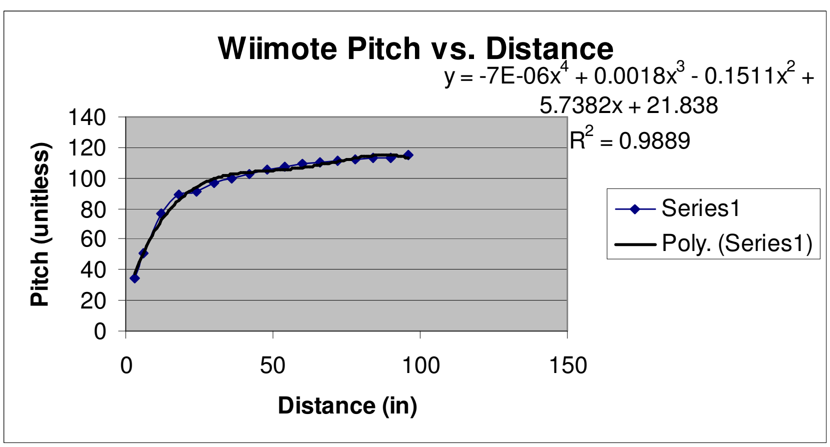
x = 10 bit adc reading

$$dist = 0.475 * 1904 * \left(\frac{x}{1024} * 256 - 7.5 \right)^{-999}$$

After discovering I needed a more precise distance measuring sensor I put a Parallax Ping Sonar on the front of the robot. The Ping has worked perfectly.

Wii Remote pitch to ranging

Another Idea I toyed with early on was using The Wii remote's center of mass pitch value for ranging. I found this method quite unreliable, which is why I is no longer present. Here is the data that was collected.



The main issue I discovered was that the Wii Remote would never record the same values for the same distances. Example: What was a 5 inch measurement one run might correspond to 12 inches after power was toggled on the Wii Remote.

Conclusion

Woody turned out to be a very successful robot. He works perfectly about 80% of the time. I impressed myself with how well I was able to implement some of his systems. The sense of accomplishment I achieved after creating PD controller, and seeing it work before my eyes was priceless. The first time I programmed Woody to perform obstacle avoidance while tracking the flame with the Wii Remote I realized this project was going to be a success.

Future work

Woody could stand for a few upgrades. It would be nice to incorporate voice recognition into the behavior list. I would also like to make Woody adaptive to attacking a flame without having to go through a centering behavior first. I would also like to eliminate the false positive he receives when a flame flickers after it is shot at. Woody tends to not recognize a flame that is dancing in the wind.

Note to Future Students

This has easily been the mot fun and educational semester I have ever had. I highly recommend this class to anyone with an interest for building robots. You will learn so much from this class that you can apply later. Do not hesitate to take this course!!!

Documentation

[1] "GP2Y0A02YK Optoelectronic Device" [Online document], 2006, [cited 2009 JAN 09], Available http://document.sharpsma.com/files/GP2Y0A02YK-DATA-SHEET.PDF

[2] "GP2D12 Optoelectronic Device" [Online document], 2005, [cited 2009 JAN 09], Available http://document.sharpsma.com/files/GP2D12-DATA-SHEET.PDF

[3] "PING)))™ Ultrasonic Distance Sensor" [Online document], 2005, [cited 2009 JAN 09], Available http://www.parallax.com/Portals/0/Downloads/docs/prod/acc/28015-PING-v1.5.pdf

[4] "L297 Stepper Motor Controllers" [Online document], 2001, [cited 2009 JAN 09], Available http://www.st.com/stonline/products/literature/ds/1334.pdf

[5] "L298 Dual Full-Bridge Driver" [Online document], 2000 [cited 2009 JAN 09], Available http://www.st.com/stonline/products/literature/ds/1773.pdf

[6] "SN754410 Quadruple Half-H Driver" [Online document], 1995 [cited 2009 JAN 09], Available http://focus.ti.com/lit/ds/symlink/sn754410.pdf

[7] " Infrared Proximity Sensor Long Range - Sharp GP2Y0A02YK0F" " [Online picture], 2009 [cited 2009 April 02], Available http://www.sparkfun.com/commerce/product_info.php?products_id=8958

[8] "Infrared Proximity Sensor Short Range - Sharp GP2D120XJ00F" [Online picture], 2009 [cited 2009 April 02], Available http://www.sparkfun.com/commerce/images/products/08959-03-L.jpg

[9] "PING))) Ultrasonic Sensor" [Online picture], 2009 [cited 2009 April 02], Available http://www.parallax.com/Portals/0/Images/Prod/2/280/28015-M.jpg

[10] "SPDT Switch Without Roller" [Online picture], 2009 [cited 2009 April 02], Available http://rsk.imageg.net/graphics/product_images/pRS1C-2110709w345.jpg

[11] "Wave Shield Add Sound to Your Arduino", [Online document] 2009 [cited 2009 March 01], Available http://www.ladyada.net/make/waveshield/

[12] Laidman, Rustle "Stepper Motors and Control Part III - Current Control of Stepper Motors" [Online document], 1999 [cited 2009 January 20] Available http://www.stepperworld.com/Tutorials/pgCurrentControl.htm

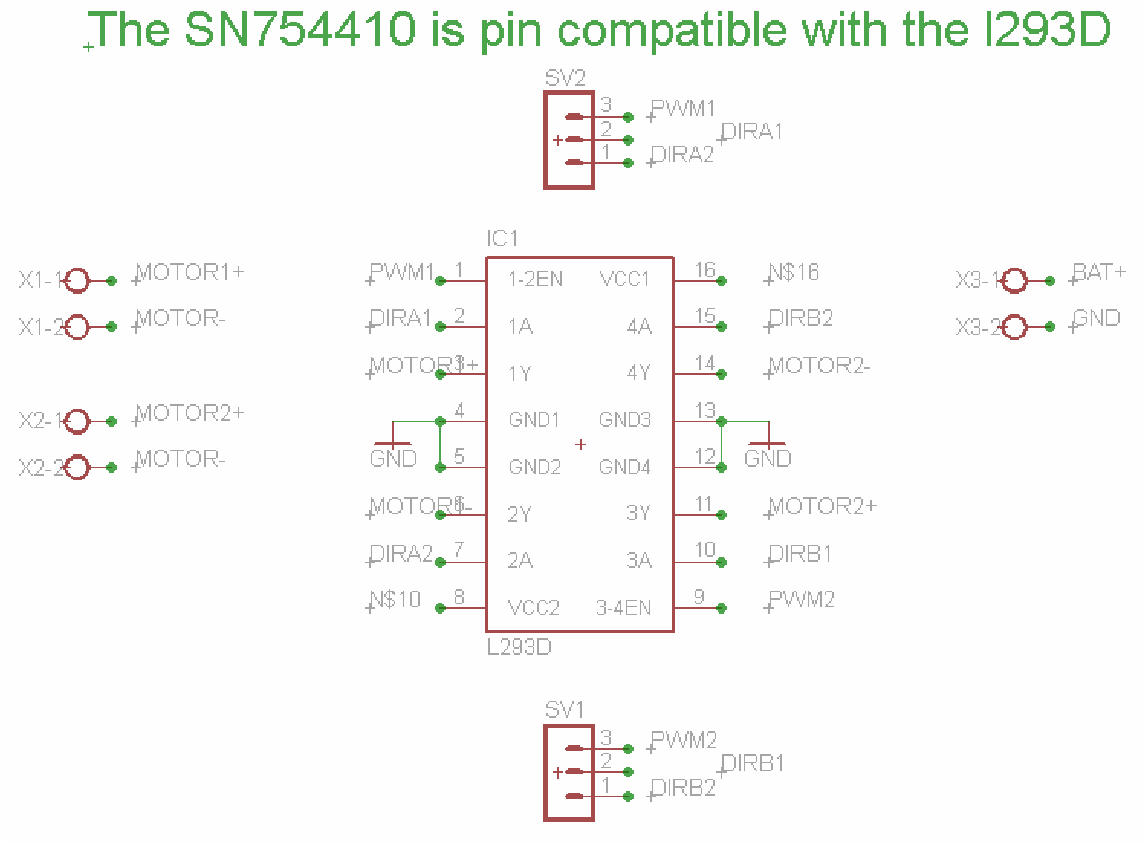
[13] "Wiimote" [Online picture] 2009 [cited 2009 April 02], http://www.etc.cmu.edu/projects/wiixercise/images/wiimote.jpg

[14] Hoeken, Zach "Sanguino" [Online document], 2009 [cited 2009 January 07] Available http://sanguino.cc

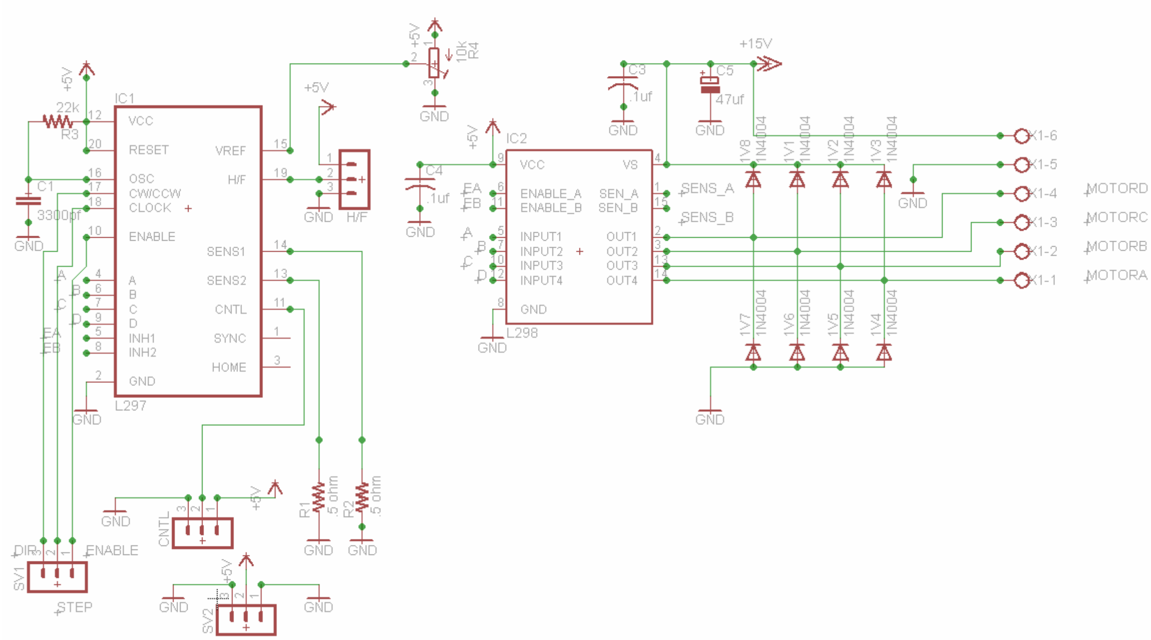
[15] Igoe, Tom "Arduino" [Online document], 2009 [cited 2009 January 07] Available http://arduino.cc

Appendix

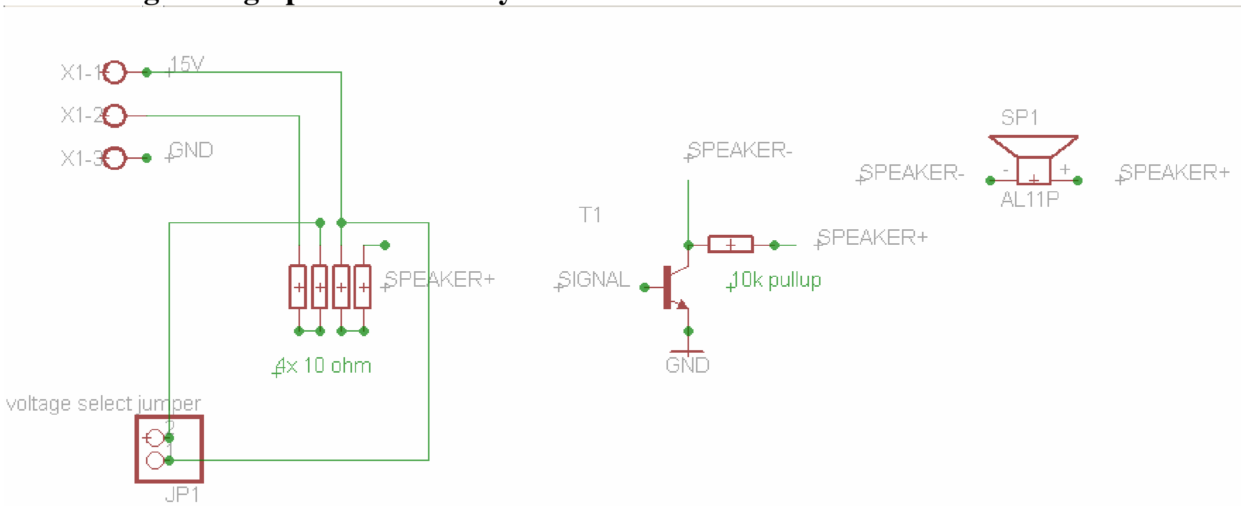
DC Motor driver



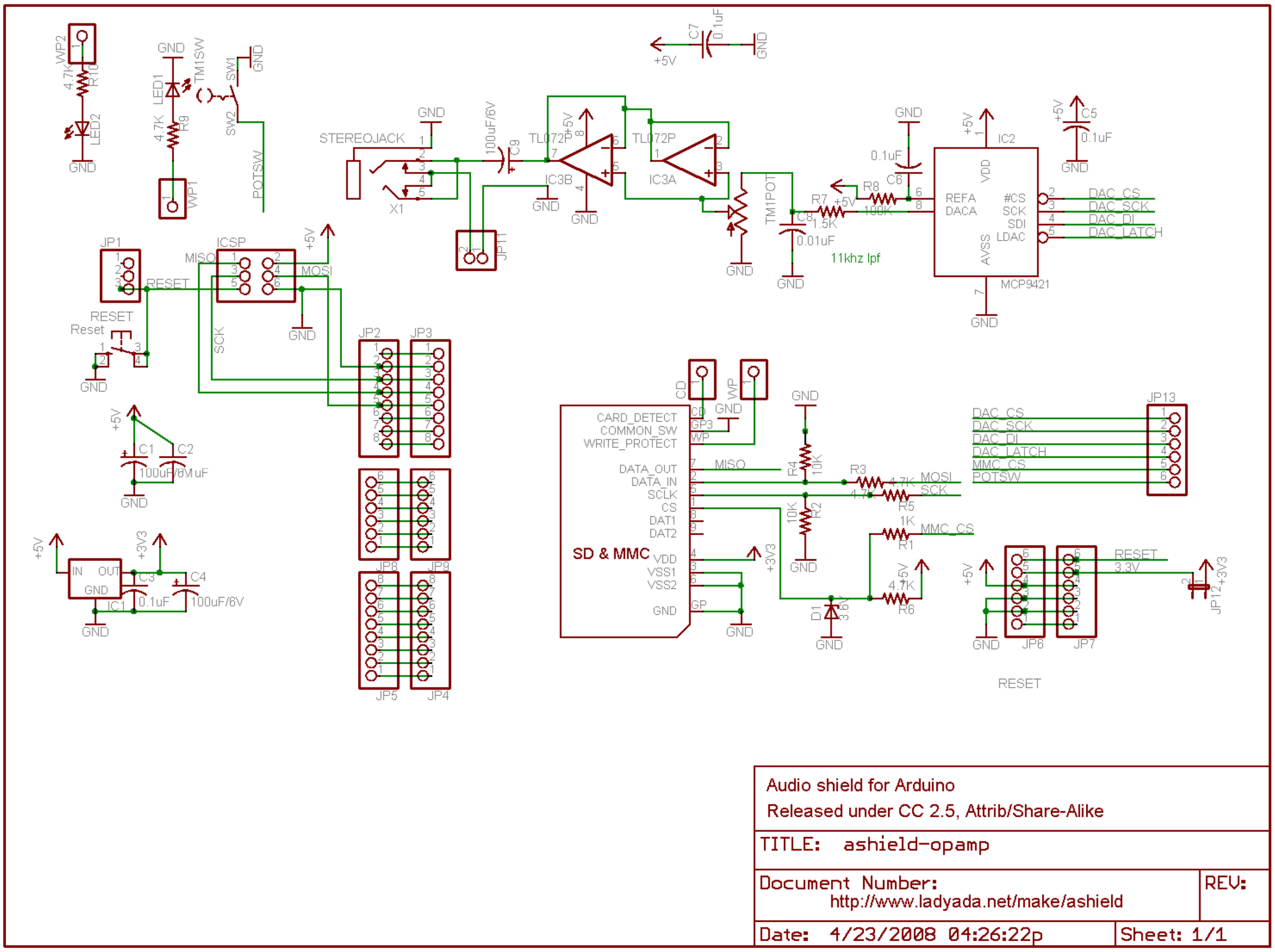
L297/l298 stepper motor controller and driver



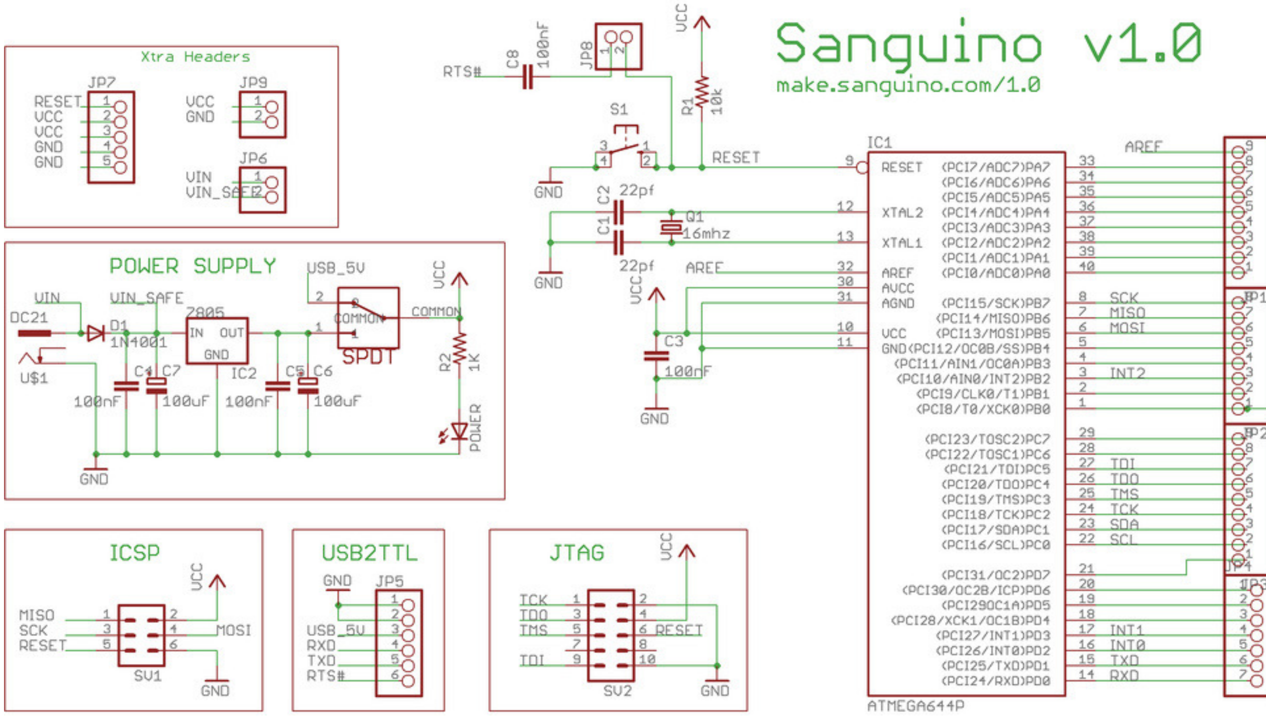
Fire Extinguishing Speaker Circuitry



Arduino Wave Shield Circuitry [11]



Sanguino Microcontroller Board [14]



Arduino Diecimila [15]