Spring 5-2005

Spray Application Robot for Greenhouse Environments

Brady Dorr Lewis
University of Tennessee - Knoxville

Spray Application Robot for Greenhouse Environments
(SARGE)

Adam Byard, Patrick Dobbs, Brady Lewis, Preston McCary

Dr. John Wilkerson, Mentor
James Wills, Mentor
Gary Honea, Mentor

University of Tennessee
Biosystems Engineering

May 2005

Spray Application Robot for Greenhouse Environments (SARGE)

Executive Summary

Chemical application in greenhouses is time consuming, labor intensive, and potentially hazardous. Alternative methods of applying chemicals in greenhouses exist, but these systems are cost prohibitive to average producers. SARGE is a robot that applies chemicals to vertically grown greenhouse crops, such as tomatoes, bell peppers, and cucumbers. The automated guidance system allows producers to leave the greenhouse during the spraying process. This decreases the time and labor costs of spraying, but more importantly makes chemical application safer for producers. The robot navigates the greenhouse using an inductive wire guidance system. A wire is installed in the greenhouse floor, and a 10 kHz signal is sent through the wire using an oscillator circuit. This signal induces voltages in inductors mounted on the robot. Using these voltages, a proportional control routine determines the appropriate output to the motors to follow the wire. The robot has two motors that control the rotation of the rear tires; the front tire is a caster wheel that is free to spin. SARGE's system software controls the modes of operation. These modes include setup, manual, and spray. In setup mode, producers input greenhouse and spraying variables. Manual mode allows users to drive the robot from one greenhouse to another using an RF controller. Spray mode applies chemicals to plants. Any errors, such as loss of wire signal or hitting an obstacle in the greenhouse, cause the robot to shut down and sound an alarm. The user interface is an LCD screen with a 4 button keypad and emergency stop button. The operator selects the appropriate menu or mode through this interface. During spraying, SARGE can shut down the right or left side of the vertical spray boom by actuating solenoid valves. This allows the robot to spray rows next to walls without wasting chemicals. Size specifications were based on a 96x30 foot greenhouse. The robot is approximately 23 inches wide and 32 inches tall. SARGE is powered by a 33 Ah battery, which was selected to allow the robot to spray three greenhouses in an eight hour day. The total cost of the prototype is $2600.

Acknowledgements

David Smith

- Cutting circuit boards
- Assistance in circuit design
- Trouble-shooting circuits
- Ordering electronic components

Craig Wagoner

- Machined NEMA boxes
- Cut mounting plate

Table of Contents

Background

Need

Chemical application in greenhouses is time consuming, labor intensive, and potentially hazardous. Producers must take time to mix chemicals, put on protective equipment, and spray the chemicals. This entire process takes approximately one hour with the actual spraying of the greenhouse taking thirty minutes. The protective equipment worn by producers ranges from gloves to a positive pressure suit made of chemical resistant material and is mandated by the EPA because the fungicides and pesticides can be hazardous to producers' health. The positive pressure suit has some a respirator to prevent the breathing of the chemicals being applied. While spraying, producers turn off the ventilation system in the greenhouse. This causes the temperature in the greenhouse to be even hotter, especially during the summer. The high temperature in the greenhouse, the protective equipment, and the weight of the chemicals being carried make spraying difficult. These chemicals can cause skin burns and respiratory ailments. An alternative method of chemical application in the greenhouse environment is needed.

Problem

Alternative methods of greenhouse chemical application exist, but are cost prohibitive to small-scale greenhouse producers. Types of alternative systems are rail-guided systems, pulley systems, and fogging systems. These systems range in cost from $10,000 to $40,000.

Goals

- Apply chemicals automatically to vertically grown greenhouse plants
- Remove operator from greenhouse during spraying
- Decrease time for chemical application
- Provide a cost effective alternative spraying system
- Create a system that is easy to implement in existing greenhouses

Objectives

- Constant application rate of 11gal/greenhouse
- Maximum spray time of 20 minutes
- Vehicle dimensions sized to fit in greenhouse rows
- Self contained power source for spraying
- Spray 3 greenhouses in an 8 hour work day
- Seal electronics to NEMA 4 standards
- Operation by one person
- Remote control operation
- Low infrastructure cost and time of installation
- Adjustable spraying for variable plant heights
- Develop graphical user interface
- Impart no permanent damage to plant
- Fully automated spraying process
- Remote or timed start to spraying process

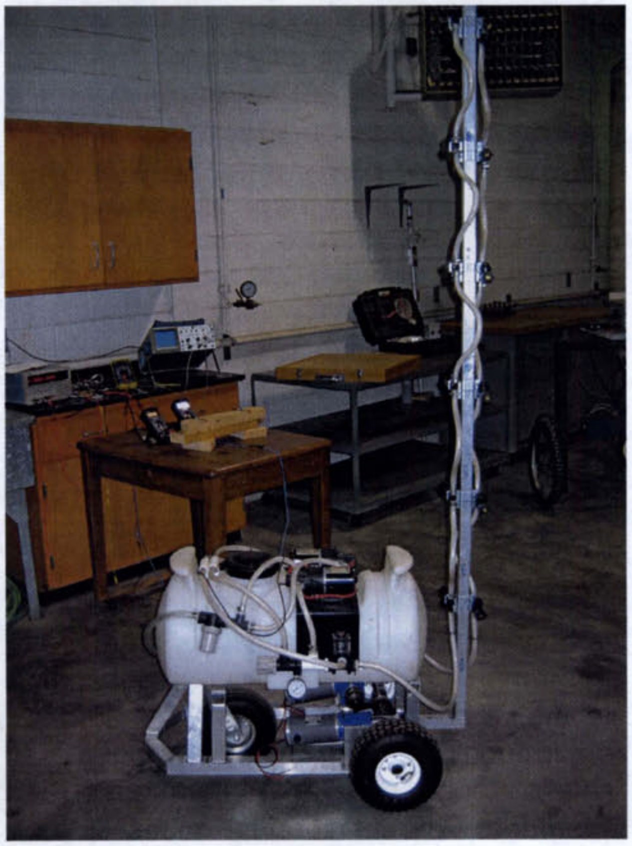
Objectives Continued

- Stop spraying on error
- Easy to clean

Introduction

The current design team inherited a previous team's prototype. The previous team's design concept was a robot that would take the place of a worker applying chemicals in a greenhouse. The previous team developed an aluminum frame outfitted with three tires, two motors, a spray tank, two pumps, valves, and a vertical spray boom. The two rear tires are connected to separate electric motors to control the steering and speed of the robot. The third wheel is mounted in the front and is a caster wheel to allow for steering by the two motors. The current design team's mentors advised the use of the previous prototype as a starting point. The mentors advised that the team should assume that the existing components are satisfactory to meet the design criteria. The previous team had researched guidance systems for the robot and selected an inductive wire guidance system; however, the previous team did not develop the guidance system. The current team reviewed several guidance systems and decided to proceed with development of the inductive wire guidance system. The team focused on developing the guidance system, user interface, and safety features of the robot.

Previous Prototype



Health and Safety
A primary reason for development of SARGE is the current problem of exposing operators to agricultural chemicals in the greenhouse environment. To achieve safety during spraying, SARGE is designed for the operator to remain outside of the greenhouse until it is safe to enter. This goal is integral to the system. The automatic guidance and timed start features allow a producer to start the robot and then leave the greenhouse. During the timed start, the robot sounds an alarm as it counts down to the spraying process. This alarm warns anyone in the greenhouse of the spraying that is about to begin. In addition, any errors that occur during spraying, like wire signal failure or traction problems, result in a stop command that ends the robot's driving and spraying. This avoids either driving off course or over applying chemicals in part of the greenhouse. Another safety feature that can be implemented is a sensor on the greenhouse door. This sensor would detect someone entering the greenhouse when the robot is spraying. Upon detection of the door opening, the sensor would turn off the signal to the guidance wire causing the robot to stop. A set of light emitting diodes, LEDs, can be implemented on top of the spray boom to warn producers of the robot's action in a greenhouse.

Human safety is also a concern when the robot is not applying chemicals in the greenhouse. EPA regulations will apply when the user is handling the chemicals to load the spray tank; though, the exact protection requirements will vary based on the chemical. In addition to chemical concerns, the general use of the robot has also been analyzed. The slow travel speed of the robot minimizes any damage that can occur if the robot hits something. If the robot does run into something, it will back up and then shut down. Another concern about the robot is mobility between greenhouses. To prevent producers from attempting to lift the robot, a manual control system has been implemented. This allows producers to drive the robot from one greenhouse to another. An analysis of the tipping moment of the robot has been performed to account for different travel conditions between greenhouses. This proved tipping of the robot to be a minimal concern. After using the robot, producers will need to clean the robot. This is facilitated through an easy to drain spray tank and the sealing of all electronics to National Electrical Manufacturers Association (NEMA) 4 standards. Producers can spray the robot with water when cleaning without damaging the electronics.

Environmental Impact
The impact on the environment is important and could have lasting implications. This system allows the appropriate amount of chemicals to be applied to the plants without excess. This is accomplished by correlating the ground speed of the robot with the flow rate of the hydraulic system. This will prevent excess chemicals from entering the environment. The vertical boom is less prone to leaking than traditional horizontal booms because the liquid drains back into the system instead of dripping from the nozzles. The simplified cleaning system allows the operator to ensure that the contaminated water is disposed of properly. This can be done by placing the robot over a drain to empty the tank and clean the robot.

Environmental impact was also a factor in component selection. Components were selected based upon ease of availability, disposal, and recyclability. This was particularly important when selecting the battery. The battery is a deep-cycle lead acid that is readily available and easily recycled at battery supply stores.

Guidance

Overview

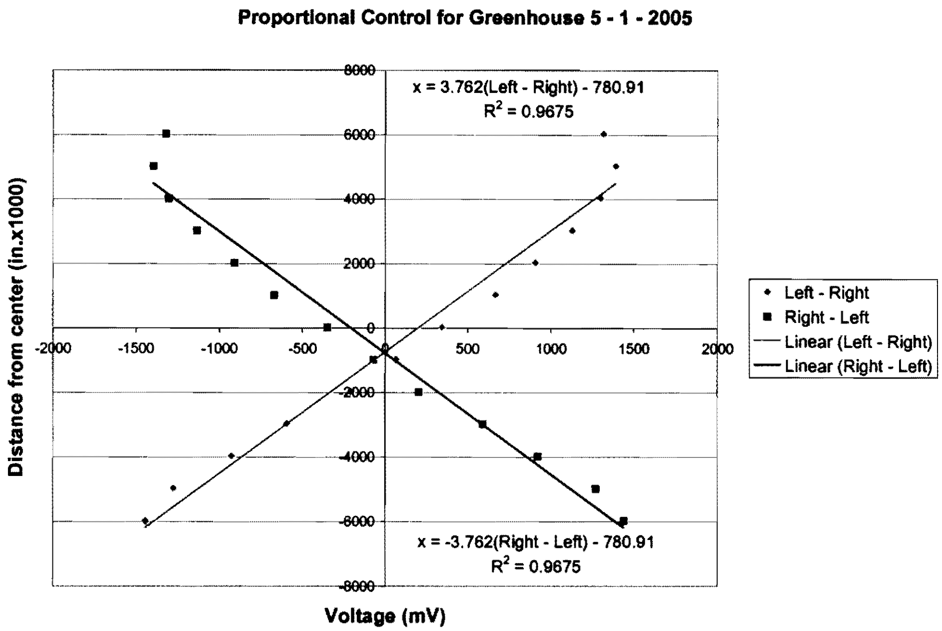
SARGE has two modes of guidance, manual mode and spray mode. Manual mode is controlled by the user using an RF remote control. Spray mode is inductive wire guidance. SARGE has two inductors; each inductor is mounted on the underside of the front frame rail on the right and left of the robot. A 10 kHz signal is generated by the oscillator circuit and runs through the wire. This signal creates a magnetic field that induces a voltage in the inductors. The voltage signal from the inductors is amplified, filtered, and rectified. The amplification is performed by a two stage amplifier with a 200x gain. The filtering is critical to reduce the noise in the signal and is done by an eight pole band pass filter set at 10 kHz. The adjusted voltage signals can be used to develop a control system that allows SARGE to follow the wire.

Manual Mode

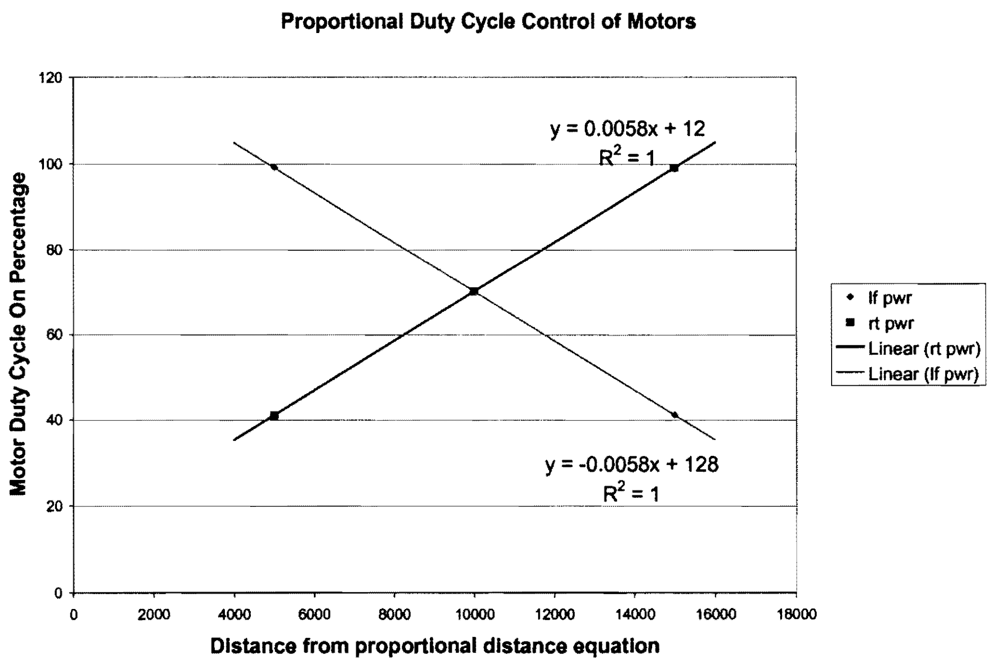
As mentioned, manual mode is the control of SARGE by the user using a remote control. It is a 2-channel FM controller typically used on remote-controlled aircraft. The receiver produces a pulse-width modulated signal that is changed as the levers on the controller are actuated. In manual mode, the guidance software measures the width of the signals and interprets them into forward, reverse, right, or; this information is sent to the motors.

Spray Mode

Spray mode has three critical parts, the main guidance/steering control, the speed control, and the turn sensing. The steering control is a proportional control routine. The error term is the difference in the right and left voltages from the inductors. Development of the routine consists of measuring voltages from the two inductors at known distances from the wire. From this information, an equation relating distance from the wire and voltage difference was developed. The software computes the distance from the wire using this equation, and then calculates the appropriate duty cycle for each motor. The development of this process is shown in the following chart.



Each duty cycle is computed from a linear equation that relates the duty cycles to the distance from the wire found from the previous step. The lowest duty cycle occurs when a given motor is closest to the wire, and the highest duty cycle occurs when that motor is farthest from the wire. This allows the robot to turn back on course if it begins to move away from the wire. The lowest duty cycle is set at forty percent because the motors do not operate effectively below a forty percent duty cycle. The development of the duty cycle calculation is shown in the chart below.



A magnetic proximity sensor measures the speed of the robot by counting the teeth on a sprocket that is mounted on the front wheel. This information is used by the software to determine if the robot is traveling the desired speed, which is dependent on the application rate set by the user. Speed and steering are multiplied to get the final output to the motors. Speed is initialized at one; therefore, the initial output to the motors is equivalent to that from the steering routine. The speed factor is adjusted by a bump routine that increments by ten percent. The routine also has a twenty percent window around center that is considered an acceptable operating range.

The third critical part of the spray mode guidance is the turn sensing. This is accomplished by counting the number of consecutive loops that one of the steering calculations is above a threshold. After exceeding the threshold for the set number of loops, the robot enters turn mode. This mode slows the robot and shuts off the hydraulic system. Slowing the robot's speed makes it easier for it to follow the curve, and shutting off the hydraulic system avoids spraying around the ends of rows which is unnecessary. The robot exits turn mode by counting the number of consecutive loops that the steering calculations have been below the turn threshold. Upon exiting turn mode, the robot resumes the same speed that it was traveling before entering turn mode and turns the hydraulic system back on for spraying. Turn mode also counts the number of rows that have been sprayed, allowing the robot to know when to turn off one side of the boom or finish spraying.

Additional Subroutines
Other subroutines that the robot can enter while in spray mode include reverse and spin subroutines. The robot senses that it is stuck if the ground speed has been zero for a second. In this case, the robot will send full power to both motors for an instant to try to get unstuck. Spin is necessary for operation on gravel, but is rarely used on concrete. If the robot goes into spin five consecutive times, it assumes there is an obstacle. It then backs up a short distance, turns off the hydraulic system, and sounds the alarm.

Recommendations
The guidance system could operate more effectively using a faster processor with more memory. When developing a proportional control routine for steering the robot, voltage and distance data are needed for each new greenhouse. The new data is needed to account for wire type, depth of wire, and greenhouse floor material. This process could be automated with a new processor.

Operating System

Two Processor System
The SARGE system uses a dual-processor setup with two Parallax BasicStamp 2 Microcontrollers. The master stamp controls the user interface, hydraulic system, and the logic operations of the robot. The guidance stamp is responsible for the inductive guidance and motor control. Due to their functions, the master stamp needed a large EEPROM memory, and the guidance needed a fast processor.

The master stamp is a BS2pe; it has 32 KB of EEPROM and can process 6,000 instructions per second. The guidance stamp is a BS2sx; it only has 16 KB of memory but can perform 10,000 instructions per second.

Communication Protocol
The two stamps communicate serially at 50 kbps using a custom protocol. Each stamp sends an 8-bit code; the different bits in the byte signify different messages necessary for the program operation.

Bit	Purpose	When 00	When 01	When 10	When 11
7	Go/Stop	Stop	n/a	n/a	Go
5-6	Set Mode	Run	Setup	Manual	Menu
0-4	Speed Variable	Decimal Number (0-32)			

Command Byte (Master to Guidance)

Bit	Purpose	When 00	When 01	When 10	When 11
6-7	Drive Status	Straight	Turn Right	Turn Left	Spin Error
5	Guidance Status	Signal Error	n/a	n/a	Signal OK
4	Speed Status	Speed Error	n/a	n/a	Speed OK
2-3	Mode Status	Run	Setup	Manual	Menu
1	n/a	n/a	n/a	n/a	n/a
0	More Data	Nothing	n/a	n/a	Data Coming

Status Byte (Guidance to Master)

The stamps communicate once during each loop of their respective modes. Since the loop time is critical for the guidance stamp, it initiates communication by sending the status byte before listening for a command.

Modular Operation
The operating system is divided into four major modes of operation with a number of single-task subroutines. The individual modes are continuous loops which require a mode command to change state. Each stamp has different tasks to perform during each mode.

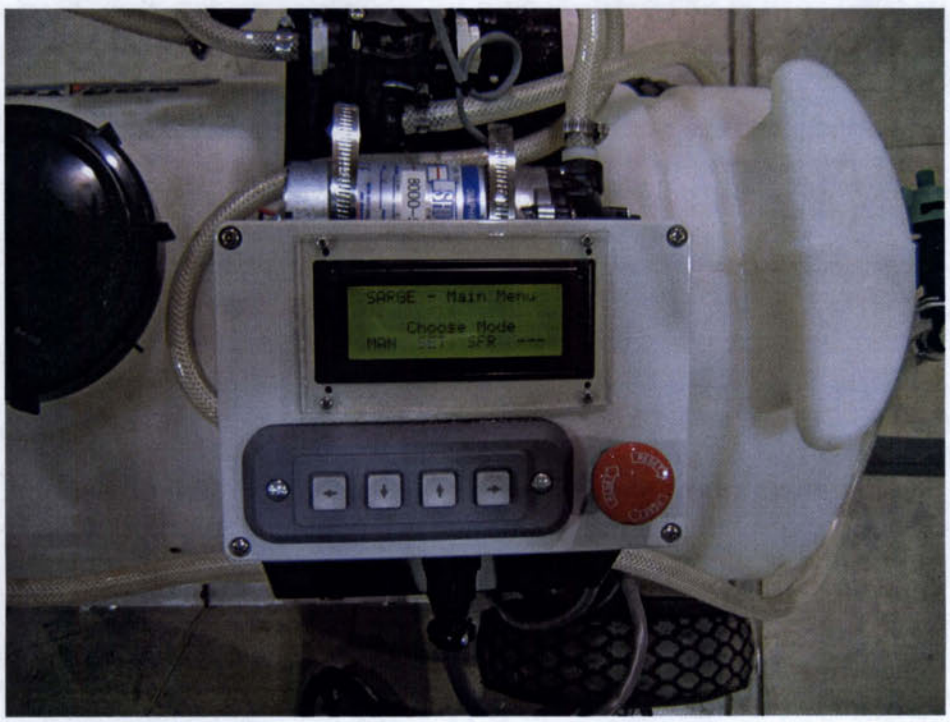
- The menu mode is the starting point of operation and allows the user to enter the other three modes of operation. The master stamp displays the menu on the LCD screen and watches the keyboard inputs. The guidance stamp listens for a mode command from the master.
- The spray mode is the stand-alone mode of the robot designed for chemical application while following the guidance wire. The master stamp keeps track of the corners the robot has navigated and controls the hydraulic system accordingly. The master stamp monitors the keyboard and the emergency stop button and also sounds the alarm. The guidance stamp measures the inductors and calculates the appropriate power levels for the motors. The guidance stamp also alerts the master stamp if there are any driving errors (spin, no guidance).
- The setup mode allows the user to input greenhouse variables prior to spraying. The menu interface allows a producer to tell the robot the parameters of the

greenhouse so that it can calculate the necessary program variables. Information gathered includes the number of turns in the greenhouse, nozzle size, desired application rate, and which boom segments to activate at the beginning and end of the spray process. The master stamp collects the information through several LCD input screens and the keyboard interface. The guidance stamp simply listens for each variable it needs for spray operation.

- The manual mode activates the manual guidance controller of the robot. The master stamp waits for the user to end the manual mode by hitting any of the keys. The guidance stamp operates the motors in response to the manual controller.

User Interface

Liquid Crystal Display (LCD)
A Liquid Crystal Display (LCD) screen was found to be the easiest way to display information to the user. The easy to read screen allows information to be displayed without any confusion. The LCD display is sealed in a NEMA 4 box to seal it from any water or chemicals.



Buttons
A series of four buttons, mounted below the LCD screen, allows the user to input information such as number of rows in the greenhouse. Also, users can move from the main menu page into any one of the sub menus such as setup and spray mode using these buttons. The red emergency stop button is easy for producers to activate and remains pressed until the producer resets the button

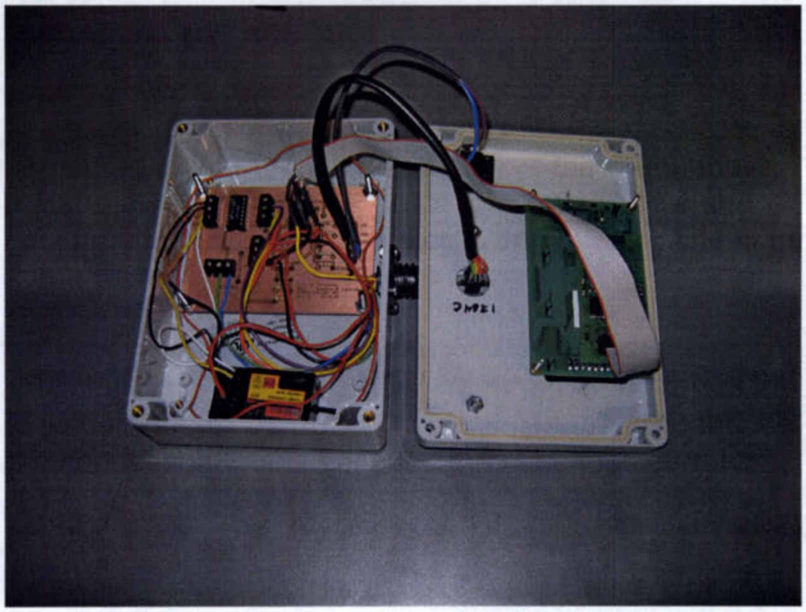
Alarm
An audible alarm has been implemented to inform anyone outside of the greenhouse that an error has occurred. The alarm also informs anyone who might still be in the greenhouse that they need to exit immediately before spraying commences. The alarm is mounted underneath the tank in a NEMA 4 box.

Recommendations
The team recommends that four light emitting diodes (LEDs) be mounted on top of the boom so that the producer could visually check the robot from outside the greenhouse. LEDs should be used instead of incandescent bulbs because they use less current.

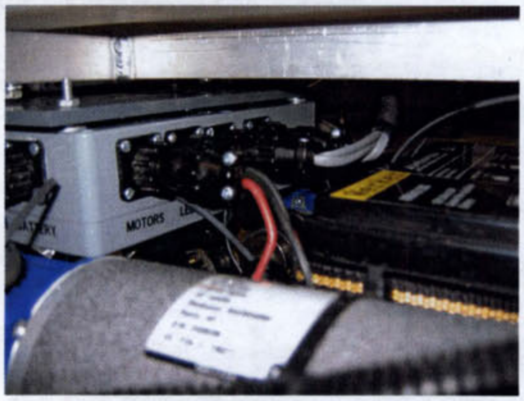
Electronics

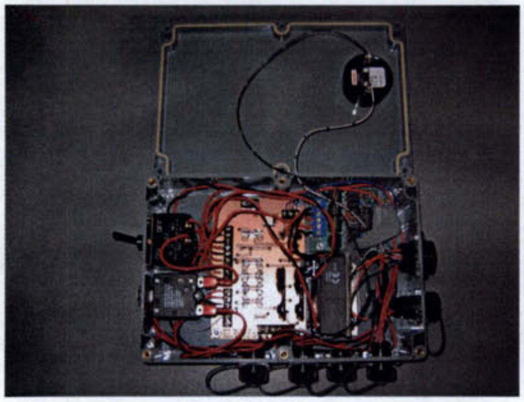
NEMA Boxes
The electronic components of the SARGE system are sealed to NEMA 4 standard. With this level of protection, a producer can easily clean the robot with a water hose or pressure washer. Due to space restrictions on the robot, the electronic components are divided into three boxes based on function. Each box is connected to the others with AMP connectors which are keyed to prevent misalignment and are screwed on to ensure a good connection with a water-tight seal. The cables connecting the boxes with each other, the pump, the solenoid valves, the motor, and the battery use a SARGE standard for connector and pin definitions, and wire color.

LCD Box
The LCD Box is placed on a bracket above the tank, allowing producers to interact with the user interface easily. This box contains the LCD screen, keypad, emergency stop button, manual control receiver, and the 8-3 encoder board. The screen is controlled through a simple serial interface. The 4-button keypad is pressure-washer safe. The manual control receiver only produces a 3V signal; analog comparators increase the voltage of the signal to 5V so that it can be reliably processed by the microcontroller. The 8-3 encoder board contains the comparator circuit and an 8-3 encoder chip. This logic chip allows 8 high/low input buttons to be connected to the microcontroller with only 3 data lines. The current design only uses 5 of the inputs with 3 extra for future functionality.



Power Box

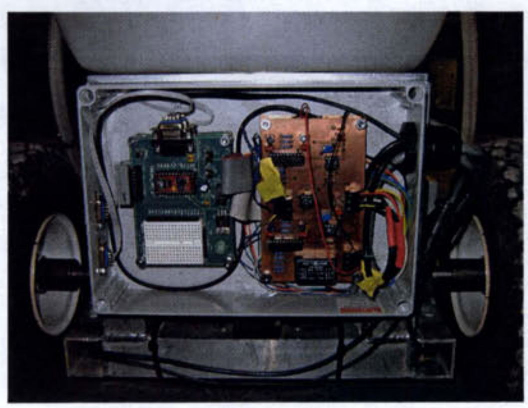




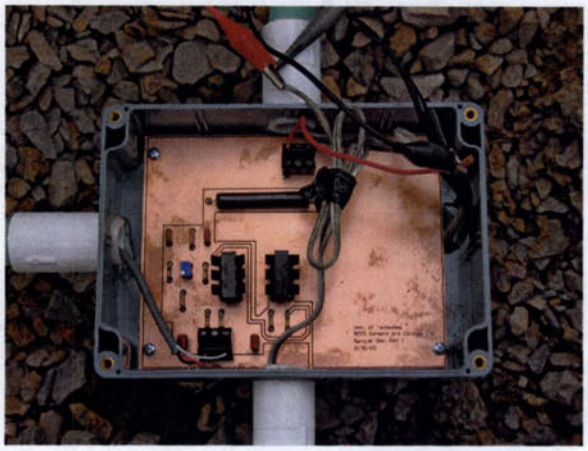
The Power Box is located underneath the tank, directly above the belts and pulleys. This box contains the main power switch, circuit breaker, speaker, H-Bridge, solid-state relay, and high power circuit board. In addition, this box connects to the battery, motors, LEDs, pump, and solenoid valves. The power switch and circuit breaker are sized for the maximum power draw of the robot, 30A, and are easily accessed from the side of the robot. The speaker is a low-power piezoelectric buzzer that alerts the user of the robot operation. The H-Bridge is used to control the motors. The H-Bridge contains dual 2-to-4 decoders and optical isolators so that the motors can be easily controlled by the microcontroller with no danger of surging the stamp. The 20A solid state relay isolates the microcontroller from the pump. The high power circuit board contains optical isolators for each of the solenoid valves, a photo-voltaic relay for the LEDs, and fuses for each motor, the pump, and both valves.

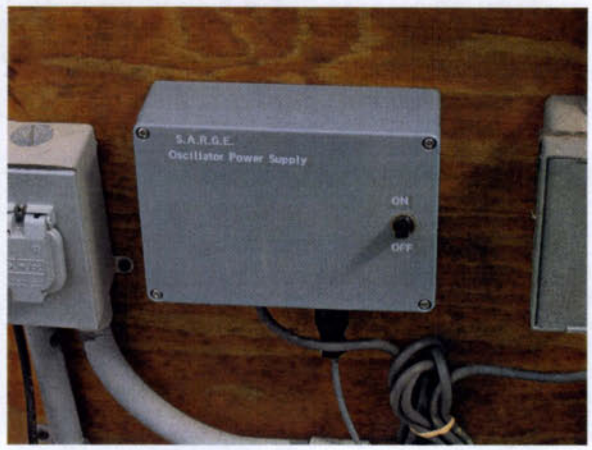
Primary Box

The primary box is located on the rear of the robot, between the boom and the frame. This box contains the microcontrollers, the digital board, and the analog board. The primary box also connects to the LCD box, the power box, the inductors, and the speed sensor. Each BasicStamp is mounted on a Parallax carrier board. They have their own 5V regulated power supply and are connected to the digital board by a 20-pin ribbon cable. The digital board makes all the connections between the microcontrollers and the rest of the system. It has a large number of screw terminals connected to the two 20-pin headers for the stamps. The digital board also holds the voltage divider and 8-bit analog-to-digital converter used to measure the battery voltage. The analog board is the heart of the guidance system. It connects to the inductors and processes the signals coming from each sensor.



Oscillator
The oscillator that is installed in the greenhouse is made up of two components: the power supply and the oscillator itself. The oscillator is buried underneath the floor of the greenhouse with the guidance wire. This is to allow the robot to travel along a continuous path. The oscillator power supply is mounted on the service panel in the greenhouse, near the 120V AC power outlet. This box is a simple rectifier and DC-to-DC converter that generates the +/- 12V needed by the oscillator. It is placed close to the AC outlet so that the low voltage 12V signal is sent underground, rather than the higher 120V. The switch on the power supply is easily accessible and controls the guidance wire.





Fluid Handling

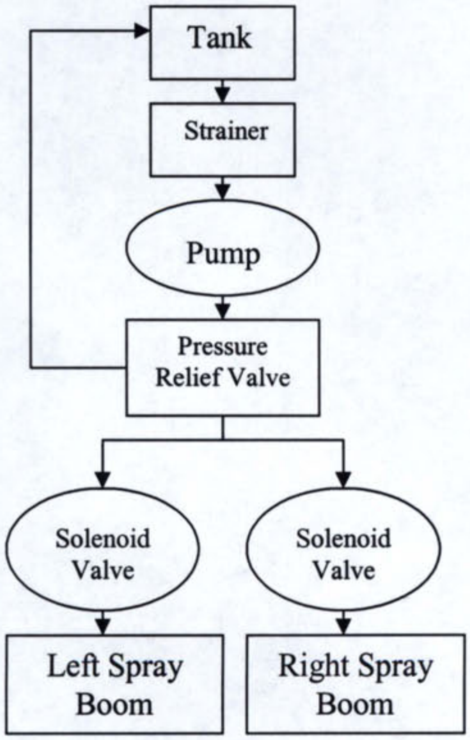
Previous Design
The goal of SARGE is to automatically apply chemicals, such as fungicides and pesticides, to vine growing crops in greenhouse environments. To meet this goal the fluid handling system inherited from the previous senior design team was evaluated. First, the nozzles were evaluated to ensure the appropriate application rate of the chemicals. Second, the pumps were evaluated for appropriate pressure and flow.

Pumps
The previous design team used two Flojet pumps that were each rated at 1.8 gpm at 60 psi. The pumps were sufficient enough to provide the desired flow of 0.96 gpm, however, each pump used 84 watts of power. The team saw the old design had a large amount of water recirculating back into the tank from the solenoid valves. Pumping this extra water wasted energy, so it was decided to evaluate the fluid system. When calculating the power requirement to size a battery, the team found that a 26 pound battery could be used instead of a 46 pound battery if power consumption were decreased. The team looked at only using one of the old pumps, but the maximum pressure rating was 60 psi, which is also the desired operating point for the nozzles. This cannot be achieved due to head losses in the system. The team decided to use one 84 watt Shurflo pump with a pump rating of 1.8 gallons per minute at 100psi. Using the manufacturer's pump curves it was determined that the operating point for our

system would be 1.1 gallons per minute at 60 psi. The pump draws water from a 15 gallon tank that is mounted on top of the aluminum frame. To ensure that the pump does not get clogged, a 100 gage strainer is in line between the tank and pump. The pump discharges to a pressure relief valve that is used to set in the appropriate pressure for the nozzles.

Solenoid valves

After the pressure relief valve the line splits and enters the two solenoid valves. Each valve controls one side of the vertical spray boom. Control over individual sides is needed to ensure that on the outside rows of the greenhouse one side of the spray boom is turned off to avoid spraying the walls with chemicals. This could cause degradation to some types of greenhouse coverings. In addition, it wastes the chemical and could be harmful to the environment. A schematic of the hydraulic system can be see below.



Nozzles

Nozzles were selected based upon the speed of travel, application rate, and row spacing. Using these criteria an optimum flow rate is achieved. The team decided to use a TeeJet Hollow Cone Spray Tip (TXVK-4) with a flow of 0.08 gpm per nozzle at 60 psi. This nozzle was selected after discussing nozzle patterns with the team mentors. Other nozzle patterns can be used as long as the flow does not exceed 0.13 gpm/min. The hollow cone offers a finely atomized spray. While spraying, the greenhouse ventilation system is shut down. Consequently, drift is not a concern. This allows the design team to use a nozzle that has a finer droplet size, such as the selected hollow cone nozzle. This type of nozzle allows for faster drying times so the producer can reenter the greenhouse sooner. The smaller droplet size also gives better coverage of the chemical on the leaf, as shown below.

Robot Specifications

Turning Radius
The frame has been modified from its original design to achieve that a turning radius of 2.5 feet is achieved. The width of the robot is approximately 23 inches and the length is approximately 38 inches. This is to ensure that the robot fits in the rows of a greenhouse.

Weight
SARGE weighs 140 pounds when the tank is empty, but when the tank is full of water and chemicals it weighs approximately 260 pounds. The frame is built of an aluminum tubing to reduce the weight.

Tipping Moment
SARGE will tip if traveling up grades steeper than 43%, which corresponds to 23°. It should be noted that driving across grades steeper than 5% will cause the robot to turn down grade. The tipping moment was calculated using formulas from *Tractors and Their Power Units* by John Wiley. This data was also experimentally proven by testing on a hydraulic tipping platform. The center of gravity was found by measuring the weight on each tire while the robot is level and by measuring the length, height, and width of the robot. Once this has been calculated the robot is tipped backwards until there is a significant change in the center of gravity. Once this information is determined the tipping moment can be calculated.

Tipping Moment

	Degree longitudinal	% Slope
sprayer empty	27	50
sprayer (4 gal)	24	45
sprayer (6 gal)	23	43
sprayer (8 gal)	24	44
sprayer (10 gal)	24	45
sprayer (12 gal)	25	47
sprayer (14 gal)	27	50

Motor Specifications
SARGE currently uses two 12 volt 0.063 hp with a right angle gearbox with a 40:1 reduction ratio. These gear motors each have an output speed of 25 rpm and a torque of 81.8 in-lb.

Yaw Rate
Yaw rate is the rate at which the robot turns about one of its back tires. Yaw rate was calculated after the sides of the frame were widened. The yaw rate is currently 45^{o}/sec.

Driving in Reverse
SARGE does have the capability to drive in reverse; however, due to the frame design by the previous team, the front wheel, a caster wheel, cannot rotate 180^{o} and does not track appropriately. This makes it impossible to drive in reverse for long periods of time since the wheel contacts frame.

Recommendations
It was discovered by the current design team that the motors do not have enough power to drive with a full tank. The team suggests that 12 volt 0.16 hp motors with a right angle gearbox be used. These motors have an output speed of 49.1 rpm and a torque of 88.7 in-lb. The pulleys connecting the motors to the axles need to be changed from a 1:1 ratio to a 2.5:1 ratio; this will reduce the speed of the wheels and increase the torque.

The team also recommends that the front of the frame be altered so that the front tire can spin freely. This should be done by widening the front of the frame.

Power Requirements

The system power requirements to operate SARGE were determined by summing the power needed by the Basic Stamp, the pump, the solenoid valves, and the motors. It should be noted that the majority of the power deals with the hydraulics because the pump and both valves require approximately 125 watts. When the tank is full and the sprayer is traveling on gravel, the motors require approximately 60 watts. While the tank is full and the sprayer is driving on concrete, the motors only need approximately 20 watts. The power needed by the motors will constantly be changing. The major variables that are responsible for this varying amount of power are the type of surface on which the sprayer is moving and the flow rate out of the tank. Equation #1 is used to calculate the power at specific times within the greenhouse.

$$P_{total} = \left[P_{BS} + P_{P} + P_{SV} + \left(\left(W_{frame} + W_{H2Ofull} - m_{H2O} * t_{s} * g\right)\mu * v * \mathrm{eff}\right)\right]$$

Equation 1

P_{total} = total power
P_{BS} = basic stamp power
P_{P} = pump power
P_{SV} = solenoid valves

W_{frame} = weight of the frame
$W_{H2Ofull}$ = weight of the tank completely full
m_{H2O} = mass flow rate of water
t_s = amount of time spraying
g = force of gravity
μ = rolling coefficient of friction
eff = efficiency
v = velocity

The type of surface directly affects the frictional forces on the wheels, and consequently, the power to run SARGE is changed. The rolling friction or rolling resistance is affected by both the coefficient of rolling friction and the weight of SARGE. The coefficient of rolling friction is determined by the relative roughness/smoothness, hardness, deformation, and molecular interaction of the surfaces in contact. Rough pavement, a softer surface, low tire pressure, and small wheel diameters will result in a higher coefficient of rolling friction. Smooth pavement, a harder surface, high tire pressure, and larger wheel diameters will result in a lower coefficient of rolling friction. These variables ultimately cause greater or lesser deformation between the ground and wheels during contact. Differences in rubber tire composition and road composition can also affect the coefficient of friction by altering the molecular interaction between the contact surfaces. Therefore, to minimize the coefficient of rolling friction:

- Drive on harder and smoother surfaces
- Use larger diameter wheels
- Use higher tire pressures
- Use smoother and thinner tire treads
- Use narrower tires and tread patterns

Currently, the team is only concerned about the sprayer traveling on gravel and concrete. The worst case scenario is gravel because it has the highest rolling coefficient of friction, which is 0.1. The coefficients of rolling friction were determined by pulling the sprayer with a spring scale. It was established that gravel is the worst case scenario because the power requirements to run the sprayer on gravel are much higher. The power to drive on concrete, however, is the lowest with a the rolling coefficient of friction is 0.03.

The level of fluid in the tank will also affect the power necessary to run the robot. As the robot travels throughout a greenhouse, the level of fluid within the tank will decrease so the power to run the robot will continually decrease. The worst case scenario obviously is when the tank is full because the sprayer needs more power. Therefore, the sprayer had to be designed to produce enough power to begin this process.

The team also sized the battery by adding up the current required by the motors, pump, valves, and basic stamp. The team concluded that a 33 AH battery was sufficient for our purposes.

Cost Analysis

The team is below the maximum cost of development of $2,000. Currently, the development cost of SARGE is $1,535.62. This is partly due to the fact that $327.30 of electronic components were donated. These components include the following:

- (2) Board of Education – manufactured by Parallax
- Basic Stamp 2sx and 2pe – manufactured by Parallax
- Maxi Dual H-Bridge – manufactured by Mondo-Tronics
- Magnetic Proximity Sensor – manufactured by Allegro

Without these contributions the total development cost would be $1,846.92. The distribution of costs throughout the year can be seen in the following chart.

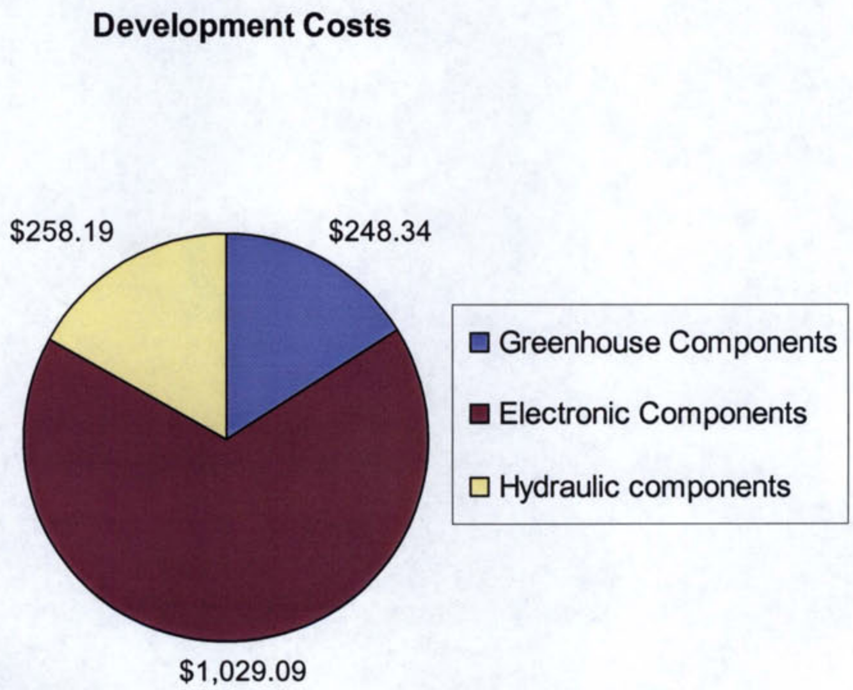


Chart 1: Development Costs

Seen in the chart above, the majority of the costs involve the electronic components, while the next large portion of the spendings include the components installed inside the greenhouse.

To produce another prototype, the team determined that the costs would be approximately $2,601.97. This calculation was based solely on the costs to produce the robot. As seen in the chart below the overall cost of the prototype can be broken down into the electronic components, hydraulic components, driving components, and frame components.

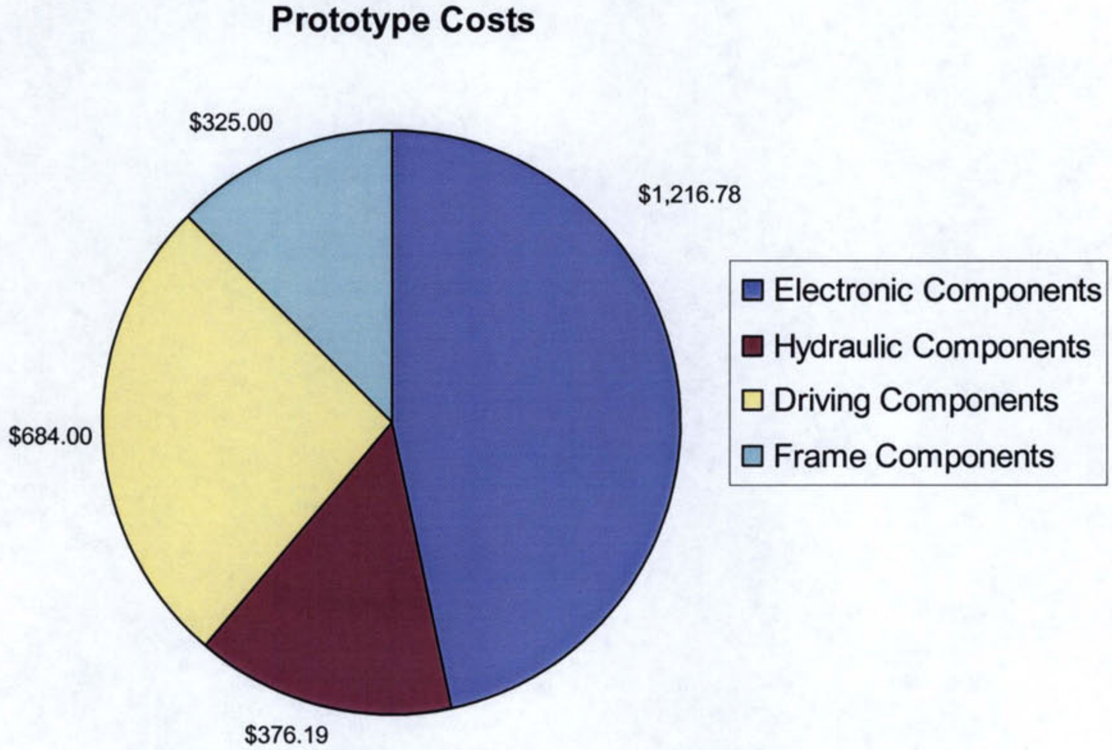


Chart 2: Prototype Costs

As in the development costs, it is apparent that the electronic components have the greatest impact on the costs, in this case $1,216.78. This value for the electronics is larger than before because this includes the board of education, Basic Stamps, Maxi Dual H-Bridge, and the magnetic proximity sensor. The cost of the hydraulics increased because the previous team bought the 15 gallon tank and solenoid valves. Also, it should be noted that this chart addresses the driving components and the frame components of the sprayer. The previous design team purchased and assembled these components. This is important because the driving parts increase the prototype costs by $684, which is the second largest expenditure, and the frame costs were $325.

Payback Period

The cost savings were estimated for a producer switching from manual spray techniques to SARGE. Since the crop yield and chemical costs would not change significantly, the labor makes up the greatest savings. Under the baseline conditions in Table 1, the annual labor costs of SARGE are $50 compared to nearly $300 for manual application. A producer's labor savings alone can pay for the robot in 6 years (Equation 1). Increased savings due to reduced health risks and ease of use would decrease the payback period. Of the variables that affect the payback period, the number of greenhouses is the most significant, see Tornado Diagram. The importance of the size of the operation makes SARGE especially beneficial for larger producers. Producers with over 15 greenhouses can recoup their investment in less than 2 years.

$$Payback = \frac{(P_{robot} - P_{manual})}{(N_{GH} * S * W)(t_{manual} - t_{robot})}$$

Equation 2

Production Variable	Baseline	Minimum	Maximum
Number of Greenhouses (N)	5	1	21
Sprays Per Year (S)	12	4	52
Manual Equipment (P_{manual})	$500	$200	$1,000
Manual Spraying Time (t_{manual})	1.0 h	0.5 h	2.0 h
Wage Rate (W)	$5.15/hr	$3.00/hr	$8.00/hr
Robot Price (P_{robot})	$2,000	-	-
Robot Setup Time (t_{robot})	0.2 h	-	-

Table 1: Sensitivity Variables

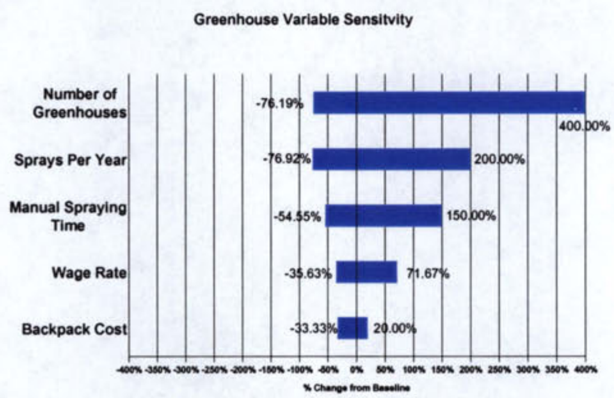


Figure 1: Tornado Diagram

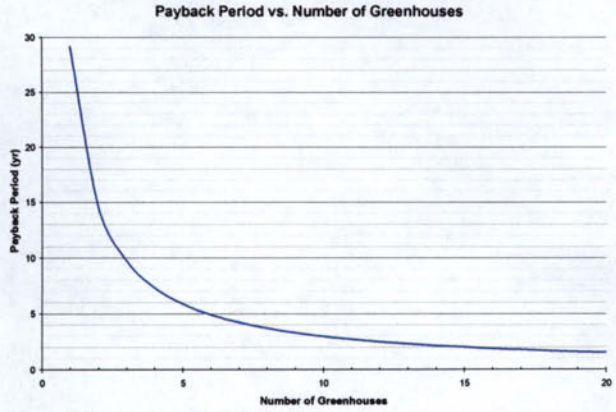


Figure 2: Payback vs. Number of Greenhouses

Literature Review

Appendix 1

Literature Review

1. Constant Application Rate

a. Title: Guidelines on Standards for Ag Pesticide Application Equipment and Related Test Procedures
Source: (Ag Library) S694.G851

b. Title: A Computerized, Small-Plot Spray Evaluation System with Variable Pipetter
Source: http://asae.frymulti.com/request2.asp?JID=3&AID=16056&CID=aeaj2004&v=20&i=3&T=1
Description: This literature gives the basic concepts for a computer program used to calculate the amount of insecticide to add to a given volume of water. Furthermore, it provides the overall system that integrates computation of proper insecticide volume, the use of a pipetter to accurately dispense water to each bottle, and the computerized, tractor-mounted spray system used for spray application.

c. Title: Data Acquisition System for Evaluating Spatial Accuracy of Selective Type Sprayers
Source: (Ag Library) Thesis 2002.S49

d. Title: Agricultural Pesticide Sprayers
Source: (Ag Library) SB953.A37 1998

e. Title: US Patent #5,755,382 – Self-Propelled Sprayer
Source: http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=/netahtml/srchnum.htm&r=1&f=G&l=50&s1=5,755,382.WKU.&OS=PN/5,755,382&RS=PN/5,755,382
Description: This is a GPS guided sprayer that uses a saddle tank for chemicals for spraying. The system also has configurable sprayer boom.

2. Vehicle Characteristics in Greenhouse Rows

a. Title: Enhancement of Turning Accuracy by Path Planning for Robot Tractor
Source: http://asae.frymulti.com/request2.asp?JID=1&AID=10029&CID=atoe2002&T=1
Description: Discussion of method for developing a turning algorithm for a robotic tractor using RTK and FOG.

b. Title: Chapter 4 Sensors and Controllers for Primary Drivers and Soil Engaging Implements
   Source: http://asae.frymulti.com/request2.asp?JID=7&AID=9453&CID=sd2002&T=1
   Description: The focus of this review is on the primary driver functions that involve soil contact. The review also covers other functions such as chemical spraying because these control objectives could conceivably be readily transferred to soil engaging systems. Outside of implement controllers, it is mostly about immediate sensing and control objectives such as machine speed, draft, and energy management.

3. Reliability

a. Title: American Society of Agricultural Engineers Standards for Test and Reliability Guidelines
   Source: ASAE EP 456 JAN01

4. Implementation

a. Title: Proceedings of Tenth International Symposium on Intelligent Control
   Source: http://ieeexplore.ieee.org/search/freesrchabstract.jsp?arnumber=525028&isnumber=11509&punumber=3999&k2dockey=525028@ieeecnfs&query=robot+guidance&pos=10&arSt=&ared=&arAuthor=
   Description: Discussion of robot guidance, decision-making, neural nets, fuzzy logic control, and visual sensing.

b. Title: A Low-Cost Robot Guidance Vision System
   Source: http://ieeexplore.ieee.org/search/freesrchabstract.jsp?arnumber=181289&isnumber=4579&punumber=1718&k2dockey=181289@ieecnfs&query=robot+guidance&pos=0&arSt=5/1&ared=5/3&arAuthor=Beattie%2C+R.J.%3B
   Description: Discussion of a system used in tool applications to operate on a visual basis using lasers and video images.

c. Title: Development of Master-Slave Robot System – Obstacle Avoidance Algorithm
   Source: http://asae.frymulti.com/request2.asp?JID=1&AID=10065&CID=atoe2002&T=1
   Description: Discussion of robots communicating with each other and decision-making to avoid a moving obstacle.

d. Title: Autonomous Robotic Vehicle for Greenhouse Spraying
   Source:
   http://asae.frymulti.com/request2.asp?JID=5&AID=16729&CID=can2004&v=&i=&T=1
   Description: Discussion of greenhouse sprayer created using a fuzzy logic system to navigate using ultra-sonic sensors. The sprayer incurred errors in travel of less than 1 inch.

e. Title: Machine Vision Based Guidance System for Automatic Rice Transplanters
   Source:
   http://asae.frymulti.com/request2.asp?JID=3&AID=12726&CID=aeaj2003&v=19&i=1&T=1
   Description: Description of machine that analyzes visual data of rice fields and makes decisions in 3 to 4 seconds.

f. Title: Development of Robot Tractor Based on RTK-GPS and Gyroscope
   Source:
   http://asae.frymulti.com/request2.asp?JID=5&AID=7297&CID=sca2001&v=&i=&T=1
   Description: Discussion of uses of RTK, FOG, and IMU through a sensor fusion algorithm. RMS error of less than 3 cm occurred in deviance from the path.

g. Title: Machine Vision Based Guidance System for an Agricultural Small-Grain Harvester
   Source:
   http://asae.frymulti.com/request2.asp?JID=3&AID=13945&CID=t2003&v=46&i=4&T=1
   Description: Discussion of guidance algorithm using monochrome cameras and traveling at a rate of 3 mph.

h. Title: A Method of Evaluating Different Guidance Systems
   Source:
   http://asae.frymulti.com/request2.asp?JID=5&AID=9153&CID=cil2002&v=&i=&T=1
   Description: Comparison of commercially available guidance systems.

i. Title: DGPS-Based Guidance of High-Speed Application Equipment
   Source:http://asae.frymulti.com/request2.asp?JID=5&AID=7370&CID=sca2001&v=&i=&T=1
   Description: Compares several modern guidance systems that have been developed to assist operators in steering along parallel paths. In this article they proved that guidance errors were least when a

low-cost Differential Global Positioning System (DGPS)-based light bar was used.

j. Title: Joint Architecture for Unmanned Ground Systems (JAUGS) Applied to Autonomous Agricultural Vehicles
Source: http://asae.frymulti.com/request2.asp?JID=1&AID=10033&CID=atoe2002&T=1
Description: This article provides information concerning JAUGS. JAUGS is a messaging standard providing a structure that enables components from many sources to communicate with each other. Plus it is interesting to note that JAUGS is built on the idea that developers will not be limited in the future as platforms evolve, applications change, and technology expands.

k. Title: US Patent #4,500,970 – Robot Vehicle Guidance System Including Checkpoint Realignment System
Source: http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=/netahtml/srchnum.htm&r=1&f=G&l=50&s1=4500970.WKU.&OS=PN/4500970&RS=PN/4500970
Description: This is a dead reckoning guidance system with a "pre-selected guidepath." The system uses guidance signals to realign itself on the path.

l. Title: US Patent #4,530,056 – Automated Guided Vehicle System
Source: http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=/netahtml/srchnum.htm&r=1&f=G&l=50&s1=4,530,056.WKU.&OS=PN/4,530,056&RS=PN/4,530,056
Description: This system uses a wire guidepath to move the vehicle from station to station. It also uses digital patterns to tell the vehicle additional information.

m. Title: US Patent #4,790,402 – Automated Guided Vehicle
Source: http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=/netahtml/srchnum.htm&r=1&f=G&l=50&s1=4,790,402.WKU.&OS=PN/4,790,402&RS=PN/4,790,402
Description: This system uses a pre-programmed path to guide the robot. A complex laser transmitter-receiver works with reflectors to keep the system on track.

n. Title: US Patent #4,817,000 – Automatic Guided Vehicle System
Source: http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=/netahtml/s

rchnum.htm&r=1&f=G&l=50&s1=4,817,000.WKU.&OS=PN/4,817,000&RS=PN/4,817,000

Description: This is a dead-reckoning system traveling over a predetermined path stored as path vectors. The system uses optical navigation and gyrocompass to correct the system.

o. Title: US Patent #5,213,176 – Self-Propelled Vehicle
Source: http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=/netahtml/srchnum.htm&r=1&f=G&l=50&s1=5,213,176.WKU.&OS=PN/5,213,176&RS=PN/5,213,176
Description: This is a self-propelled vehicle that is able to move in all directions on a floor. The patent outlines 3 and 4 wheel models.

p. Title: US Patent #6,044,183 – Robot Vision Using Target Holes, Corners, and Other Object Features
Source: http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=/netahtml/srchnum.htm&r=1&f=G&l=50&s1=6,044,183.WKU.&OS=PN/6,044,183&RS=PN/6,044,183
Description: This is a high-speed optical guidance system that uses objects in the environment to orient off of. It is envisioned for industrial robots.

q. Title: US Patent #6,389,329 – Mobile Robots and their Control System
Source: http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=/netahtml/srchnum.htm&r=1&f=G&l=50&s1=6,389,329.WKU.&OS=PN/6,389,329&RS=PN/6,389,329
Description: This is a robot guidance system using infrared beams emitted from a fixed station. The software algorithms to guide the robot back to the base station.

r. Title: US Patent #6,522,951 – Method and Device for Controlling a Robot
Source: http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=/netahtml/srchnum.htm&r=1&f=G&l=50&s1=6,522,951.WKU.&OS=PN/6,522,951&RS=PN/6,522,951
Description: This is a more general system that controls a robot on the basis from its own sensors. It allows the robot to use several different kinds of sensors for guidance.

s. Title: US Patent #6,671,582 – Flexible Agricultural Automation
Source: http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=/netahtml/s

rchnum.htm&r=1&f=G&l=50&s1=6,671,582.WKU.&OS=PN/6,671,582&RS=PN/6,671,582

Description: This is a general agricultural operation system that uses a variety of sensing systems and software programming to provide flexible solutions to labor intensive tasks.

t. Title: Basics about Inductive Track Guidance
Source: www.geotting.biz/en/inductive/intro_2.html
Description: General overview of sensors, signals, amplifiers, and control engineering of inductive track guidance. Different types of vehicles are discussed also.

5. Safety

a. Title: Automotive Radar for Adaptive Cruise Control and Collision Warning/Avoidance
Source: http://ieeexplore.ieee.org/search/freesrchabstract.jsp?arnumber=629084&isnumber=13678&punumber=4974&k2dockey=629084@ieecnfs&query=cruise+control&pos=0&arSt=16&ared=20&arAuthor=Eriksson%2C+L.H.%3B+As%2C+B.-O.%3B
Description: Discussion of autonomous intelligent cruise control (AICC) also known as adaptive cruise control (ACC). Article mentions testing of ACC in the Swedish car industry and the positive results.

b. Title: An Assessment of the Control and Safety Needs of Autonomous Agricultural Vehicles and Implement Systems on Farm Property
Source: http://asae.frymulti.com/request2.asp?JID=1&AID=10032&CID=atoe2002&T=1
Description: Discussion of safety of automated farm vehicles and an analysis of advantages, disadvantages, and costs of sensors and controls in this field.

c. Title: Personal Protective Equipment, Code of Federal Regulations 40 Part 170
Source: www.epa.gov/oppfead1/safety/workers/equip.htm
Description: The Worker Protection Standard requires that labels of pesticides used on farms, and forests, nurseries and greenhouses list the type of personal protective equipment that must be worn with each product.

d. Title: Pesticide Safety: A guide for gardeners and Homeowners
Source: www.ext.nodak.edu/extpubs/plantsci/hortcrop/ncr590w.htm

Description: General knowledge of pesticides are given regarding how to choose an appropriate pesticide, how to handle them properly, and how to store and dispose of them.

e. Title: Specifications for Alarm Systems Utilized in Agricultural Structures
   Source: ASAE Standards 2004 (ASAE S417.1 FEB03)
   Description: Establish specifications for fixed installation alarm systems utilized in agricultural structures. It also defines characteristics and requirements for components, wiring and service of fixed installation alarm systems.

Guidance Code

Appendix 2

```
'{$STAMP BS2sx}
'{$PBASIC 2.5}

'SARGE
'1 May 2005
'This code was tested in the greenhouse environment.  The proportional steering calcuations (x) are modified
'for the GH conditions.  The cornering routine is not exactly correct and the thresholds could be adjusted.

DIRS=$ff                                'Set all pins to be outputs
DIR2 = 0                                'Set pin 2 to be an input
DIR3 = 0                                'Set pin 3 to be an input

                                        'Define Stamp Pins
pwmcom CON 0                            'The serial communication pin to the PWM_Pal
mastercom PIN 1                         'The serial communication pin to the Master Stamp
manthrot PIN 2                          'The manual throttle data line from controller
manstr PIN 3                            'The manual steering data line from controller
rrev PIN 4                              'Ground connection for the Right motor
lrev PIN 5                              'Ground connection for the Left motor
rfwd PIN 12                             'Power connection for Right motor
lfwd PIN 13                             'Power connection for Left motor
ad_data PIN 6                           'Data line to the A/D converter
ad_clk PIN 7                            'Clock line to the A/D converter
ad_cs PIN 8                             'Chip select to the A/D converter


                                        'Define data communication variables
masterbaud CON 30                       'Baudrate to the Master Stamp - 50kbps, non-inverted, 8-bit, no parity
pwmbaud CON 110                         'Baudrate to the PWM_Pal  - 19.2kbps, non-inverted, 8-bit, no parity

                                        'Define system variables
master VAR Byte                         'The serial communication FROM the Master Stamp
slave VAR Byte                          'The serial communication TO the Master Stamp

                                        'Define manual guidance variables
mandir VAR Bit                          'The current direction for manual guidance, 0=Backward, 1=Forward
manerr VAR Bit                          'Manual guidance controller error counter, 0=No Error, 1=Error

                                        'Define spraying variables
speed VAR Byte                          'The speed factor in the guidance calculation
loopcnt VAR Byte                        'Loop count, number of times loop has ran
config VAR Nib                          'Configuration string for A/D chip
v VAR Word                              'Voltage coming out of A/D chip
vlf VAR Word                            'Voltage from left inductor
vrt VAR Word                            'Voltage from right inductor
prt VAR Word                            'Power setting for Right motor, based on steering alone (Also pulse width of steering channel of manual guidance)
plf VAR Word                            'Power setting for Left motor, based on steering alone (Also pulse width of throttle channel of manual guidance)
onrt VAR Word                           'On time of Right motor (duty cycle)
offrt VAR Byte                          'Off time of Right motor (100-duty cycle)
onlf VAR Word                           'On time of Left motor (duty cycle)
offlf VAR Byte                          'Off time of Left motor (100-duty cycle)
guidthresh CON 50                       'Guidance Threshold, minimum voltage required from inductors for operation
hertz VAR Word                          'Number of teeth seen by speed sensor in a second
setfreq VAR Nib                         'Desired number of teeth to be seen by speed sensor in a second
x VAR Word                              'Distance from center position of wire (inches*1000)
spincount VAR Nib                       'Number of times spin routine has been implemented consecutively
'-----------------------------------------------------------------------------------------------------------------------------------------
start:                                  'Start Main Program
LOW rfwd                                'Turn off all motors
LOW lfwd
LOW lrev
LOW rrev

PAUSE 250                               'Wait for Master Stamp to enter menu mode
'-----------------------------------------------------------------------------------------------------------------------------------------
menu:
  'DEBUG CR,"Menu Mode"
  SEROUT pwmcom,pwmbaud,["!PWMSS",%00110011]        'Set PWM_Pal to enable motors 1 & 2
  SEROUT pwmcom,pwmbaud,["!PWMM1",0,0,100,0]        'Set Motor 1 to 0%
  SEROUT pwmcom,pwmbaud,["!PWMM2",0,0,100,0]        'Set Motor 2 to 0%
  LOW rfwd                                          'Turn off all motors
  LOW lfwd
  LOW lrev
  LOW rrev
  SERIN mastercom,masterbaud,[master]               'Recieve command string from master
  IF master.BIT7 = %0 THEN                          'If run/stop bit is zero, stop
    'DEBUG CR,"STOP"
    PAUSE 100
  ELSEIF master.NIB1 = %1000 THEN                   'If instructed to go to spray mode, go there
```

```
    'DEBUG CR,"Spray"
    slave = %00110001                          'Set status string to slave mode
    loopcnt = 0                                'Initialize Loop count variable
    READ 0,setfreq                             'Read desired wheel frequency from EEPROM
    setfreq = 8                                'Debugging override of SETFREQ
    PAUSE 250                                  'Wait for Master stamp to get into spray mode
    GOTO spray                                 'Go to spray mode
  ELSEIF master.NIB1 = %1010 THEN              'If instructed to go to setup mode, go there
    'DEBUG CR,"Setup"
    slave = %00110101                          'Set status string to setup mode
    GOTO setup                                 'Go to setup mode
  ELSEIF master.NIB1 = %1100 THEN              'If instructed to go to manual mode, go there
    GOTO manual                                'Go to manual mode
  ELSE                                         'If nothing is received, wait
    'DEBUG CR,"Wait"
    PAUSE 100
  ENDIF
GOTO menu                                      'Return to main menu

'------------------------------------------------------------------------------------------------------------
manual:
'DEBUG CLS,"Manual Mode"
PAUSE 150                                      'Wait for Master Stamp to enter manual mode
SEROUT pwmcom,pwmbaud,["!PWMSS",%00000000]     'Set PWM_Pal to disable motors 1 & 2
slave = %00111000                              'Set status string to run in manual mode
SEROUT mastercom,masterbaud,[slave]            'Send status string to master
SERIN mastercom,masterbaud,[master]            'Receive command string from master

IF master.NIB1 = %1100 THEN                    'If manual mode is recieved

  PULSIN manthrot,1,plf                        'Measure throttle pulse width from controller
  PULSIN manstr,1,prt                          'Measure steering pulse width from controller

  'DEBUG CR,"Throttle ",DEC throttle
  'DEBUG CR,"Steering ",DEC steering
  'PAUSE 500

  IF plf < 1000 THEN                           'If the pulse width is less than 1000, assume controller error
    IF manerr = 1 THEN                         'If this is the 2nd consecutive error, end manual mode
      LOW rrev                                 'Stop both motors
      LOW lrev
      LOW rfwd
      LOW lfwd
      PAUSE 1000                               'Wait for Master Stamp to listen
      slave = %00111100                        'Set status string to menu mode
      SEROUT mastercom,masterbaud,[slave]      'Send status string to master
      SERIN mastercom,masterbaud,[master]      'Receive command string from master
      GOTO menu                                'Go to main menu
    ELSE
      manerr = 1                               'If this is the 1st error, set error varible and continue
    ENDIF
  ELSE
      manerr = 0                               'If there is no error, reset error variable
  ENDIF
                                               'Make drive adjustments based on throttle/steering position
  IF plf > 2525 THEN                           'If throttle is greater than threshold, going forward
    IF mandir = 0 THEN                         'If just had been going backwards, stop and wait before going forward
      LOW rrev
      LOW lrev
      LOW rfwd
      LOW lfwd
      PAUSE 10
    ENDIF
    mandir = 1                                 'Set direction variable to forward
    LOW rrev                                   'Set reverse lines to low (enable forward motion)
    LOW lrev
    IF prt < 1240 THEN                         'If steering is less than threshold, turning left
      HIGH rfwd
      LOW lfwd
    ELSEIF prt > 2160 THEN                     'If steering is greater than threshold, turning right
      HIGH lfwd
      LOW rfwd
    ELSE                                       'Otherwise, going straight
      HIGH rfwd
      HIGH lfwd
    ENDIF
  ELSEIF plf < 1525 THEN                       'If throttle is less than threshold, going backwards
    IF mandir = 1 THEN                         'If just had been going forwards, stop and wait before going backwards
      LOW rrev
      LOW lrev
      LOW rfwd
```

```
      LOW lfwd
      PAUSE 10
    ENDIF
    mandir = 0                                   'Set direction variable to backward
    HIGH rrev                                    'Set forward lines to low (enable backward motion)
    HIGH lrev
    IF prt < 1240 THEN                           'If steering is less than threshold, turning right
      LOW rfwd
      HIGH lfwd
    ELSEIF prt > 2160 THEN                       'If steering is greater than threshold, turning left
      LOW lfwd
      HIGH rfwd
    ELSE                                         'Otherwise, go straight
      LOW lfwd
      LOW rfwd
    ENDIF
  ELSE                                           'If throttle is not activated, stop everything
    LOW rrev
    LOW lrev
    LOW rfwd
    LOW lfwd
  ENDIF
  GOTO manual                                    'Return to manual loop
ELSE                                             'If manual mode is not recieved, go to menu
  GOTO menu                                      'Go to main program menu
ENDIF
'---------------------------------------------------------------------------------------------------------------------------------------------------------------------
spray:
SEROUT mastercom,masterbaud,[slave]              'Send status string to master
SERIN mastercom,masterbaud,[master]              'Receive command string from master
'DEBUG CR,"Spray Mode"


IF master.NIB1 = %1000 THEN                      'If command string says to run in spray mode, continue

  IF loopcnt = 0 THEN                            'If this is the first loop, load EEPROM and set PWM
    DIR12 = 0                                    'Set direction of pin 12 to input (allows PWM to control pin)
    DIR13 = 0                                    'Set direction of pin 13 to input (allows PWM to control pin)
    SEROUT pwmcom,pwmbaud,["!PWMSS",%00110011]   'Set PWM_Pal to enable motors 1 & 2
    SEROUT pwmcom,pwmbaud,["!PWMSP",%00001000]   'Set PWM_Pal Counter/Phase to Enable Counter #4

    speed = 10                                   'Set speed variable to 1

    SEROUT pwmcom,pwmbaud,["!PWMX4"]             'Clear counter on Counter #4 - Pin 11

    LOW rrev                                     'Set R Reverse Pin to low for forward travel
    LOW lrev                                     'Set L Reverse Pin to low for forward travel

    loopcnt = 1                                  'Set loop count variable to 1
  ENDIF
                                                 'Measure Left Voltage
  config = %1101                                 'Set A/D configuration (Start,Single Mode,Channel 0,MSBF)
  GOSUB Dataread                                 'Use Data Read Subroutine to collect data
  vlf = v                                        'Set output from Data Read Subroutine to right voltage variable
                                                 'Measure Right Voltage
  config = %1111                                 'Set A/D configuration (Start,Single Mode,Channel 1,MSBF)
  GOSUB Dataread                                 'Use Data Read Subroutine to collect data
  vrt = v                                        'Set output from Data Read Subroutine to left voltage variable

  IF (vrt < guidthresh+50) AND (vlf < guidthresh) THEN   'If there is no voltage from the inductors, stop the robot (right hand side is always higher, add 50mV)
    slave = %00010001                            'Set status string to guidance error
    SEROUT mastercom,masterbaud,[slave]          'Send status string to master
    SEROUT pwmcom,pwmbaud,["!PWMM1",0,0,100,0]   'Set Motor 1 to 0%
    SEROUT pwmcom,pwmbaud,["!PWMM2",0,0,100,0]   'Set Motor 2 to 0%
    GOTO menu                                    'Return to main program loop
  ELSE

    IF vlf >= vrt AND (vlf-vrt) <= 3440 THEN     'If the left voltage is greater than the right voltage
       x = 9219 + ((19*(vlf - vrt))/5)           'Calculate distance off of the wire, x is the distance from the wire with an offset to stay non-negative
       IF x < 19000 AND x > 411 THEN             'Make sure x is a reasonable number
         plf = 118 - ((48*(x/100))/100)          'Calculate duty cycle to left and right motors based on x data
         prt = ((48*(x/100))/100) + 22
       ELSE                                      'If x calculation results in unreasonable data, stop motors
         plf = 0
         prt = 0
       ENDIF
    ELSEIF vlf < vrt AND (vrt - vlf) <= 2426 THEN   'If the right voltage is greater than the left voltage
       x = 9219 - ((19*(vrt - vlf))/5)           'Calculate distance off of the wire, x is the distance from the wire with an offset to stay non-negative
       IF x < 19000 AND x > 411 THEN             'Make sure x is a reasonable number
         plf = 118 - ((48*(x/100))/100)          'Calculate duty cycle to left and right motors based on x data
         prt = ((48*(x/100))/100) + 22
```

```
  ELSE                                                      'If x calculation results in unreasonable data, stop motors
    plf = 0
    prt = 0
  ENDIF
ELSE                                                        'If voltage difference is too great to enter x equation
  plf = 99                                                  'Then correct for being off the wire
  prt = 5
ENDIF

IF plf > 100 THEN                                           'If power is greater than 100, set it to 99
  plf = 99
ENDIF

IF prt > 100 THEN                                           'If power is greater than 100, set it to 99
  prt = 99
ENDIF

'DEBUG CLS,"Right Voltage (mV): ",DEC vrt
'DEBUG CR,"Left Voltage (mV):  ",DEC vlf
'DEBUG CR,"Left Power %: ",DEC plf,CR,"Right Power %:",DEC prt
'DEBUG CR,"x =",DEC x
'PAUSE 250
                                                            'This is an elementary way of starting to deal with speed and steering together
onlf = ((plf*speed)/10)                                     'Calculate high signal for left motor
onrt = ((prt*speed)/10)                                     'Calculate high signal for right motor

'DEBUG CR,"onlf initial = ",DEC onlf
'DEBUG CR,"onrt initial= ",DEC onrt
'DEBUG CR,"speed = ",DEC speed
'DEBUG CR,"mastern1b0 = ",BIN master.NIB0

IF master.NIB0 = %0010 OR master.NIB0 = %0001 THEN          'If the lower nibble from the master indicates a turn, go into turn mode
  onrt = ((prt*8)/10)                                       'Set the duty cycle of each motor to 80% of what it would be otherwise (very slow)
  onlf = ((plf*8)/10)
ENDIF

IF onlf >= 100 THEN                                         'If the duty cycle is greater than 100%, set it to 99
  onlf = 99
ENDIF
IF onrt >= 100 THEN                                         'If the duty cycle is greater than 100%, set it to 99
  onrt = 99
ENDIF


offrt = 100-onrt                                            'Calculate off signal for right motor
offlf = 100-onlf                                            'Calculate off signal for left motor

'DEBUG CR,"x = ",DEC x
'DEBUG CR,"onlf new = ",DEC onlf
'DEBUG CR,"onrt new = ",DEC onrt

SEROUT pwmcom,pwmbaud,["!PWMM2",onrt.LOWBYTE,onrt.HIGHBYTE,offrt,0]  'Send pulse width parameters to PWM_Pal for Motor 1 (right)
SEROUT pwmcom,pwmbaud,["!PWMM1",onlf.LOWBYTE,onlf.HIGHBYTE,offlf,0]  'Send pulse width parameters to PWM_Pal for Motor 2 (left)

'Display voltages during driving
slave = %00110011                                           'Set status string to notify master of incoming data
SEROUT mastercom,masterbaud,[slave]                         'Send status string to master
PAUSE 5                                                     'Wait for master to begin listening again
SEROUT mastercom,masterbaud,[vrt.LOWBYTE,vrt.HIGHBYTE,vlf.LOWBYTE,vlf.HIGHBYTE,onrt.LOWBYTE,onlf.LOWBYTE]  'Send driving variables to master
PAUSE 150                                                   'Wait for master to begin listening again

IF loopcnt >= 5 THEN                                        'Read the counter every 5 loops ~1 sec

  loopcnt = 1                                               'Reset loop counter

  SEROUT pwmcom,pwmbaud,["!PWMC4"]                          'Set Counter #4  - Pin 11 to send counter value
  SERIN pwmcom,pwmbaud,[hertz.LOWBYTE,hertz.HIGHBYTE]       'Save counter value to variable
  SEROUT pwmcom,pwmbaud,["!PWMX4"]                          'Clear Counter #4 - Pin 11

  hertz = (hertz-1)                                         'Convert counter values into hertz, for some reason counter seems to add one extra

  IF hertz > 1000 THEN                                      'Throw out buffer overruns in counter
    hertz = 0
  ENDIF

  IF master.NIB0 = %0010 OR master.NIB0 = %0001 THEN
    speed = speed
  ELSE
    IF hertz <= 1 THEN                                      'If it hasn't moved, assume spinning, go to spin subroutine
      GOSUB spin
```

```
        ELSE
          spincount = 0                                 'If it has moved, reset spin counter
          IF hertz < (setfreq - 1)  THEN                'If below desired speed, increase motor speed
            speed = speed + 1
          ELSEIF hertz > (setfreq + 1) THEN             'If above desired speed, decrease motor speed
            speed = speed - 1
          ELSE                                          'If it is at desired speed +/- 1 hertz, stay the same
            speed = speed
          ENDIF
        ENDIF
      ENDIF

      'DEBUG CR,"Counter Value: ",DEC hertz," Hz"
      'DEBUG CR,"Speed:         ",DEC speed," %"
      'PAUSE 5000

    ENDIF

    loopcnt = loopcnt + 1                               'Increment loop count variable

'    DEBUG CR,"Loop Count: ",DEC loopcnt

    IF prt > 82 THEN                                    'If the steering component is greater than 82, inform Master Stamp with turning left status
      slave = %10110001
    ELSEIF plf > 82 THEN                                'If the steering component is greater than 82, inform Master Stamp with turning right status
      slave = %01110001
    ELSE                                                'If the steering component is greater than 82, inform Master Stamp with going straight status
      slave = %00110001
    ENDIF

  GOTO spray                                            'Return to spray loop
  ENDIF
ELSE                                                    'If spray mode is not recieved from master, go to main menu
  GOTO menu
ENDIF
'-----------------------------------------------------------------------------------------------------------------------------------------------
setup:
'DEBUG CR,"Waiting for Master"
SERIN mastercom,masterbaud,[master]                     'Receive command string from master
'DEBUG CR,"Master Received"
IF master.NIB1 = %1010 THEN                             'If command string says setup, save the included setfreq variable to EEPROM
  setfreq = master.NIB0
  'DEBUG CR,"SETFREQ = ",DEC setfreq
  WRITE 0,setfreq
  'DEBUG CR,"SETFREQ saved"
  PAUSE 100
ELSE
  GOTO menu                                             'If command string doesn't say setup, go to main menu
ENDIF
'-----------------------------------------------------------------------------------------------------------------------------------------------
DataRead:                                               'Begin Data Read Subroutine
LOW ad_cs                                               'Activate A/D
SHIFTOUT ad_data,ad_clk,LSBPRE,[config\4]               'Send configuration data to A/D
SHIFTIN ad_data,ad_clk,MSBPOST,[v\12]                   'Receive Measurement from A/D
HIGH ad_cs                                              'Deactivate A/D
v = ((v/819)*1000)+((((v//819)*10)/819)*100)+((((((v//819)*10)//819)*10)/819)*10)+(((((((v//819)*10)//819)*10)//819)*10)/819)  'Convert bits to mV (retain decimal places)
RETURN                                                  'Return To code
'-----------------------------------------------------------------------------------------------------------------------------------------------
spin:
SEROUT pwmcom,pwmbaud,["!PWMM1",99,0,1,0]               'Turn both motors full on
SEROUT pwmcom,pwmbaud,["!PWMM2",99,0,1,0]
IF spincount >= 5 THEN                                  'If it has spun 5 times in a row, backup
  GOTO backup
ELSE
  spincount = spincount + 1                             'Increment spin count variable
ENDIF
PAUSE 250                                               'Wait to climb out of stuck state
RETURN
'-----------------------------------------------------------------------------------------------------------------------------------------------
backup:
SEROUT pwmcom,pwmbaud,["!PWMM1",0,0,100,0]              'Turn both motors off
SEROUT pwmcom,pwmbaud,["!PWMM2",0,0,100,0]
PAUSE 100                                               'Wait to prevent H-Bridge from blowing up
HIGH rrev                                               'Turn on reverse lines
HIGH lrev
PAUSE 1000                                              'Backup for 1 second
LOW rrev
LOW lrev
PAUSE 10000                                             'Wait 10 seconds
GOTO menu                                               'Go to the main menu
```

Master Code

Appendix 3

```
'{$STAMP BS2pe}
'{$PBASIC 2.5}

'SARGE
'1 May 2005

DIRS=$ff                                'Set all ports to be outputs
DIR4 = 0                                'Set pin 4 to be an input
DIR5 = 0                                'Set pin 5 to be an input
DIR6 = 0                                'Set pin 6 to be an input

                                        'Define Stamp Pins
lcdpin PIN 0                            'The serial communication pin to the LCD screen
slavecom PIN 1                          'The slave communication pin to the Slave Stamp
ad_data PIN 7                           'Data line on the A/D converter
ad_cs PIN 8                             'Chip select line on the A/D converter
ad_clk PIN 9                            'Clock line on the A/D converter
spkr PIN 11                             'Connection to speaker
'led PIN 12                             'Connection to LED
svr PIN 14                              'Connection to right solenoid valve
svl PIN 13                              'Connection to left solenoid valve
pmp PIN 15                              'Connection to pump

                                        'Define data communication variables
slavebaud CON 0                         'Baudrate to Slave Stamp - 50kbps, non-inverted, 8-bit, no parity
lcdbaud CON 16468                       'Baudrate to LCD - 9600 bps, inverted, 8-bit, no parity

                                        'Define system variables
keybrd VAR Nib                          'The data string read by keyboard
slave VAR Byte                          'The serial communication FROM Slave Stamp
master VAR Byte                         'The serial communication TO the Slave stamp
vrt VAR Word                            'Voltage level of Right inductor, (Also Battery Voltage, and Times for beeping)
vlf VAR Word                            'Voltage level of Left inductor, (Also Battery Level, and Lag for beeping)
prt VAR Byte                            'Power setting for Right motor
plf VAR Byte                            'Power setting for Left motor
message VAR Byte                        'LCD message read from EEPROM
addr VAR Word                           'Memory location of LCD message
length VAR Byte                         'Length of LCD message
strt VAR Word                           'Starting memory location for LCD message
sprdir1 VAR Nib                         'Direction of travel from Slave stamp now (also number of decimal places for setup subroutine)
sprdir2 VAR Nib                         'Direction of travel from Slave stamp last time
rtcount VAR Word                        'Counter for right turns (also intial value for setup routine)
lfcount VAR Word                        'Counter for left turns (also maximum value for setup routine)
stcount VAR Word                        'Counter for going straight (also application rate during setup routine)
turn VAR Bit                            'True/False if turning
rows VAR Nib                            'Number of rows in the Greenhouse
speed VAR Byte                          'The ground speed of the robot (ft/s, then converted to hertz)
boomstrt VAR Nib                        'Setup mode of the boom at the start
boomend VAR Nib                         'Setup mode of boom at the end
turncount VAR Byte                      'Counter for the number of turns made

                                        ' Define LCD Functions
clrLCD CON 12                           ' Clear entire LCD screen.
posCmd CON 16                           ' Position cursor.
backlight CON 14                        ' Turn on backlight
block CON 255                           ' Block symbol
'------------------------------------------------------------------------------------------------------------------------------------------------------------------
start:

PAUSE 500                               'Wait half a second for LCD screen to turn on

strt = 520                              'Clear the screen and turn on the backlight
GOSUB disp

STORE 2                                 'Change to Memory Page 2
READ 0,boomstrt                         'Read boomstrt variable from EEPROM
READ 10,boomend                         'Read boomend variable from EEPROM
READ 20,rows                            'Read rows variable from EEPROM

menu:

GOSUB sproff                            'Turn off the hydraulic system

strt = 0                                'Display the main menu screen
GOSUB disp

PAUSE 250                               'Wait for Slave Stamp to begin listening

ui:                                     'Begin user interface loop

GOSUB keyread                           'Check to see if any keys have been pressed
```

```
IF keybrd = 2 THEN                                      'keybrd=2 means the manual button has been pressed
  strt = 60                                             'Remind operator to turn on controller
  GOSUB disp
  PAUSE 2500                                            'Pause to display message
  strt = 95                                             'Notify operator that robot is changing modes
  GOSUB disp
  PAUSE 1000                                            'Pause to display message
  master = %11000000                                    'Set command string to run in manual mode
  SEROUT slavecom,slavebaud,[master]                    'Send command string to slave
  GOTO manual                                           'Go to manual subroutine
ELSEIF keybrd = 6 THEN                                  'keybrd=6 means the setup button has been pressed
  PAUSE 250                                             'Wait for slave to begin listening
  master = %10100000                                    'Set command string to run in setup mode
  SEROUT slavecom,slavebaud,[master]                    'Send command string to slave
  GOTO setup                                            'Go to setup subroutine
ELSEIF keybrd = 5 THEN                                  'keybrd=5 means the spray button has been pressed
  GOSUB battread                                        'Check battery level from a/d
  strt = 315                                            'Display batter voltage and power level
  GOSUB disp
  SEROUT lcdpin,lcdbaud,[poscmd,95,DEC vrt DIG 4,DEC vrt DIG 3,".",DEC3 vrt]
  SEROUT lcdpin,lcdbaud,[poscmd,119,DEC vlf]
  FOR length = 1 TO 10                                  'Fill in the power meter (50% = 5 blocks and 5 underscores)
    IF length <= (vlf/10) THEN
      SEROUT lcdpin,lcdbaud,[poscmd,length+127,block]
    ELSE
      SEROUT lcdpin,lcdbaud,[poscmd,length+127,"_"]
    ENDIF
  NEXT
  PAUSE 1000                                            'Pause to display message
  IF vlf > 25 THEN                                      'If battery level is greater than 25%, OK to continue
    strt = 380                                          'Notify operator of satisfactory battery level
    GOSUB disp
    PAUSE 1000                                          'Pause to display message
    GOTO countdown                                      'Go to countdown
  ELSE                                                  'If battery level is less than 25%, too little power
    strt = 395                                          'Notify operator of unsatisfactory battery level
    GOSUB disp
    PAUSE 1000                                          'Pause to display message
    master = %00000000                                  'Set command string to stop in menu mode
    SEROUT slavecom,slavebaud,[master]                  'Send command string to slave
    GOTO menu                                           'Go to main menu
  ENDIF
ELSE                                                    'If no buttons have been pressed, or the 4th button which doesn't do anything
  master = %00000000                                    'Set command string to stop in menu mode
  SEROUT slavecom,slavebaud,[master]                    'Send command string to slave
ENDIF
GOTO ui                                                 'Go to user interface loop

'-----------------------------------------------------------------------------------------------------------------------------
manual:
strt = 120                                              'Display manual mode screen
GOSUB disp

GOSUB sproff                                            'Turn off hydraulic system

ui2:                                                    'Begin user interface loop

SERIN slavecom,slavebaud,[slave]                        'Receive status string from slave
PAUSE 10                                                'Wait for slave to begin listening
SEROUT slavecom,slavebaud,[master]                      'Send command string to slave (variable initially defined in main menu)

IF slave.NIB0 = %1000 THEN                              'If lower nibble of slave status indicates manual mode, everything is ok
  strt = 160                                            'Display success message on screen
  GOSUB disp

  GOSUB keyread                                         'Check to see if any keys have been pressed

  IF keybrd = 7 THEN                                    'If no buttons have been pressed, continue operation
    master = %1100000                                   'Set command string to run in manual mode
    GOTO ui2                                            'Return to user interface loop
  ELSE                                                  'If a button has been pressed, stop
    master = %00000000                                  'Set command sring to stop in menu mode
    GOTO menu                                           'Go to main menu
  ENDIF
ELSE                                                    'If lower nibble of slave status does not say manual mode, controller error
  master = %00000000                                    'Set command string to stop in menu mode
  PAUSE 10
  strt = 185                                            'Display controller error string
  GOSUB disp
  GOTO errormenu                                        'Go to error menu subroutine
```

```
ENDIF
'-------------------------------------------------------------------------------------------------------------------------------------------
spray:
strt = 230                                              'Display spray mode screen
GOSUB disp

IF boomstrt = 1 THEN                                    'If boom startup indicates, the left side should start off
  HIGH svr
  LOW svl
ELSEIF boomstrt = 2 THEN                                'If boom startup indicates, the right side should start off
  LOW svr
  HIGH svl
ELSEIF boomstrt = 3 THEN                                'If boom startup indicates, that both sides should start off
  LOW svr
  LOW svl
ELSE                                                    'Otherwise, open both valves
  HIGH svr
  HIGH svl
ENDIF

ui3:                                                    'Begin user interface loop
SERIN slavecom,slavebaud,[slave]                        'Recieve status string from slave
IF slave & %00111111 = %00110001 THEN                   'If the lower 6 bits of slave string indicates spray mode, check for keybrd
  PAUSE 10                                              'Wait for slave to begin listening
  SEROUT slavecom,slavebaud,[master]                    'Send command string to slave (variable initially defined in main menu)

  sprdir2 = sprdir1                                     'Save previous sprayer direction to sprdir2
  sprdir1 = slave.NIB1                                  'Save current direction to sprdir1

  IF sprdir1 = sprdir2 THEN                             'If the two directions are the same, nothing has changed
    IF sprdir1 = %0111 THEN                             'If the direction is right, add to right counter
      rtcount = rtcount + 1
    ELSEIF sprdir1 = %1011 THEN                         'If the direction is left, add to left counter
      lfcount = lfcount + 1
    ELSE                                                'If the direction is straight, add to straight counter
      stcount = stcount + 1
    ENDIF
  ELSE                                                  'Otherwise, it has changed directions, reset the counters
    rtcount = 0
    lfcount = 0
    stcount = 0
  ENDIF

  IF turn = 0  THEN                                     'If going straight, detect a turn
    IF lfcount >= 3 THEN                                'If there have been 3 lefts in a row, turning left
       'DEBUG CR,"Turning Left"
       turn = 1                                         'Activate turn mode
       master.NIB0 = %0010                              'Inform Slave stamp of turn mode
    ELSEIF rtcount >= 3 THEN                            'If there have been 3 rights in a row, turning right
       'DEBUG CR,"Turning Right"
       turn = 1                                         'Activate turn mode
       master.NIB0 = %0001                              'Inform Slave stamp of turn mode
    ELSE                                                'Otherwise going straight
       'DEBUG CR,"Going Straight"
        master.NIB0 = %0000                             'Inform Slave stamp of straight
    ENDIF
  ELSE                                                  'If already turning
    IF stcount>=25 THEN                                 'If there have been 25 straights in a row, finished turning
      'SEROUT lcdpin,lcdbaud,[poscmd,64,"Straight"]
      master.NIB0 = %0000                               'Inform Slave of straight
      turn = 0                                          'Deactivate turn mode
      turncount = turncount + 1                         'Increment # of turns
    ENDIF
  ENDIF

  IF turn=1 THEN                                        'If turning, turn off pump
    'LOW pmp
  ELSE                                                  'If not turning, turn on pump
    'HIGH pmp
  ENDIF

  IF turncount = (rows-1) THEN                          'If you are on the last row, check boom end variable
    IF boomend = 1 THEN                                 'If boom end indicates, the left side should end off
      LOW svl
    ELSEIF boomend = 2 THEN                             'If boom end indicates, the right side should end off
      LOW svr
    ELSEIF boomend = 3 THEN                             'If boom end indicates, both sides should end off
      LOW svr
      LOW svl
    ENDIF
  ELSEIF turncount > 1 THEN                             'If not on first or last row, turn on both valves
```

```
    HIGH svr
    HIGH svl
  ENDIF

  IF turncount >= rows THEN                                'If you've turned through all the rows, stop
    GOSUB sproff                                           'Turn of hydraulic system
    master.NIB1 = %0000                                    'Set command string to stop
    GOTO menu
  ENDIF

  GOSUB keyread                                            'Check to see if any buttons have been pressed
  IF keybrd = 7 THEN                                       'If no buttons have been pressed, continue
    master.NIB1 = %1000                                    'Set command string to run in spray mode
    GOTO ui3                                               'Go to user interface loop
  ELSE                                                     'If button has been pressed, end spraying
    master.NIB1 = %0000                                    'Set command string to stop in menu mode
    GOTO menu                                              'Go to main menu
  ENDIF
ELSEIF slave & %00111111 = %00110011 THEN                  'If lower 6 bits of slave string indicate more variables are coming, receive additional data
  SERIN slavecom,slavebaud,[vrt.LOWBYTE,vrt.HIGHBYTE,vlf.LOWBYTE,vlf.HIGHBYTE,prt,plf]          'Receive driving variables from slave
  strt = 485                                               'Display driving variables on screen
  GOSUB disp

  SEROUT lcdpin,lcdbaud,[poscmd,88,DEC vlf DIG 3,".",DEC3 vlf,poscmd,98,DEC vrt DIG 3,".",DEC3 vrt,poscmd,108,DEC plf,poscmd,118,DEC prt]
  master.NIB1 = %1000                                      'Set command string to run in spray mode
  GOTO ui3                                                 'Go to user interface
ELSEIF slave & %00111111 = %00010001 THEN                  'If lower 6 bits of slave string indicate guidance error, notify user
  GOSUB sproff                                             'Turn off hydraulic system
  master = %00000000                                       'Set command string to stop in menu mode
  strt = 550                                               'Display error message to operator
  GOSUB disp
  GOTO errormenu                                           'Go to error menu subroutine
ELSE                                                       'If nothing is received from slave, assume communication error
  master = %00000000                                       'Set command string to stop in menu mode
  GOTO menu                                                'Go to main menu
ENDIF
'-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------
setup:

GOSUB sproff                                               'Turn of hydraulic system

strt = 785                                                 'Display initial setup screen
GOSUB disp
PAUSE 1000                                                 'Pause to display message
strt = 825                                                 'Ask user how many rows
GOSUB disp
rtcount = rows                                             'Set initial value to that loaded from EEPROM
sprdirl = 0                                                'Set 0 decimal places
lfcount = 25                                               'Set maximum to 25 rows
GOSUB numinput                                             'Go to number input subroutine
strt = 880                                                 'Display successful row save message
GOSUB disp
SEROUT lcdpin,lcdbaud,[poscmd,93,DEC rtcount]
rows = rtcount                                             'Save subroutine variable to rows
PAUSE 1000                                                 'Pause to display the message
strt = 950
GOSUB disp                                                 'Ask user desired application rate
rtcount = 155                                              'Set initial value to 155 gal/ac
sprdirl = 0                                                'Set 0 decimal places
lfcount = 225                                              'Set maximum of 225 gal/ac
GOSUB numinput                                             'Go to number input subroutine
strt = 1010                                                'Display successful app rate save message
GOSUB disp
SEROUT lcdpin,lcdbaud,[poscmd,93,DEC rtcount]
stcount = rtcount                                          'Save application rate temporarily
PAUSE 1000                                                 'Pause to display message
strt = 1060                                                'As user the nozzle flowrate
GOSUB disp
rtcount = 8                                                'Set initial value to 0.08
sprdirl = 2                                                'Set 2 decimal places
lfcount = 50                                               'Set maximum value to 0.50
GOSUB numinput                                             'Go to number input subroutine
strt = 1120                                                'Display successful nozzle size save message
GOSUB disp
SEROUT lcdpin,lcdbaud,[poscmd,93,DEC rtcount DIG 2,".",DEC2 rtcount]
speed = ((((12*rtcount))*59)*11)/((stcount/10)*45)         'Calculate ground speed in ft/s based on app rate and nozzle size (T-Jet manual formula)
PAUSE 1000                                                 'Pause to display message
strt = 1165                                                'Display calculated speed value ft/s
GOSUB disp
SEROUT lcdpin,lcdbaud,[poscmd,93,DEC speed DIG 2,".",DEC2 speed]
PAUSE 1000                                                 'Pause to display message
```

```
speed = (speed*15)/100                       'Calculate desired hertz from speed (circumference of circle, 40 teeth/revolution)
'DEBUG "SETFREQ = ",DEC speed]
master.NIB1 = %1010                          'Set command string to setup mode
master.NIB0 = speed                          'Set lower half of command string to speed variable (setfreq on slave)
SEROUT slavecom,slavebaud,[master]           'Send command to slave
strt = 1205                                  'Ask user which side should be off at beginning
GOSUB disp
ui6:
GOSUB keyread                                'Check keyboard for user input
IF keybrd = 2 THEN                           'If the user selects left
  boomstrt = 1                               'Set boom start to left off
  strt = 1290                                'Display that left side will start off
  GOSUB disp
ELSEIF Keybrd = 6 THEN                       'If the user selects neither
  boomstrt = 4                               'Set boom start to neither
  strt = 1340                                'Display that neither side will start off
  GOSUB disp
ELSEIF keybrd = 5 THEN                       'If the user selects both
  boomstrt = 3                               'Set boom start to both
  strt = 1390                                'Display that both sides will start off
  GOSUB disp
ELSEIF keybrd = 4 THEN                       'If the user selects right
  boomstrt = 2                               'Set boom start to right
  strt = 1440                                'Display that right side will start off
  GOSUB disp
ELSE
  GOTO ui6                                   'Go back to user interface loop
ENDIF
PAUSE 1000                                   'Pause to display message
strt = 1490                                  'Ask user which side should be off at end
GOSUB disp
ui7:
GOSUB keyread                                'Check keyboard for user input
IF keybrd = 2 THEN                           'If user selects left
  boomend = 1                                'Set boom end to left
  strt = 1575                                'Display that left will end off
  GOSUB disp
ELSEIF Keybrd = 6 THEN                       'If user selects neither
  boomend = 4                                'Set boom end to neither
  strt = 1625                                'Display that neither will end off
  GOSUB disp
ELSEIF keybrd = 5 THEN                       'If user selects both
  boomend = 3                                'Set boom end to both
  strt = 1675                                'Display that both will end off
  GOSUB disp
ELSEIF keybrd = 4 THEN                       'If user selects right
  boomend = 2                                'Set boom end to right
  strt = 1725                                'Display that right will end off
  GOSUB disp
ELSE
  GOTO ui7                                   'Go back to user interface loop
ENDIF
STORE 2                                      'Change to Memory Page 2
WRITE 0,boomstrt                             'Save boom start to EEPROM
WRITE 10,boomend                             'Save boom end to EEPROM
WRITE 20,rows                                'Save rows to EEPROM
STORE 1                                      'Change to Memory Page 1
PAUSE 1000                                   'Pause to display message
strt = 1780                                  'Notify user that setup is complete
GOSUB disp
PAUSE 5000                                   'Pause to display message
GOTO menu                                    'Return to main menu
'-----------------------------------------------------------------------------------------------------------------------
battread:                                    'Begin Data Read Subroutine
HIGH ad_cs                                   'Deactivate A/D
LOW ad_cs                                    'Activate A/D
SHIFTIN ad_data,ad_clk,MSBPOST,[vrt\9]       'Receive Measurement from A/D
vrt = ((vrt/17)*1000)+((((vrt//17)*10)/17)*100)+((((((vrt//17)*10)//17)*10)/17)*10)+(((((((vrt//17)*10)//17)*10)//17)*10)/17)  'Convert bits to mV (retain decimal places)
HIGH ad_cs                                   'Deactivate A/D
vlf = ((vrt - 11650)*2)/27                   'Calculate battery level based on voltage reading
IF vlf > 100 THEN                            'Reset battery level if greater than 100%
  vlf = 100
ENDIF
'DEBUG CLS,"Voltage = ",DEC vrt
'DEBUG CR,"Power = ",DEC vlf,"%"
RETURN                                       'Return to code
'-----------------------------------------------------------------------------------------------------------------------
countdown:                                   'Begin countdown subroutine
'''MOVE THESE BLOCKS TO END OF COUNTDOWN TO ALLOW COUNTDOWN TO WORK
master = %10000000                           'Set command string to run in spray mode
SEROUT slavecom,slavebaud,[master]           'Send command string to slave
```

```
turn = 0                                               'Reset spraying variables
sprdir1 = 0
sprdir2 = 0
rtcount = 0
lfcount = 0
stcount = 0
GOTO spray                                             'Goto spray subroutine
'''ORIGINAL BEGINNING OF COUNTDOWN
GOSUB sproff                                           'Turn off hydraulic system
strt = 1815                                            'Display that spraying will begin in 60s
GOSUB disp
vlf = 1000                                             'Count down with increasing frequency and instances, each beep sequence last 10 secs
vrt = 5
GOSUB spkrout
SEROUT lcdpin,lcdbaud,[poscmd,140,"5"]
vlf = 500
vrt = 10
GOSUB spkrout
SEROUT lcdpin,lcdbaud,[poscmd,140,"4"]
vlf = 250
vrt = 20
GOSUB spkrout
SEROUT lcdpin,lcdbaud,[poscmd,140,"3"]
vlf = 125
vrt = 40
GOSUB spkrout
SEROUT lcdpin,lcdbaud,[poscmd,140,"2"]
vlf = 63
vrt = 80
GOSUB spkrout
SEROUT lcdpin,lcdbaud,[poscmd,140,"1"]
HIGH spkr
PAUSE 10000
LOW spkr
SEROUT lcdpin,lcdbaud,[poscmd,140,"0"]
'---------------------------------------------------------------------------------------------------------------------------
spkrout:                                                 'Begin speaker out subroutine
FOR length = 1 TO vrt
  GOSUB keyread
  HIGH spkr
  PAUSE vlf
  LOW spkr
  GOSUB keyread
  PAUSE vlf
NEXT
RETURN
'---------------------------------------------------------------------------------------------------------------------------
keyread:                                                 'Begin keyread subroutine
keybrd = INB & %0111                                     'Check 3 lower bits of 2nd nibble
'DEBUG CR,"Keyboard ",DEC keybrd
IF keybrd = 3  THEN                                      'If stop button is pressed, go to stop subroutine
  GOTO stp                                               'Go to stop subroutine
ENDIF
RETURN
'---------------------------------------------------------------------------------------------------------------------------
stp:                                                     'Begin stop subroutine
GOSUB sproff                                             'Turn of hydraulic system
strt = 435                                               'Display STOP message
GOSUB disp
master = %0000000                                        'Set command string to stop in menu mode
SERIN slavecom,slavebaud,2500,timeout,[slave]            'Receive status string from slave
PAUSE 10                                                 'Wait for slave to begin listening
SEROUT slavecom,slavebaud,[master]                       'Send command string to slave
timeout:
FOR length = 1 TO 10                                     'Sound speaker and send command string ten times
  SEROUT slavecom,slavebaud,[master]
  HIGH spkr
  PAUSE 50
  LOW spkr
  PAUSE 950
NEXT
GOTO menu                                                'Go to main menu
'---------------------------------------------------------------------------------------------------------------------------
sproff:                                                  'Begin sprayer off subroutine
LOW svr                                                  'Turn off SVR
LOW svl                                                  'Turn off SVL
LOW pmp                                                  'Turn off pump
RETURN                                                   'Return to code
'---------------------------------------------------------------------------------------------------------------------------
errormenu:                                               'Begin error menu subroutine
GOSUB keyread                                            'Check if keys have been pressed
```

```
IF keybrd = 4 THEN                                          'If menu button is pressed, go to main menu
  master = %000000                                          'Set command string to stop in menu mode
  GOTO menu                                                 'Go to main menu
ELSE                                                        'If menu button is not pressed, check again
  GOTO errormenu
ENDIF
'-------------------------------------------------------------------------------------------------------------------
disp:                                                       'Display subroutine
STORE 1                                                     'Change to Memory Page #1
READ strt,length                                            'Find the length of the message in the first memory location after start
FOR addr = (strt+1) TO (strt+length)                        'Read each byte from EEPROM beginning at the start point, and going until the length
  READ addr,message                                         'Read the message byte
  SEROUT lcdpin,lcdbaud,[message]                           'Send message to LCD screen
NEXT
RETURN
'-------------------------------------------------------------------------------------------------------------------
numinput:                                                   'Number input subroutine for setup
strt = 925                                                  'Display lower button definitions
GOSUB disp
ui5:
GOSUB keyread
IF keybrd = 5 AND rtcount <= (lfcount-1) THEN               'If up button is pressed, and not at maximum, increment up 1
  rtcount = rtcount + 1
ELSEIF keybrd = 6 AND rtcount >= 2 THEN                     'If down button is pressed, and not at 1, decline by 1
  rtcount = rtcount - 1
ELSEIF keybrd = 4 THEN                                      'If done is pressed return to code
  RETURN
ENDIF
IF sprdirl = 0 THEN                                         'If no decimals, display current value
  SEROUT lcdpin,lcdbaud,[poscmd,126,"      ",poscmd,126,DEC rtcount]
ELSEIF sprdirl = 2 THEN                                     'If two decimals, format data string before displaying
  SEROUT lcdpin,lcdbaud,[poscmd,126,"      ",poscmd,126,DEC rtcount DIG 2,".",DEC2 rtcount]
ENDIF
PAUSE 50                                                    'Wait to prevent extra key presses
GOTO ui5                                                    'Return to user interface loop
```

```
' {$STAMP BS2pe}
' {$PBASIC 2.5}

'SARGE
'1 May 2005

'This progam loads the LCD display messages into the EEPROM of the Master stamp.  The BS2pe has 15 pages of memory for saving information, the program is on page 0
'and the LCD information has been saved on page 1.  Each data Write command first saves the length of the screen and then the string itself.  The disp subroutine is
'the same as in the Master program and is used for debugging purposes.  Essentially this program is loaded onto the Master stamp before the main master program.
'Since this saves to EEPROM, the stamp "should" only need to be loaded once.

lcdpin PIN 0
lcdbaud CON 16468

message VAR Byte
addr VAR Word
length VAR Byte
strt VAR Word

' Define LCD Functions
clrLCD CON 12       ' Clear entire LCD screen.
posCmd CON 16       ' Position cursor.
backlight CON 14    ' Turn on backlight
block CON 255       ' Block symbol

STORE 1

'WRITE 0,54,clrlcd,backlight,poscmd,65,"SARGE - Main Menu",posCmd,109,"Choose Mode",posCmd,125,"MAN",posCmd,130,"SET",posCmd,135,"SPR",posCmd,140,"---"
'WRITE 60,30,clrlcd,posCmd,64,"Make Sure Controller",poscmd,91,"is on"
'WRITE 95,20,posCmd,105,"Entering Manual..."
'WRITE 120,38,clrlcd,backlight,posCmd,67,"SARGE - Manual",posCmd,125,"END",posCmd,130,"END",posCmd,135,"END",posCmd,140,"END"
'WRITE 160,20,posCmd,105,"Manual Mode Active"
'WRITE 185,42,clrlcd,backlight,posCmd,66,"SARGE - ERROR",posCmd,105,"Error: Controller",posCmd,140,"MENU"
'WRITE 230,40,clrlcd,backlight,posCmd,66,"SARGE - Spraying",posCmd,125,"END",posCmd,130,"END",posCmd,135,"END",posCmd,140,"END"
'WRITE 275,36,clrlcd,poscmd,65,"  Position Robot       Over Wire"
'WRITE 315,64,clrlcd,poscmd,65,"Checking Battery",poscmd,85,"Voltage = ",poscmd,102,"V",poscmd,105,"Power Level = ",poscmd,122,"%",poscmd,125,"0% ",poscmd,139,"100%"
'WRITE 380,13,clrlcd,poscmd,89,"Battery OK"
'WRITE 395,38,clrlcd,poscmd,86,"Battery Too Low     Please Recharge"
'WRITE 435,6,clrlcd,2,"STOP"
'''WRITE 445,37,clrlcd,backlight,posCmd,67,"SARGE - Setup",posCmd,125,"END",posCmd,130,"END",posCmd,135,"END",posCmd,140,"END"
'WRITE 485,35,posCmd,84,"VLF=",poscmd,94,"VRT=",poscmd,104,"PLF=",poscmd,110,"%   ",poscmd,114,"PRT=",poscmd,120,"%  "
'WRITE 520,2,clrlcd,backlight
'WRITE 525,20,posCmd,105,"Warning: No Offset"
'WRITE 550,43,clrlcd,backlight,posCmd,66,"SARGE - ERROR",posCmd,105,"Error: No Guidance",posCmd,140,"MENU"
'WRITE 600,73,clrlcd,backlight,posCmd,67,"SARGE - Setup",poscmd,85,"Would you like to",poscmd,106,"load a saved
GH?",posCmd,125,"YES",posCmd,130,"NO",posCmd,135,"---",posCmd,140,"END"
'WRITE 675,58,clrlcd,backlight,posCmd,67,"SARGE - Setup",poscmd,84,"Which GH would you",poscmd,104,"like to load? (1-6)"
'WRITE 740,42,clrlcd,backlight,posCmd,67,"SARGE - Setup",poscmd,84,"Success: GH #",poscmd,112," loaded."
'WRITE 785,37,clrlcd,backlight,posCmd,67,"SARGE - Setup",poscmd,85,"Begin new GH setup"
'WRITE 825,51,clrlcd,backlight,posCmd,67,"SARGE - Setup",poscmd,84,"How many rows are",poscmd,106,"there? (1-25)"
'WRITE 880,42,clrlcd,backlight,posCmd,67,"SARGE - Setup",poscmd,84,"Success:    rows",poscmd,113,"saved"
'WRITE 925,20,poscmd,124,">>1",posCmd,133,"DWN",posCmd,137,"UP",posCmd,140,"DONE"
'WRITE 950,59,clrlcd,backlight,posCmd,67,"SARGE - Setup",poscmd,84,"What is the desired appl. rate? (gal/ac)"\
'WRITE 1010,45,clrlcd,backlight,posCmd,67,"SARGE - Setup",poscmd,84,"Success:     gal/ac",poscmd,113,"saved"
'WRITE 1060,54,clrlcd,backlight,posCmd,67,"SARGE - Setup",poscmd,84,"What is the nozzle",poscmd,104,"flowrate? (GPM)"
'WRITE 1120,43,clrlcd,backlight,posCmd,67,"SARGE - Setup",poscmd,84,"Success:      GPM",poscmd,113,"saved"
'WRITE 1165,36,clrlcd,backlight,posCmd,67,"SARGE - Setup",poscmd,85,"Speed =      ft/s"
'WRITE 1205,84,clrlcd,backlight,posCmd,67,"SARGE - Setup",posCmd,84,"Which side should be",posCmd,104,"off in the 1st
row?",posCmd,125,"LEFT",posCmd,130,"NONE",posCmd,135,"BOTH",posCmd,140,"RGHT"
'WRITE 1290,48,clrlcd,backlight,poscmd,67,"SARGE - SETUP",poscmd,84,"The left side will",poscmd,104,"start OFF"
'WRITE 1340,47,clrlcd,backlight,poscmd,67,"SARGE - SETUP",poscmd,84,"Neither side will",poscmd,104,"start OFF"
'WRITE 1390,45,clrlcd,backlight,poscmd,67,"SARGE - SETUP",poscmd,84,"Both sides will",poscmd,104,"start OFF"
'WRITE 1440,49,clrlcd,backlight,poscmd,67,"SARGE - SETUP",poscmd,84,"The right side will",poscmd,104,"start OFF"
'WRITE 1490,84,clrlcd,backlight,poscmd,67,"SARGE - Setup",poscmd,84,"Which side should be",poscmd,104,"off in the end
row?",posCmd,125,"LEFT",posCmd,130,"NONE",posCmd,135,"BOTH",posCmd,140,"RGHT"
'WRITE 1575,49,clrlcd,backlight,poscmd,67,"SARGE - SETUP",poscmd,84,"The left side will",poscmd,104,"finish OFF"
'WRITE 1625,48,clrlcd,backlight,poscmd,67,"SARGE - SETUP",poscmd,84,"Neither side will",poscmd,104,"finish OFF"
'WRITE 1675,46,clrlcd,backlight,poscmd,67,"SARGE - SETUP",poscmd,84,"Both sides will",poscmd,104,"finish OFF"
'WRITE 1725,50,clrlcd,backlight,poscmd,67,"SARGE - SETUP",poscmd,84,"The right side will",poscmd,104,"finish OFF"
'WRITE 1780,33,clrlcd,backlight,poscmd,67,"SARGE - SETUP",poscmd,87,"Setup Complete"
'WRITE 1815,39,poscmd,85,"Startup Sequence",poscmd,124,"Spray Begins in 60s"

strt = 1815
GOSUB disp

END

disp:
STORE 1
READ strt,length
FOR addr = (strt+1) TO (strt+length)
```

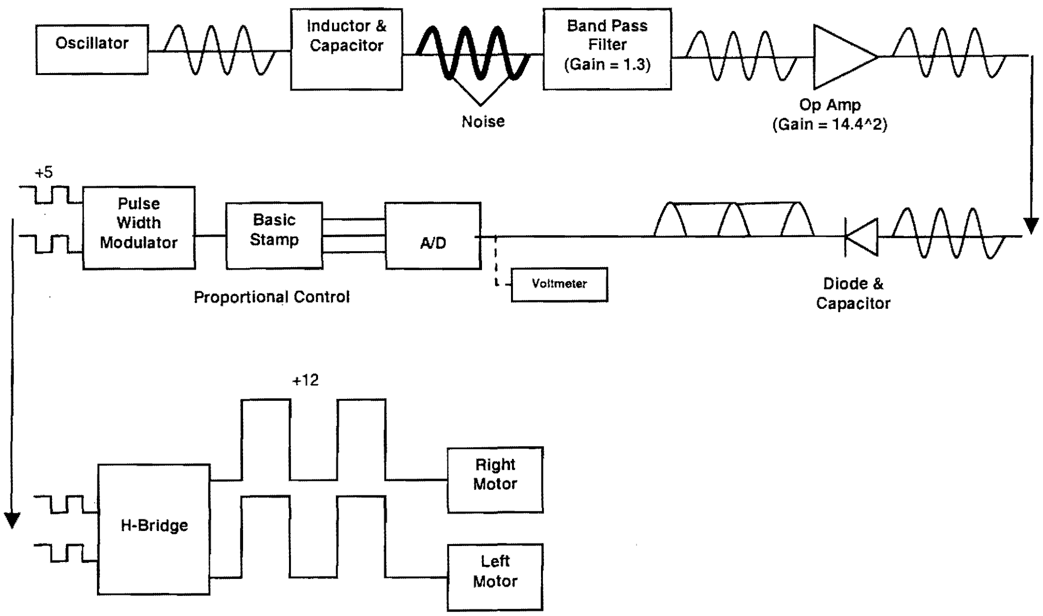
```
  READ addr,message
  SEROUT lcdpin,lcdbaud,[message]
  DEBUG message
NEXT
RETURN
```

Guidance Charts

Appendix 4

Automatic Sprayer System Guidance Options

Guidance System	Advantages	Disadvantages
Dead Reckoning (Pre-Programmed)	• Easy to setup • No infrastructure	• Compounding error terms • Memory management • Different setup in various greenhouses
GPS	• Established agriculture technology	• May not get adequate signal indoors
Inductive	• Inexpensive components • Inconspicuous • Clear path for the robot • Wire is off-the-shelf • Minimal infrastructure	• Integrated into the greenhouse (not self-contained) • Difficult to troubleshoot/replace • Changing infrastructure of house
Light (Laser/Infrared)	• Easy programming • Used in many situations	• Possible obstructions • Difficult to turn • Difficult to align transmitter/receiver
Machine Vision	• Easy to setup • No infrastructure	• High computer requirements • Difficult to programming • Clear markers for path following • Lighting variables
Radar/Ultrasonic	• No infrastructure	• Difficult programming • Possible changes in greenhouse affect performance • Difficult to turn
Radio	• Independent of obstructions	• Imprecise beams • Expensive transmitters • Interference • Difficult to turn • Difficult to align transmitter/receiver
Track/Cable	• Clear path for robot • No errors • "old" technology	• Expensive • Trip hazard • Dependent on greenhouse • Difficult installation

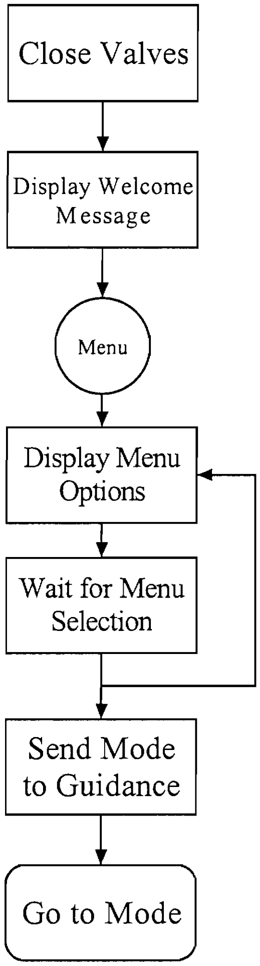

Oscillator
Inductor & Capacitor
Noise
Band Pass Filter (Gain = 1.3)
Op Amp (Gain = 14.4^2)
Diode & Capacitor
A/D
Voltmeter
Basic Stamp
Pulse Width Modulator
+5
Proportional Control
+12
H-Bridge
Right Motor
Left Motor

Flow Charts of Operating System

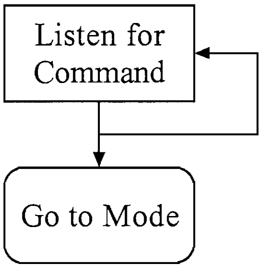
Appendix 5

Mode: Menu

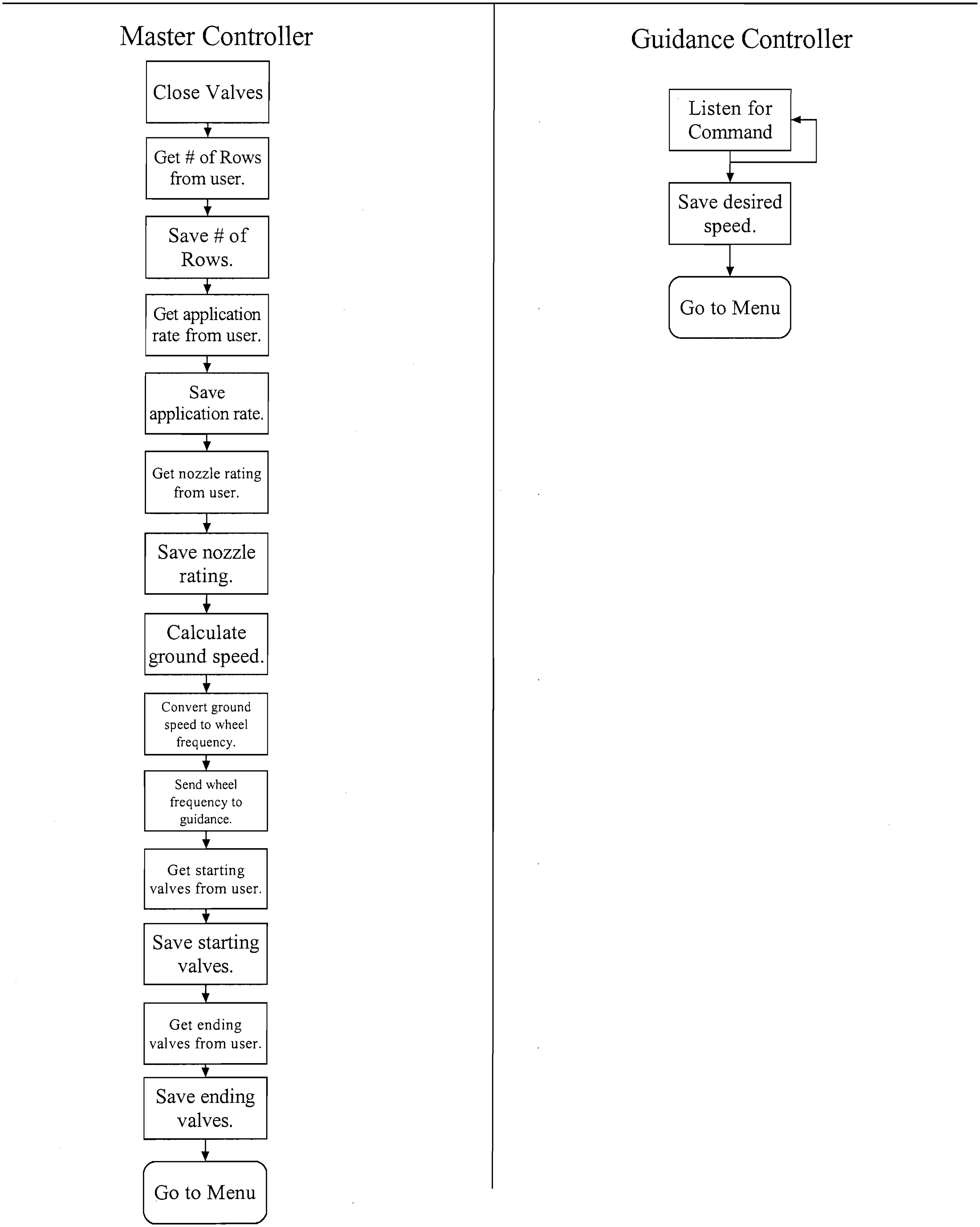
Master Controller



Guidance Controller

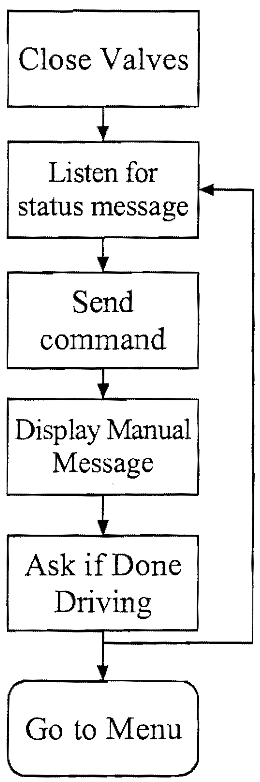


Mode: Setup

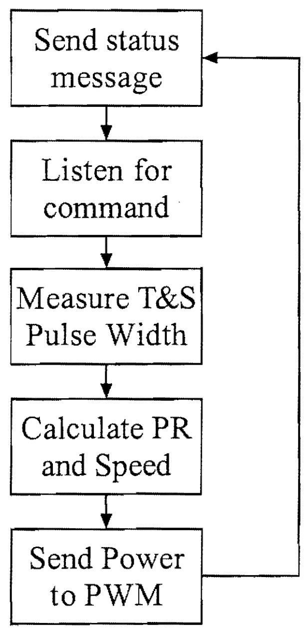

Master Controller
Close Valves
Get # of Rows from user.
Save # of Rows.
Get application rate from user.
Save application rate.
Get nozzle rating from user.
Save nozzle rating.
Calculate ground speed.
Convert ground speed to wheel frequency.
Send wheel frequency to guidance.
Get starting valves from user.
Save starting valves.
Get ending valves from user.
Save ending valves.
Go to Menu
Guidance Controller
Listen for Command
Save desired speed.
Go to Menu

Mode: Manual Guidance

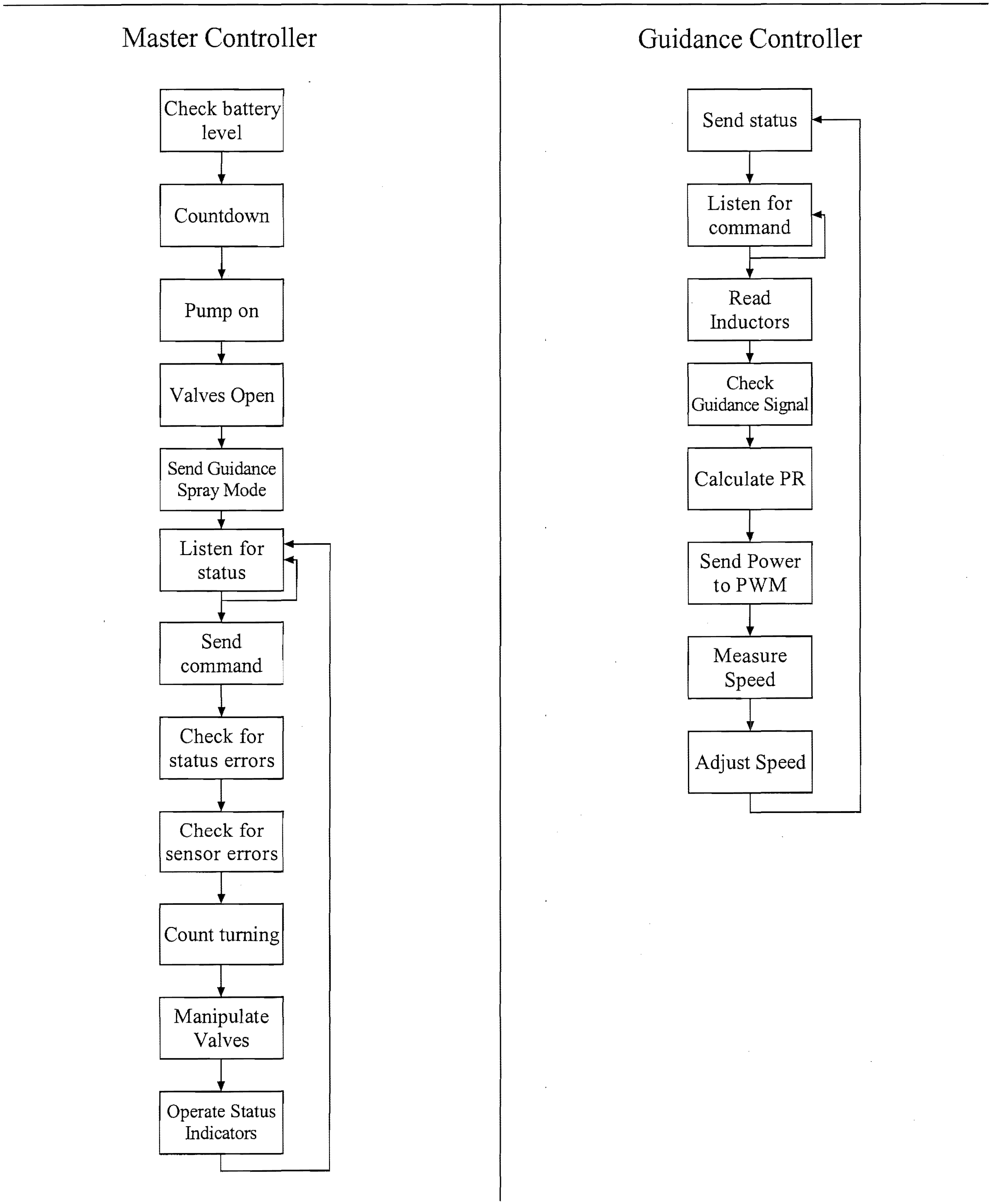
Master Controller



Guidance Controller



Mode: Spraying



Subroutines

Master Controller	Guidance Controller
ADC0831 Voltage Sampling	LTC1298 Voltage Sampling
Countdown	Spin Correction
Speaker Buzz	Backup on Stuck
Keyboard Check	
Stop	
Sprayer Off	
Error Menu	
Display from EEPROM	
Number Input	

Pin Definitions

Appendix 6

Basic Stamp - Pin Definitions
3/11/2005

Pin	Master Stamp	Guidance Stamp
0	LCD Serial	PWM Serial
1	Guidance Serial	Master Serial
2	FREE	Manual Guidance - Throttle
3	FREE	Manual Guidance - Steering
4	UI - Encoder A	H-Bridge Reverse - Right Motor
5	UI - Encoder B	H-Bridge Reverse - Left Motor
6	UI - Encoder C	Data - LTC1298
7	Data - ADC0831	Clock - LTC1298
8	Chip Select - ADC0831	Chip Select - LTC1298
9	Clock - ADC0831	PWM Counter 2 - FREE
10	FREE	PWM Counter 3 - FREE
11	Speaker	PWM Counter 4 - Speed Sensor
12	LED Indicator	PWM M1 - Right Motor
13	Valve Control - Right	PWM M2 - Left Motor
14	Valve Control - Left	PWM M3 - CAUTION
15	Pump Control	PWM M4 - CAUTION

Cable Standards

Appendix 7

Cable Connections

Battery Connector

2	+12	Red
3	Gnd	Black

Solenoid Valve Connector

1	SVR	+12	Red
2	SVR	Gnd	Black
3	SVL	+12	Black
4	SVL	Gnd	Red

Pump Connector

1	+12	Red
4	Gnd	Black

Motor Connector

1	MR	+12	Red
2	MR	Gnd	Black
3	ML	+12	Red
4	ML	Gnd	Black

SARGE Bus

H-Bridge B2	1	Black on Green
H-Bridge A2	2	Black on Orange
H-Bridge B1	3	Green
H-Bridge A1	4	Orange
Speaker	5	Black on White
LED	6	White
SVR	7	White on Black
SVL	8	Black on Blue
Pump	9	Blue
5V	12	White on Red
Gnd	13	Black
12V	14	Red

LCD Bus

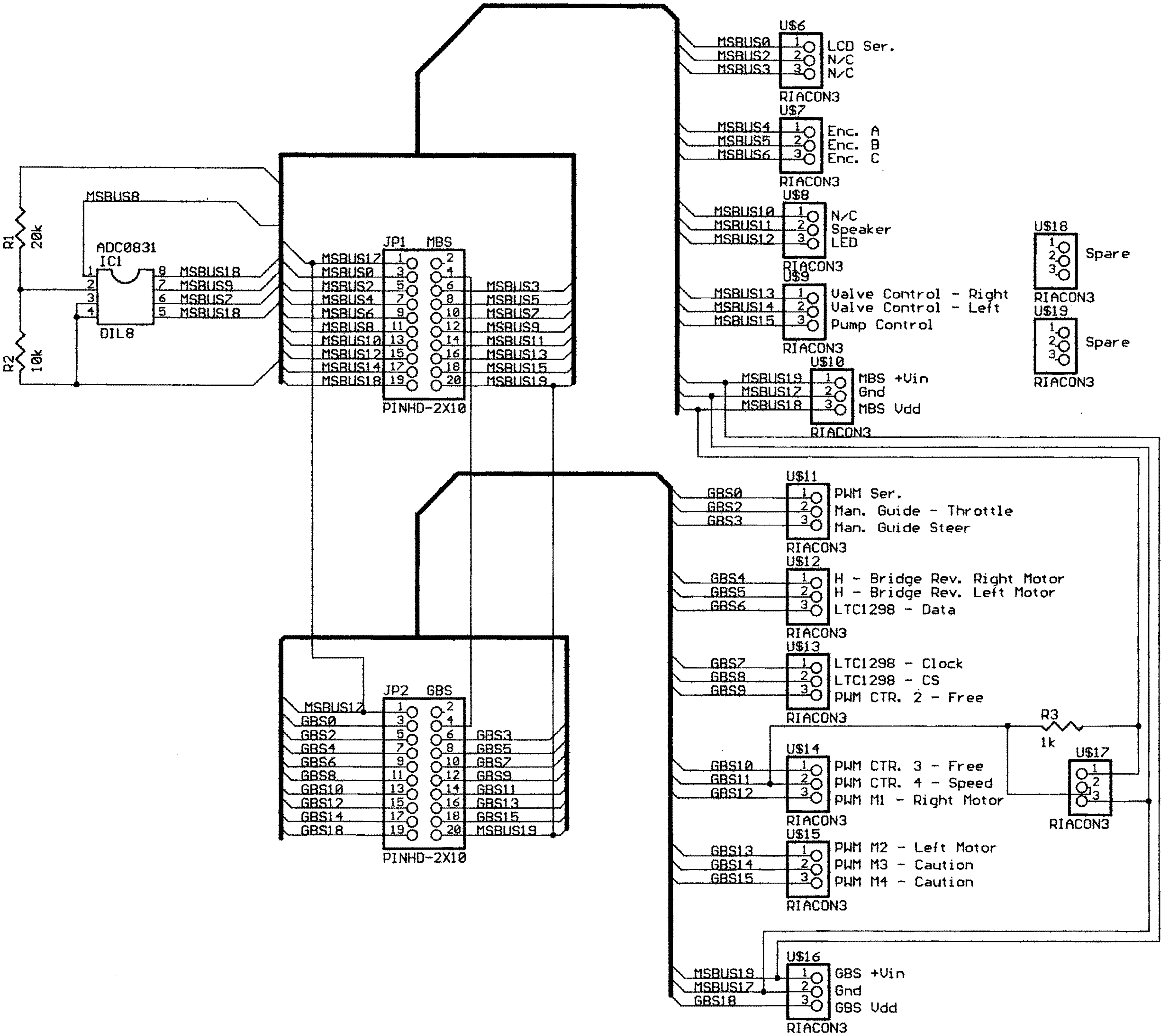
Master Pin 0	1	White
Master Pin 4	2	Purple
Master Pin 5	3	Yellow
Master Pin 6	4	Orange
Guidance Pin 2	5	Blue
Guidance Pin 3	6	Green
5V	7	Red
Gnd	8	Black

Primary Box

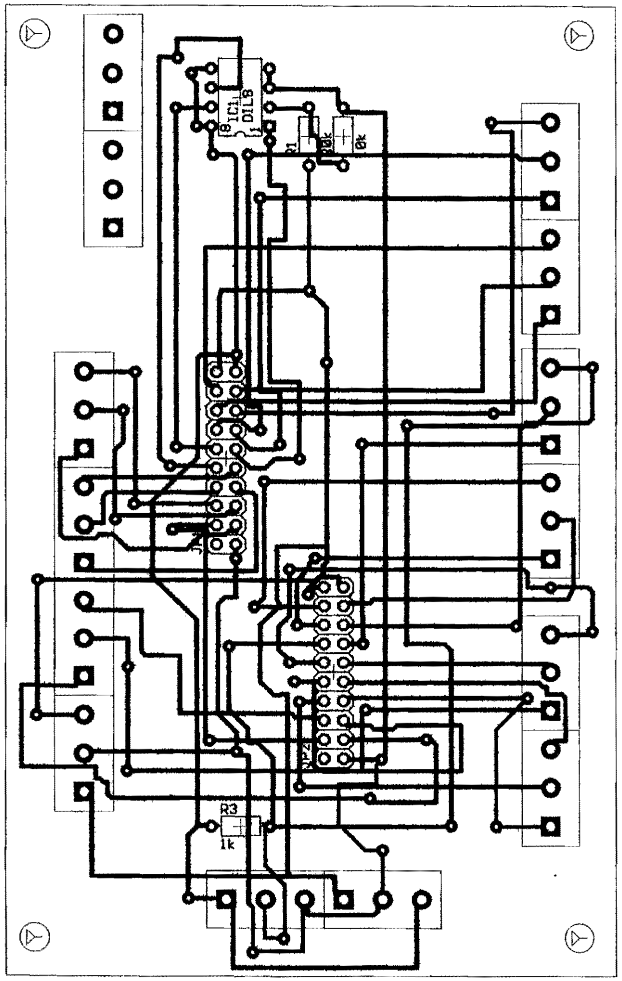
A/D data	Blue
A/D Clk	White on Blue
A/D CS	Black on Blue

Schematics of Boards

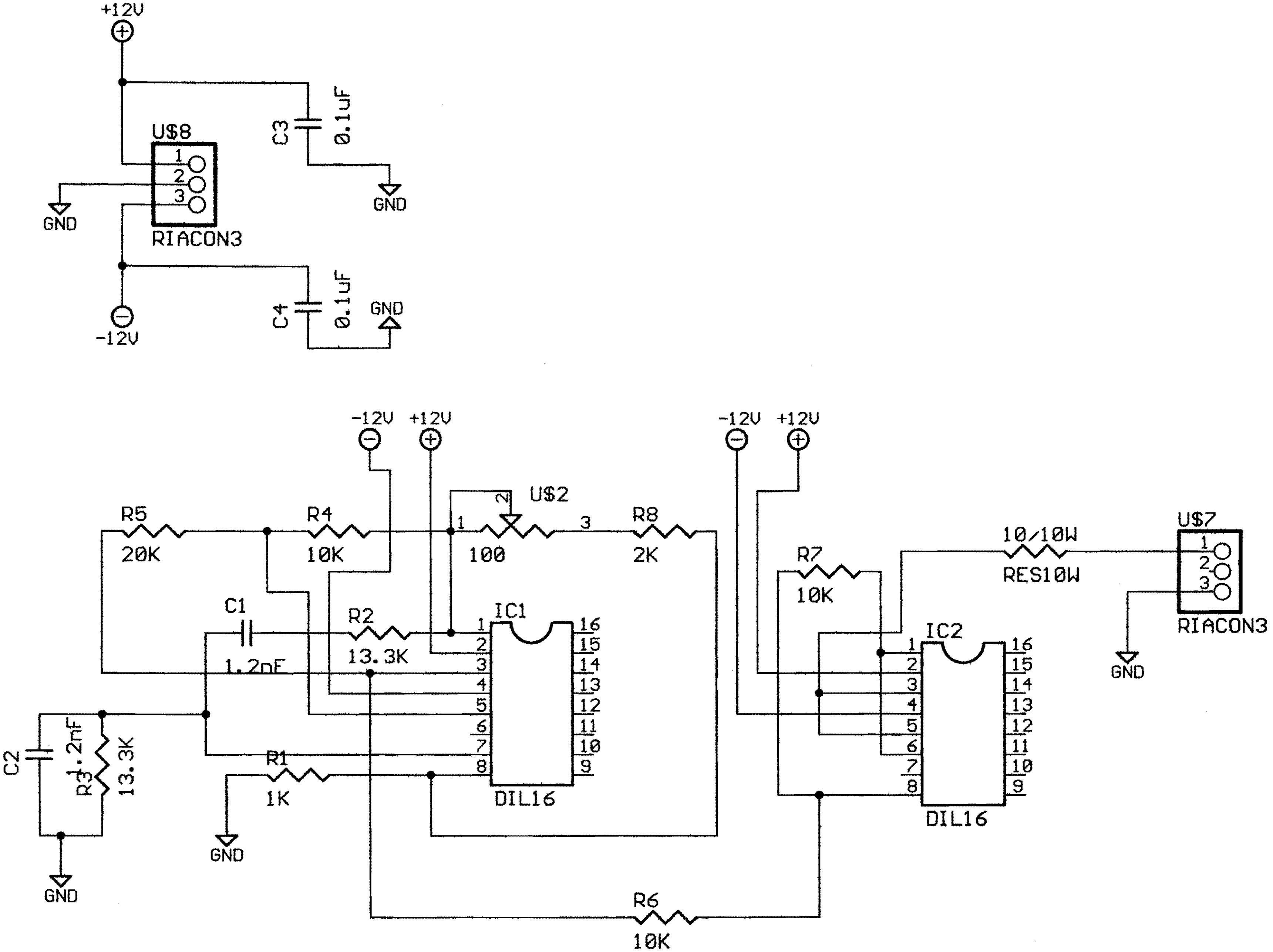
Appendix 8



5/02/2005 11:29:27a C:\Program Files\Eagle\Projects\Auto_Sprayer_Digital.sch (Sheet: 1/1)

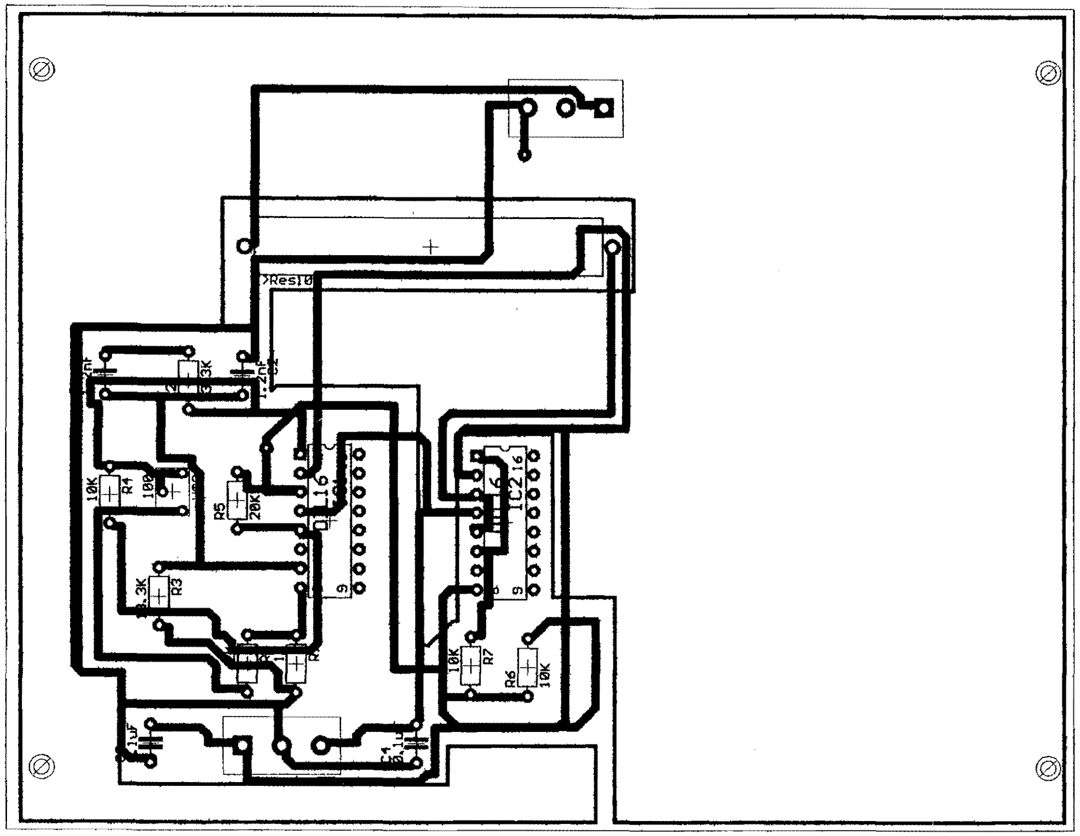


5/02/2005 11:30:12a C:\Program Files\Eagle\Projects\Auto_Sprayer_Digital.brd

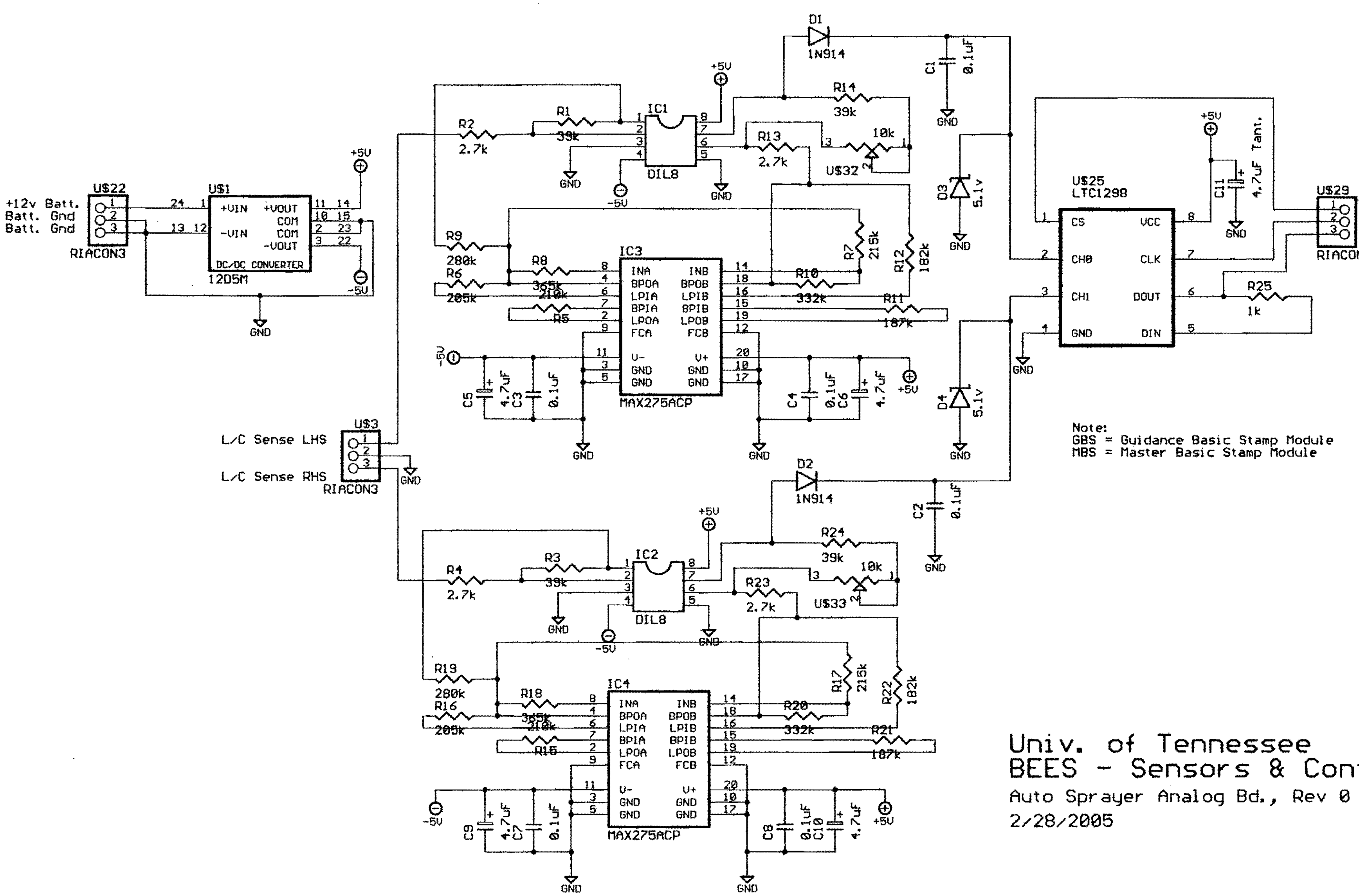


Univ. of Tennessee
BEES - Sensors & Controls Lab
Oscillator Board

2/28/2005

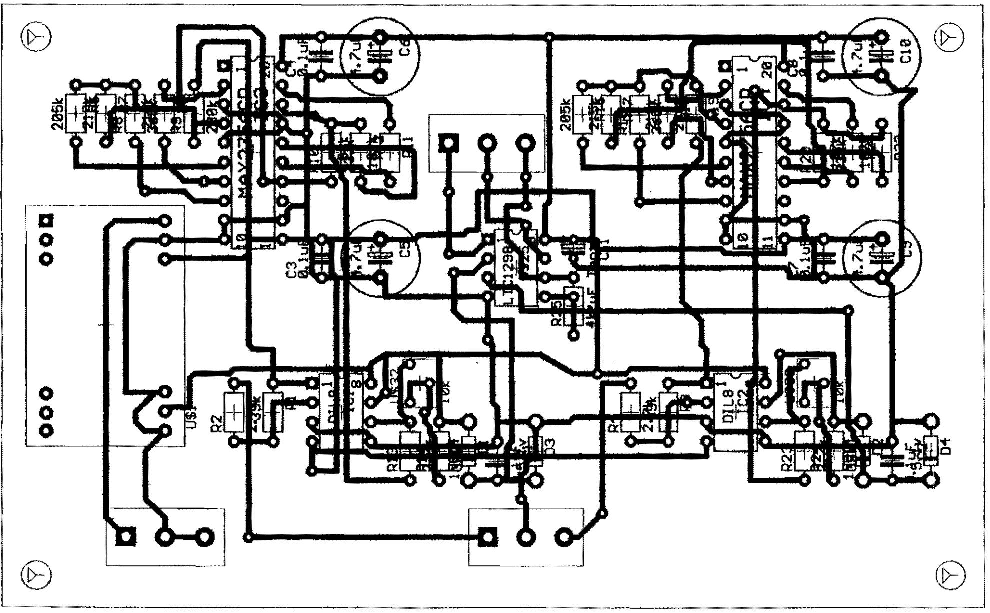


5/02/2005 11:45:16a C:\Program Files\Eagle\Projects\Sprayer_Osc_Rev1.brd

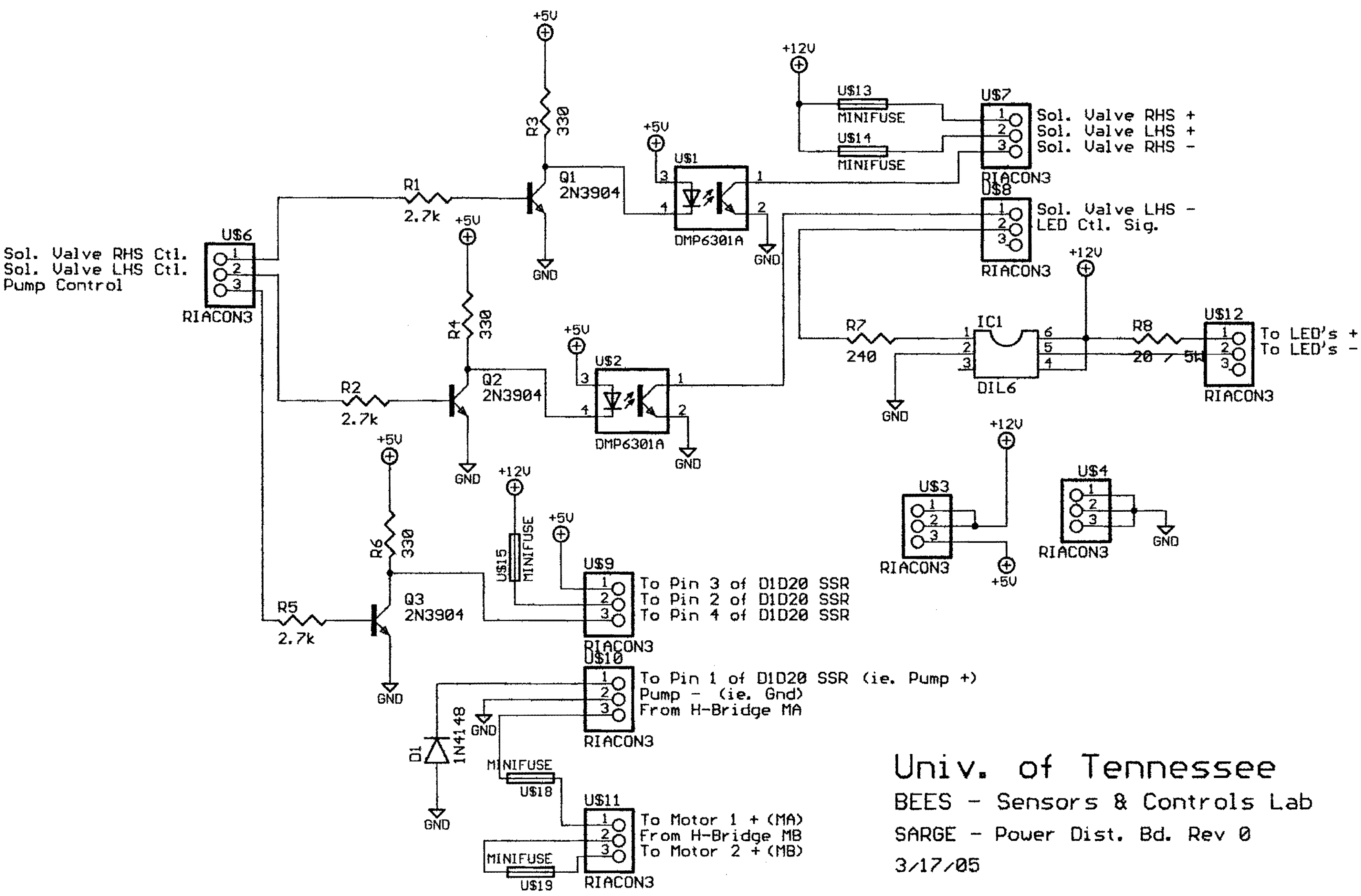


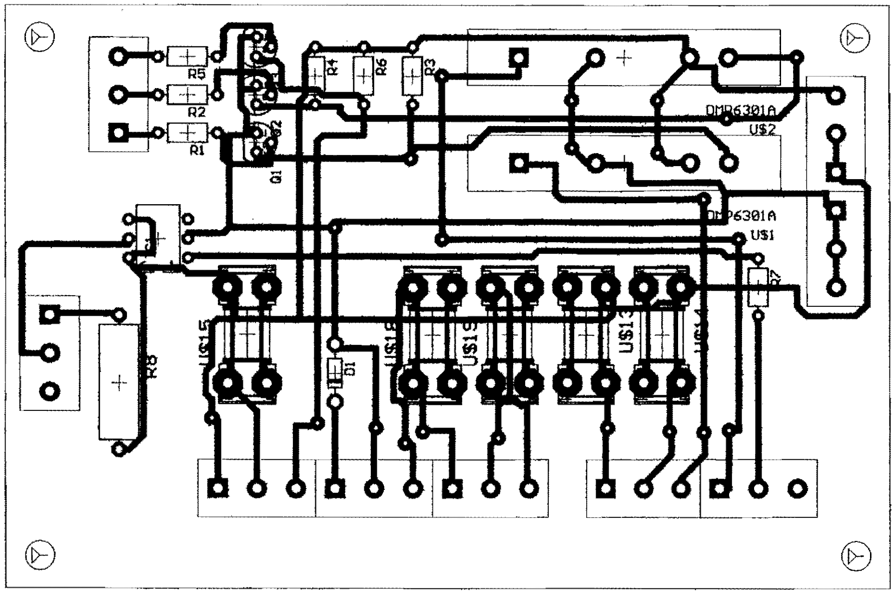
Univ. of Tennessee
BEES - Sensors & Controls Lab
Auto Sprayer Analog Bd., Rev 0
2/28/2005

5/02/2005 11:43:43a f=0.824 C:\Program Files\Eagle\Projects\AutoSprayer_Analog.sch (Sheet: 1/1)

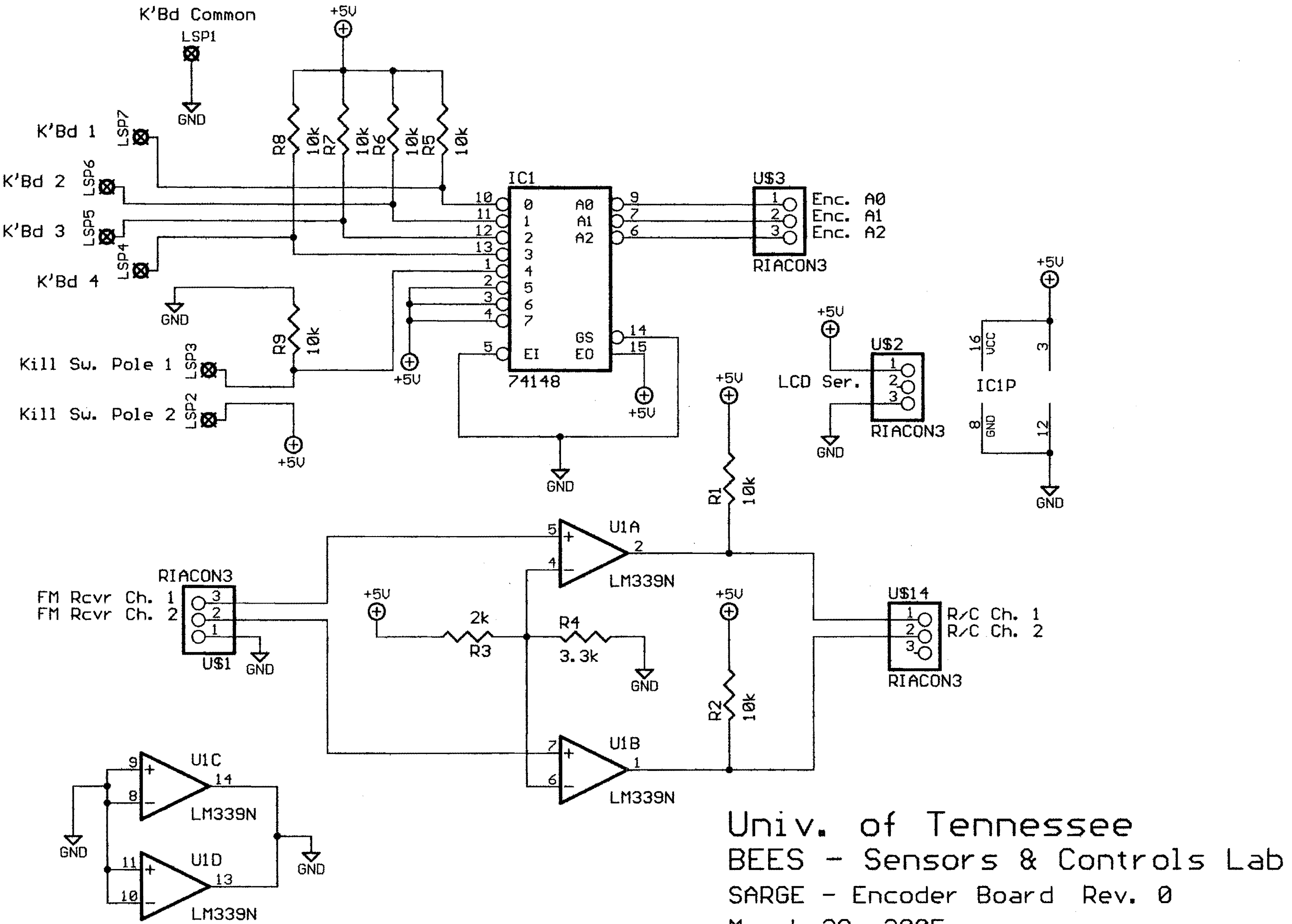


5/02/2005 11:43:28a C:\Program Files\Eagle\Projects\AutoSprayer_Analog.brd

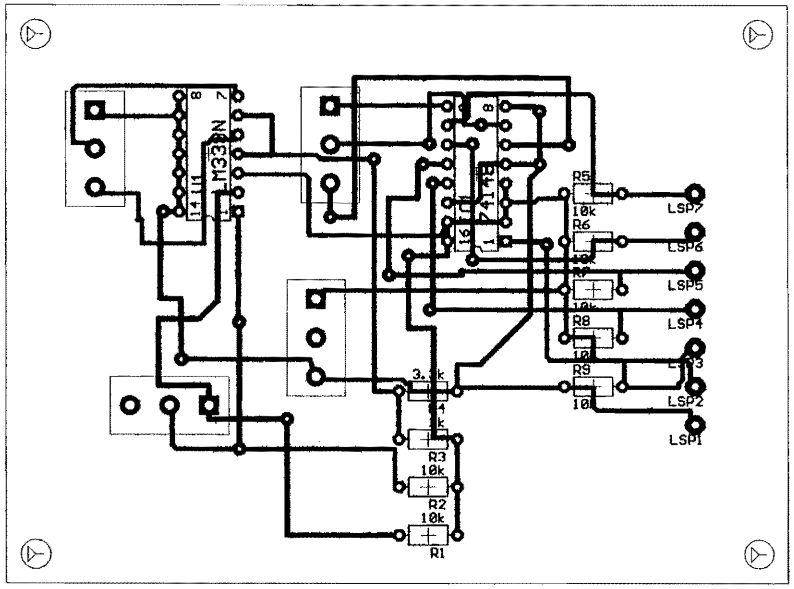

+5V
+12V
U$13
MINIFUSE
U$14
MINIFUSE
U$7
Sol. Valve RHS +
Sol. Valve LHS +
Sol. Valve RHS -
RIACON3
U$8
Sol. Valve LHS -
LED Ctl. Sig.
RIACON3
R3
330
Q1
2N3904
R1
2.7k
U$1
DMP6301A
GND
U$6
Sol. Valve RHS Ctl.
Sol. Valve LHS Ctl.
Pump Control
RIACON3
R4
330
Q2
2N3904
R2
2.7k
U$2
DMP6301A
R7
240
IC1
DIL6
R8
20 / 5W
U$12
To LED's +
To LED's -
RIACON3
R6
330
Q3
2N3904
R5
2.7k
U$15
MINIFUSE
U$9
To Pin 3 of D1D20 SSR
To Pin 2 of D1D20 SSR
To Pin 4 of D1D20 SSR
RIACON3
U$10
To Pin 1 of D1D20 SSR (ie. Pump +)
Pump - (ie. Gnd)
From H-Bridge MA
RIACON3
D1
1N4148
MINIFUSE
U$18
U$11
To Motor 1 + (MA)
From H-Bridge MB
To Motor 2 + (MB)
RIACON3
MINIFUSE
U$19
U$3
RIACON3
U$4
RIACON3
Univ. of Tennessee
BEES - Sensors & Controls Lab
SARGE - Power Dist. Bd. Rev 0
3/17/05



5/02/2005 11:39:16a C:\Program Files\Eagle\Projects\Auto_Sprayer_Power.brd



Univ. of Tennessee
BEES - Sensors & Controls Lab
SARGE - Encoder Board Rev. 0
March 29, 2005



5/02/2005 11:35:39a C:\Program Files\Eagle\Projects\Auto_Sprayer_Encoder.brd

Cost Analysis

Appendix 9

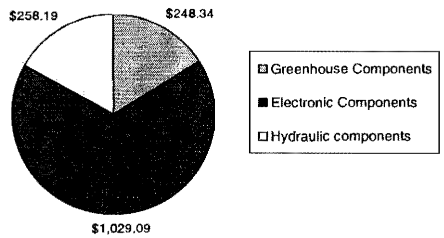
Account: E01-1115-001-1001901

Greenhouse Robot Sprayer - Development Costs

Description	Link	Manufacturer	Manfacturer Part #	Distributor	Distributor Part #	Quantity	Price	Our Cost	Total Cost
PVC Tubing & joints	http://www.lowes.com			Lowes		1	$111.90	$111.90	$111.90
Garden Hose	http://www.lowes.com			Lowes		1	$32.94	$32.94	$32.94
Wire	http://www.stokeselectric.com			Stokes Electrical Supply		1	$103.50	$103.50	$103.50
Hydraulic components				Blount GB Farm COOP			$19.47	$19.47	$19.47
Nozzles, Strainer	http://www.spray.com			Spraying Systems Company			$51.10	$51.10	$51.10
Ball Valves, hose barb fittings				IPW			$81.51	$81.51	$81.51
Hose Clamps	http://www.lowes.com			Lowes		1	$8.80	$8.80	$8.80
Pressure Gauge	http://www.lowes.com			Lowes		1	$7.61	$7.61	$7.61
Pump		Shurflow		Shurflow		1	$89.70	$89.70	$89.70
LCD Display		Seetron		Scott Edwards Electric		1	$68.00	$68.00	$68.00
NEMA Box	http://www.graybar.com			Graybar Electric		1	$25.01	$25.01	$25.01
Board of Education	http://www.parallax.com/detail.asp?product_id=BS2SX-IC	Parallax	28150	Parallax		2	$65.00	$0.00	$130.00
BASIC Stamp 2pe	http://www.parallax.com/detail.asp?product_id=BS2SX-IC	Parallax		Parallax		1	$75.00	$0.00	$75.00
BASIC Stamp 2sx	http://www.parallax.com/detail.asp?product_id=BS2SX-IC	Parallax	BS2SX-IC	Parallax	BS2SX-IC	1	$59.00	$0.00	$59.00
Maxi Dual H-Bridge	http://www.robotstore.com/catalog/display.asp?pid=264	Mondo-Tronics	3-773	RobotStore.com	3-773	1	$59.00	$0.00	$59.00
Magnetic Proximity Sensor	http://www.allegromicro.com	Allegro	ATS665LSG	Allegro		1	$4.30	$0.00	$4.30
PWMPAL	http://www.parallax.com/detail.asp?product_id=28020	Parallax	28020	Parallax	28020	1	$34.95	$34.95	$34.95
Electronic Components	http://www.digikey.com	Digi-Key		Digi-Key			$632.56	$632.56	$632.56
DC DC Converter	http://www.astrodyne.com			Astrodyne, Inc		1	$58.96	$58.96	$58.96
Battery		Werker		Batteries Plus		1	$70.00	$70.00	$70.00
Lab Supplies				Digi-Key / Newark Inone			$139.61	$139.61	$139.61
							Total:	$1,535.62	$1,862.92

Greenhouse Components	$248.34
Electronic Components	$1,029.09
Hydraulic components	$258.19

Development Costs



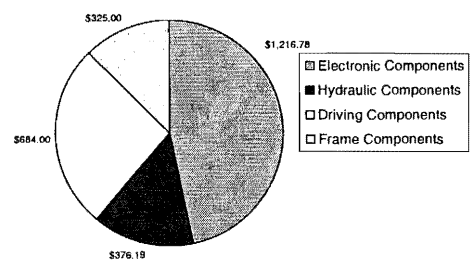
Greenhouse Robot Sprayer - Prototype

Description	Quantity	Price	Total Cost
Hydraulic components		$19.47	$19.47
Nozzles, Strainer		$51.10	$51.10
Ball Valves, hose barb fittings		$81.51	$81.51
Hose Clamps	1	$8.80	$8.80
Pressure Gauge	1	$7.61	$7.61
Pump	1	$89.70	$89.70
14 gallon tank	1	$30.00	$30.00
Solenoid valves	2	$44.00	$88.00
LCD Display	1	$68.00	$68.00
NEMA Box	1	$25.01	$25.01
Board of Education	2	$65.00	$130.00
BASIC Stamp 2pe	1	$75.00	$75.00
BASIC Stamp 2sx	1	$59.00	$59.00
Maxi Dual H-Bridge	1	$59.00	$59.00
Magnetic Proximity Sensor	1	$4.30	$4.30
PWMPAL	1	$34.95	$34.95
Electronic Components		$632.56	$632.56
DC DC Converter	1	$58.96	$58.96
Battery	1	$70.00	$70.00
Frame/Body/Boom		$75.00	$75.00
Frame welding/component assembly		$250.00	$250.00
Pillow-block bearing	4	$19.00	$76.00
Timing belt pulley	4	$15.00	$60.00
Timing belt	2	$9.00	$18.00
12V wheel motors	2	$235.00	$470.00
Drive wheel/tire	2	$20.00	$40.00
Caster wheel	1	$20.00	$20.00
		Total:	$2,601.97

Electronic Components	$1,216.78
Hydraulic Components	$376.19
Driving Components	$684.00
Frame Components	$325.00

$30.00

Prototype Costs



Power Requirements

Appendix 10

W_H2O (N)	W_frame (N)	V_tank (m^3)	density_H2O (kg/m^3)	total_distance (ft)	time per GH (sec)	g (m/s^2)
557	513	0.0416	1000	640	711	9.81

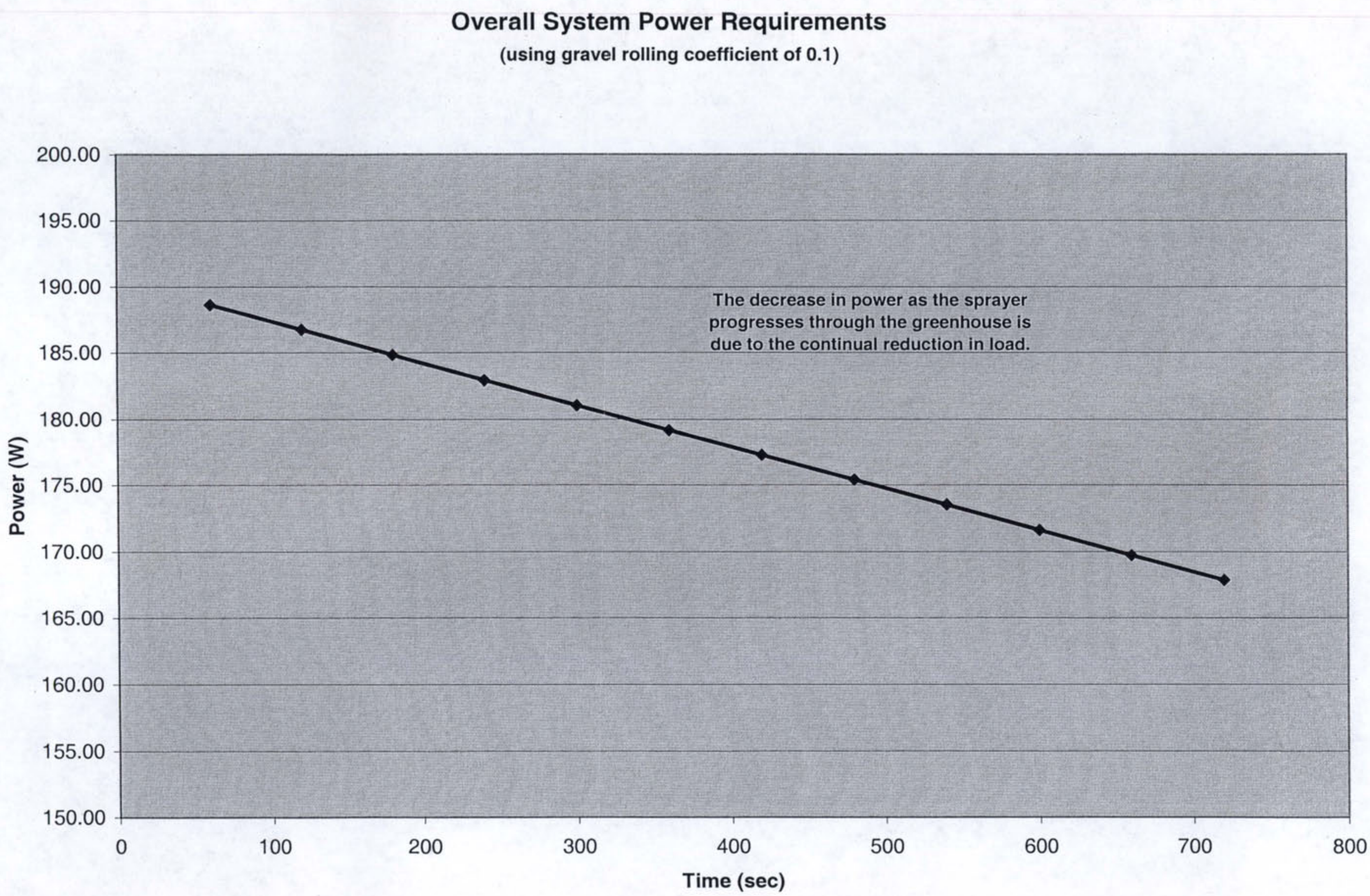
Electronics (W)	P_pump+P_valves (W)	velocity (ft/s)	velocity (m/s)
7	124.8	0.9	0.27432

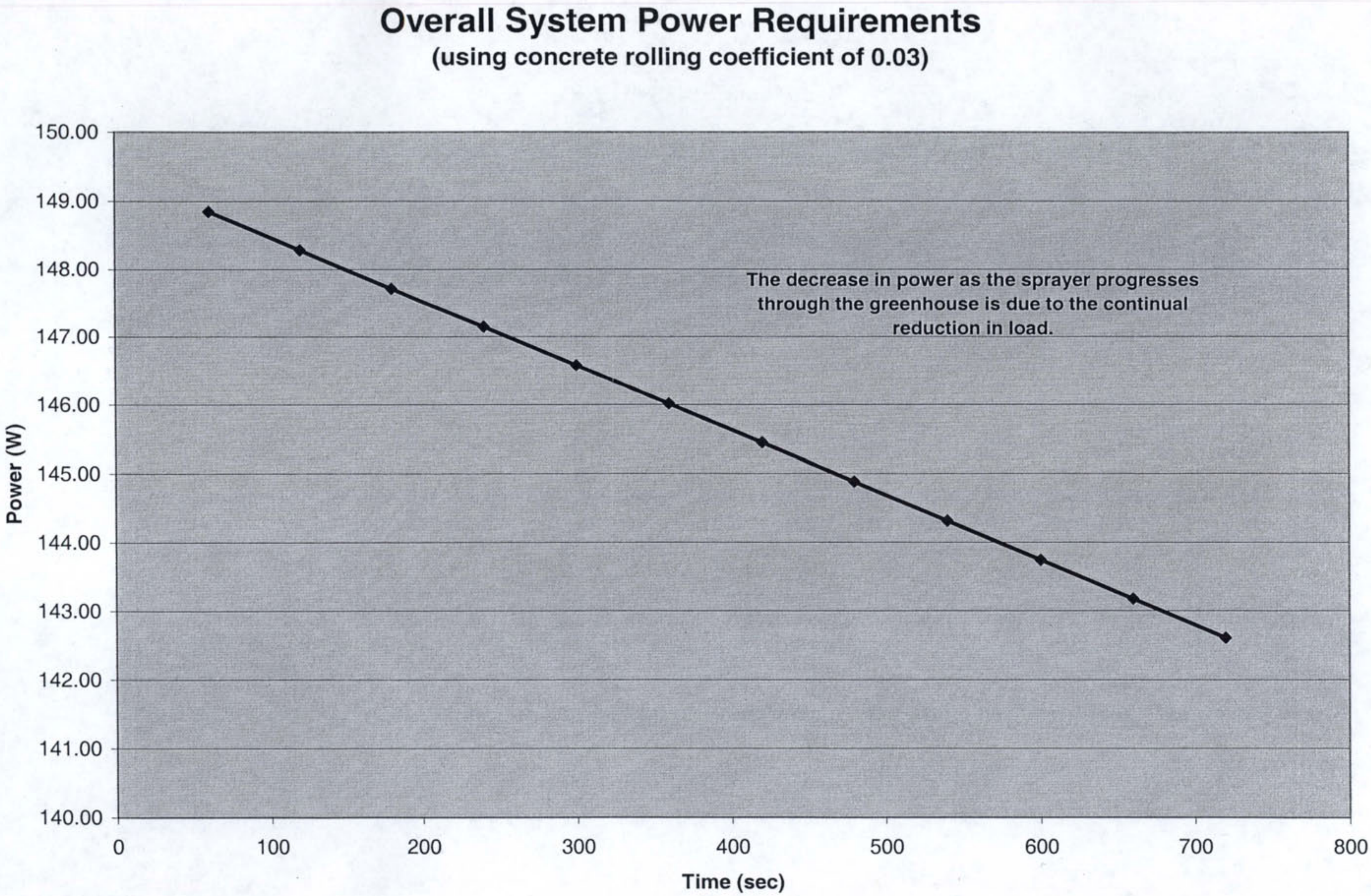
mass flow of H2O (kg/s)
0.0586

time (sec)	distance (m)	Power_battery (W) (gravel rolling coef. = .1)	Power_battery (W) (concrete rolling coef. = .03)	Power_battery (W) (with static coef. = .8)
0	0	0.00	0.00	0.00
60	16	188.63	148.85	586.43
120	33	186.74	148.28	571.30
180	49	184.85	147.71	556.17
240	66	182.96	147.15	541.04
300	82	181.06	146.58	525.92
360	99	179.17	146.01	510.79
420	115	177.28	145.44	495.66
480	132	175.39	144.88	480.54
540	148	173.50	144.31	465.41
600	165	171.61	143.74	450.28
660	181	169.72	143.18	435.15
720	198	167.83	142.61	420.03
	linear equation =	y = -0.0315x + 190.52	y = -0.0095x + 149.42	y = -0.2521x + 601.55
	total power in W (integrated) =	129009.6	105120	367771.68

To minimize coefficient of rolling friction:

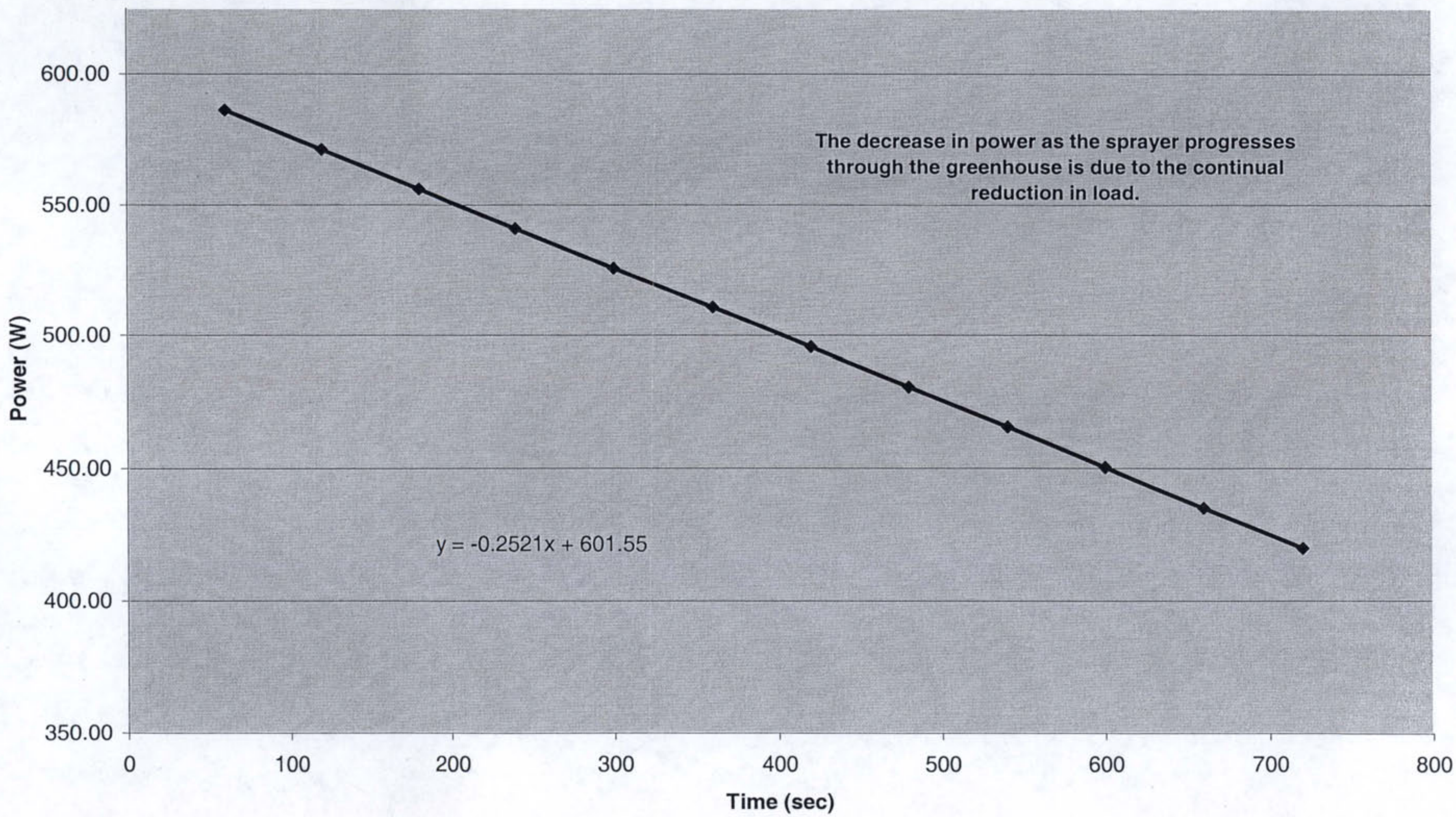
- Drive on harder and smoother surfaces
- Larger diameter wheels — *with larger wheels there will be a larger rotational kinetic energy requirement*
- Higher tire pressures
- Smoother and thinner tire treads
- Narrower tires and tread patterns


Overall System Power Requirements
(using gravel rolling coefficient of 0.1)
200.00
195.00
190.00
185.00
180.00
175.00
170.00
165.00
160.00
155.00
150.00
Power (W)
0
100
200
300
400
500
600
700
800
Time (sec)
The decrease in power as the sprayer progresses through the greenhouse is due to the continual reduction in load.


Overall System Power Requirements
(using concrete rolling coefficient of 0.03)
150.00
149.00
148.00
147.00
146.00
145.00
144.00
143.00
142.00
141.00
140.00
Power (W)
0
100
200
300
400
500
600
700
800
Time (sec)
The decrease in power as the sprayer progresses through the greenhouse is due to the continual reduction in load.

Overall System Power Requirements

(using static coefficient of 0.8)



Battery Specs

Appendix 11

Werker.

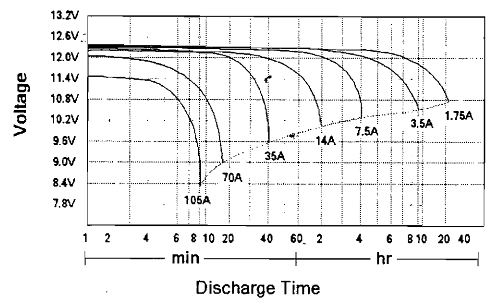
Sealed Lead Acid Absorbed Glass Mat

Technical Specifications

WKA12-33C

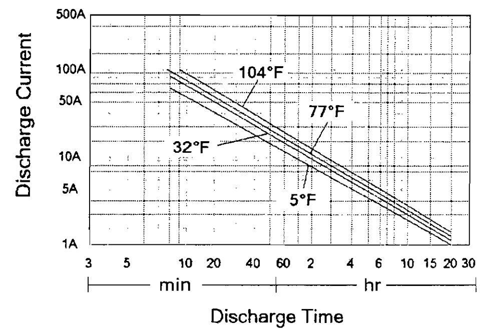
Discharge Characteristics

Duration of Discharge @ 77°F



Discharge Duration vs. Discharge Current

Duration of Discharge @ Various Temperatures



Specifications

All Specifications Are Rated at 77°F Unless Otherwise Noted

Nominal Voltage			12V
Ampere Hour Capacity (20hr Rate to 1.75VPC)			35Ah
Dimensions		inches	millimeters
	Length	7.72"	196mm
	Width	5.16"	131mm
	Height	6.10"	155mm
	Height w/Term.	6.57"	167mm
Weight		25.35lbs	
Case Plastic		ABS Resin	
Maximum Charge Current		0.3C or 10.5A	
Recommended Charging	Float Use Voltage		2.28V/Cell
	Float Use Current		<5A
	Cycle Use Voltage		2.45V/Cell
	Cycle Use Current		<10.5A
Shelf Life		1 Month	97%
		2 Months	91%
		3 Months	83%
Temperature Range	Charge	32°F to 104°F	
	Discharge	5°F to 122°F	
	Storage	5°F to 104°F	
Capacity Affected by Temperature (20hr rate)		77°F	100%
		32°F	85%
		5°F	64%

Capacity Ratings

Capacity @ 77°F	20 hour rate	35.0Ah
	10 hour rate	32.6Ah
	5 hour rate	29.8Ah
	1 hour rate	21.0Ah

Wattage Ratings

Discharge Rate	End Voltage	
	1.75V/Cell	1.67V/Cell
5 min. rate	253.0	265.0
10 min. rate	174.0	183.0
15 min. rate	137.0	145.0
30 min. rate	81.7	85.6
45 min. rate	60.9	63.3
60 min. rate	50.3	52.0

Ascent Battery Supply LLC
925 Walnut Ridge Drive
Hartland, WI 53029

Ascent

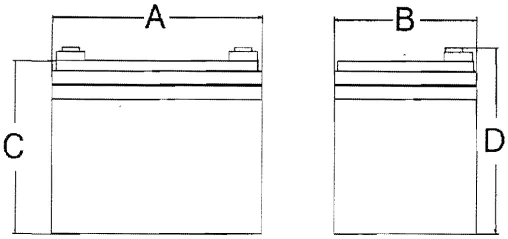
Werker.

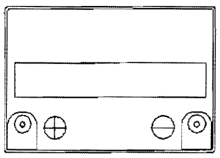
Sealed Lead Acid Absorbed Glass Mat

Technical Specifications

WKA12-33C

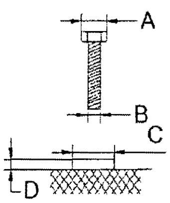
Physical Dimensions





	in	mm
A	7.64	196
B	5.11	131
C	6.05	155
D	7.02	180

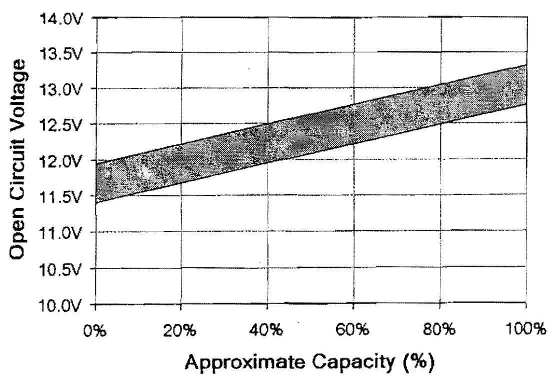
Terminal



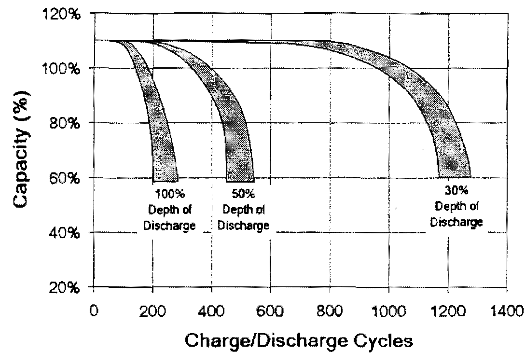
	in	mm
A	0.390	10
B	0.195	5
C	0.624	16
D	0.156	4

Open Circuit Voltage vs. Capacity

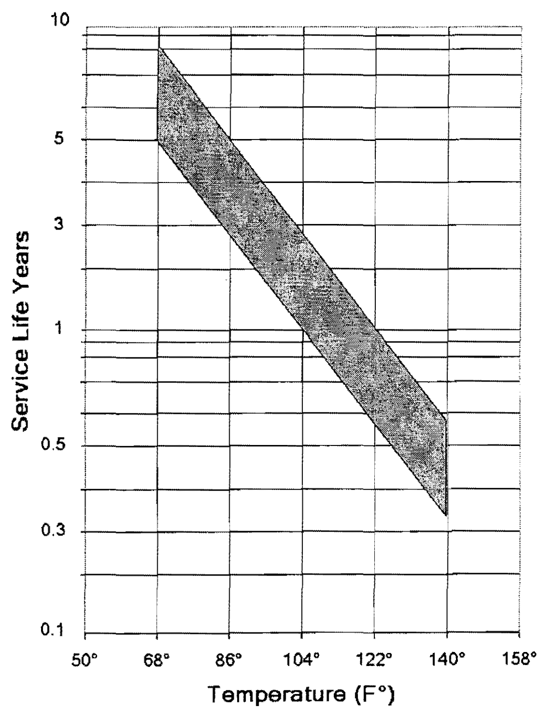
Estimated Residual Capacity @ 77°F



Cycle Life vs. Depth of Discharge



Effect of Temperature on Float Life



ISO 9001

MH 20720

The information contained on this specification is generally descriptive only and is not intended to make or imply any representation guarantee or warranty with respect to any cells and batteries. Cell and battery design/specification are subject to modification without notice.

Ascent Battery Supply LLC
925 Walnut Ridge Drive
Hartland, WI 53029

Ascent

Fluid Spreadsheet

Appendix 12

Nozzle Size 0.06 gpm @ 60 psi

Time min.	Int. Flow Meter Reading	Final Flow Meter Reading	Flow Gallons	Flow Rate gpm	Pressure psi
12	127.2	135.4	8.2	0.68	60
12	135.4	144.2	8.8	0.73	60
12	144.5	153.2	8.7	0.72	60

Nozzle Size 0.08 @ 60 psi

Time min.	Int. Flow Meter Reading	Final Flow Meter Reading	Flow Gallons	Flow Rate gpm	Pressure psi
10	153.2	163.2	10	1	60
10	163.2	173.3	10.1	1.01	60
10	173.3	183.1	9.8	0.98	60

# of Nozzles	12	Greenhouse Length	96 ft
Row Spacing	60 in.	Greenhouse Width	30 ft

Ground Speed	Flow Rate GPM @ 60 psi	Application Rate GPA	Application Rate Gallons/Greenhouse
1 ft/sec	0.04	69.7	4.61
0.68 MPH	0.05	87.1	5.76
	0.06	104.5	6.91
	0.07	122.0	8.06
	0.08	139.4	9.22
	0.09	156.8	10.37
	0.1	174.2	11.52
	0.11	191.7	12.67
	0.12	209.1	13.82
	0.13	226.5	14.98

Pump Specs

Appendix 13

SHURflo®

First in Fluid Innovation™

PRODUCT DATA SHEET

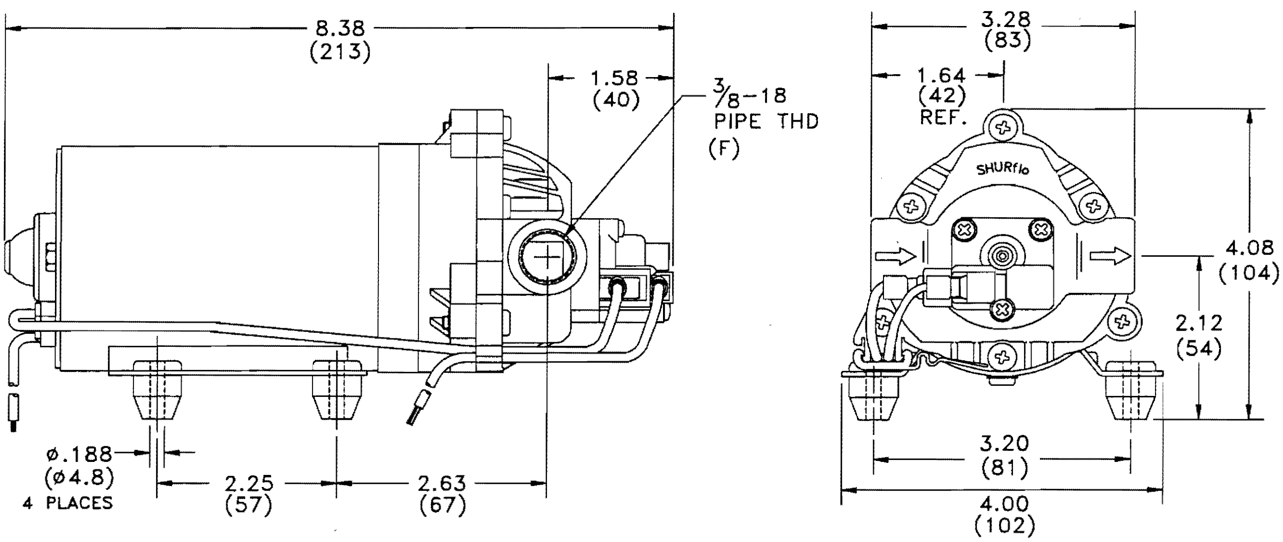
MODEL: 8000-543-238

SPECIFICATIONS:

MODEL NUMBER: 8000-543-238
PUMP DESIGN: Positive Displacement 3 Chamber Diaphragm Pump
CHECK VALVE: (1-Way Operation) Prevents Reverse Flow
CAM: 3.0 Degree
MOTOR: Permanent Magnet, P/N 11-111-00
VOLTAGE: 12 VDC Nominal
PRESSURE SWITCH: Adjustable Shut-Off (Range 80-100 PSI)
Factory Set @100 PSI, Turn On 85 PSI ±5 PSI
LIQUID TEMPERATURE: 170 Degrees Fahrenheit (77 Degrees Centigrade) Max.
PRIME: Self-Priming Up To 10 Ft. Vertical,
Max. Inlet Pressure 30 PSI (2.1 Bar)
PORTS: 3/8"-18 NPT Female
MATERIAL OF CONSTRUCTION:
- **PLASTICS-** Polypropylene
- **VALVES-** Viton
- **DIAPHRAGM-** Santoprene
- **FASTENERS-** Zinc Plated Steel

NET WEIGHT: 4.7 Lbs (2.13 Kg)
DUTY CYCLE: Intermittent (See Temperature Rise Chart)
TYPICAL APPLICATIONS: Agricultural Spraying

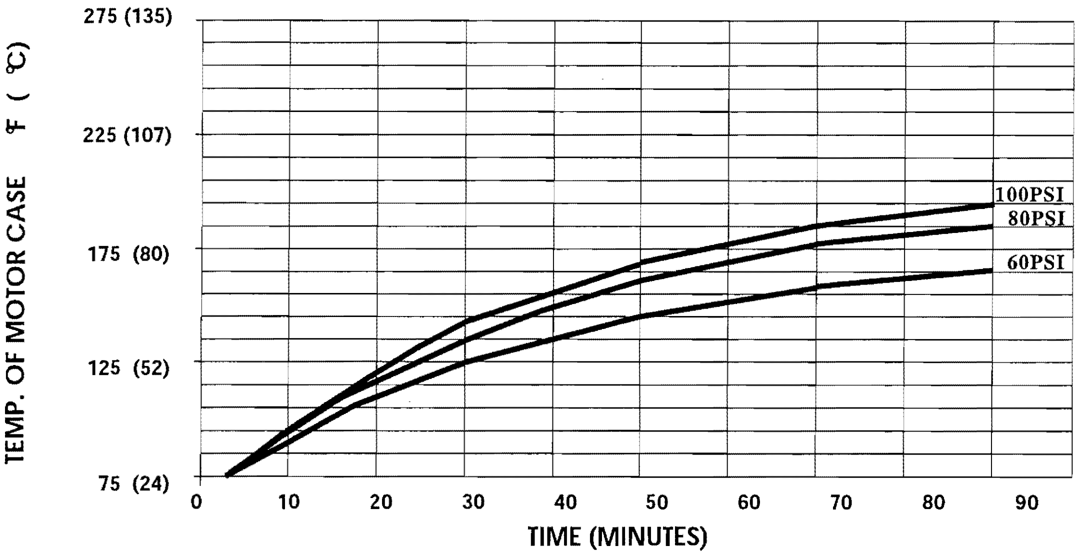
DIMENSIONS:



FILE#8000-543-238 REVISED: 11/02 ISSUED: 9/17/93

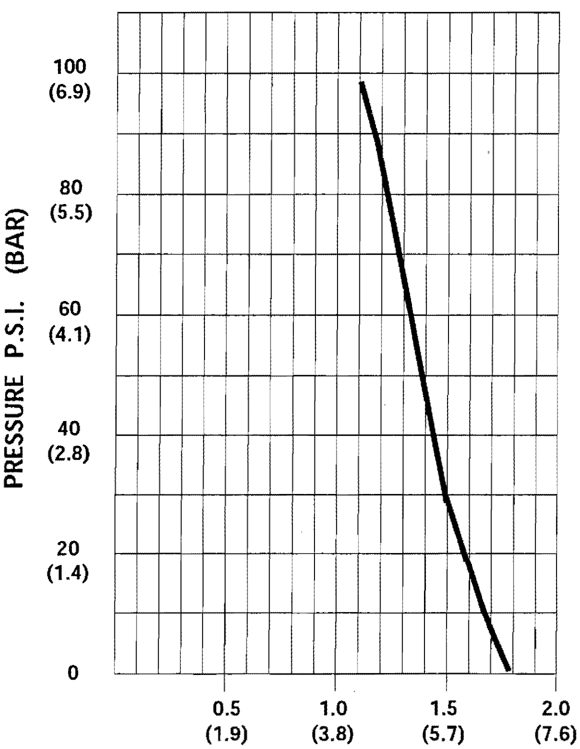
MODEL: 8000-543-238

TEMPERATURE RISE



THIS GRAPH IS FOR USE AS A DESIGN GUIDE. IT IS BASED ON RUNNING CONTINUOUSLY WITH AN AMBIENT TEMPERATURE OF 70° F IN STILL AIR.

TYPICAL PERFORMANCE



FLOW - GALLONS PER MINUTE (LITERS PER MINUTE)

PRESSURE (PSI)	FLOW (GPM/LIT)	RPM MIN/MAX	CURRENT (AMPS)	VOLTAGE (VOLTS)
OPEN	1.8/6.8	2206/2660	3.0	12VDC
10	1.52/5.8	2170/2635	3.6	"
20	1.45/5.5	2130/2590	4.2	"
30	1.36/5.1	2090/2540	4.9	"
40	1.31/5.0	2060/2495	5.5	"
50	1.26/4.8	2030/2440	6.0	"
60	1.22/4.6	2005/2380	6.5	"
70	1.16/4.4	1980/2350	7.1	"
80	1.11/4.2	1940/2330	7.6	"
90	1.03/3.9	1915/2260	8.3	"
100	0.96/3.6	1900/2225	8.7	"

-SPECIFICATIONS SUBJECT TO CHANGE WITHOUT NOTICE.
-ALL DATA BASED ON TESTING WITH WATER.

5900 KATELLA AVENUE, CYPRESS CA. 90630 (562) 795-5200 (800) 854-3218 FAX (562) 795-7554
SHURflo EAST, 52748 PARK SIX COURT, ELKHART, IN 46514 (574) 262-0478/(800) 762-8094/FAX (574) 262-0478
SHURFLO LIMITED, UNIT 5, STERLING PARK, GATWICK ROAD, WEST SUSSEX, RH10 2QT, UK 44(0) 1293-424-000 FAX 44 (0) 1293-421-880

Motor Specs

Appendix 14

GROSCHOPP

ELECTRIC MOTORS • GEARMOTORS • MOTOR SETS

420 15th Street
Sioux Center, IA 51250-2100
Voice: 712-722-4135

gro@groschopp.com
www.groschopp.com
Fax: 712-722-1445

MOTORTEC℠

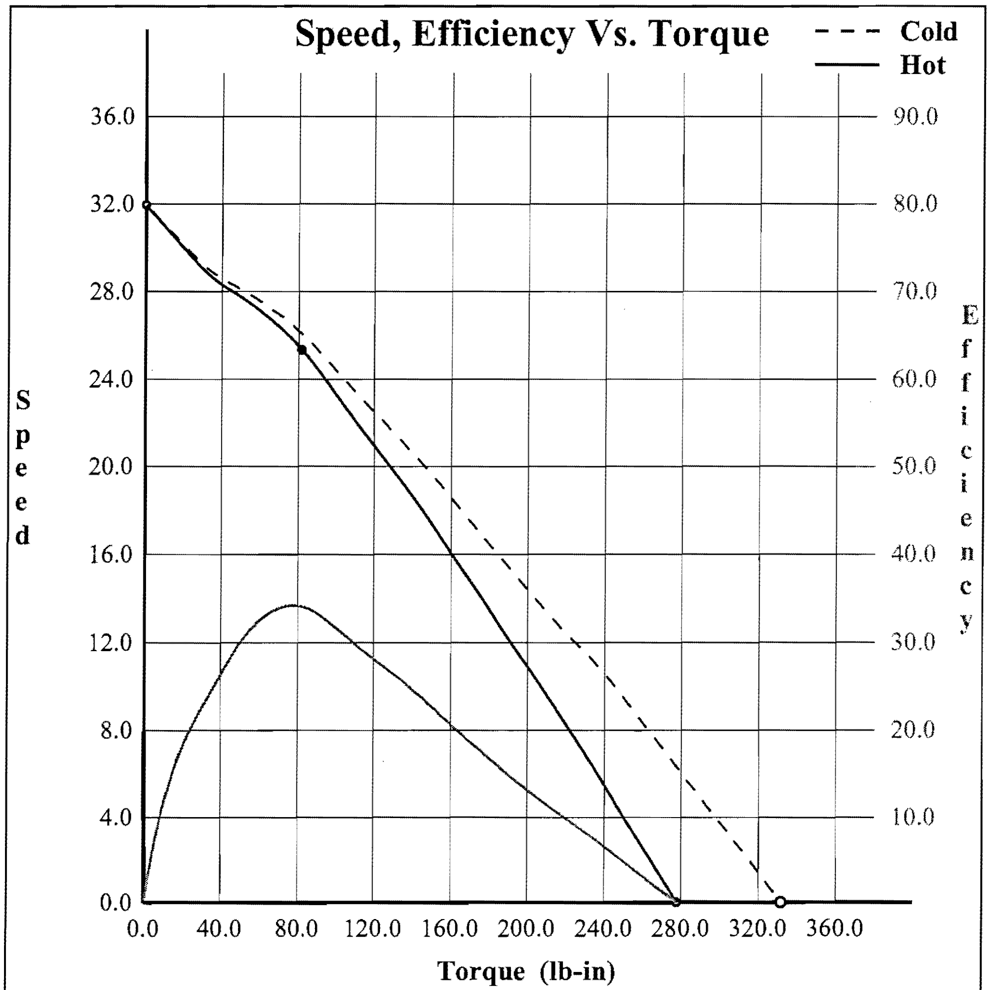
DATA SHEET

Jan 19, 2005

Model # PM8014-RA3040M

Customer: GG-Web 20207
Application:
Quote:

Motor: PM 8014 12 v DC
2 Poles 1050 rpm
Gearbox: RA 3000M 40:1 Ratio



Rating:
Speed: 25.4 rpm
Torque: 81.8 lb-in
Current: 6.4 amps
Output: 25 watts
Output: 0.0329 HP

Duty Cycle:
On: Continuous
Off:

Efficiency:
Gearbox: 52.4 %
Motor: 65.1 %
System: 34.1 %

Start/Stall Conditions:
Current: 24.01 amps
Torque: 331.84 lb-in

Notes:

Catalog No: 20207
PM8014-209-012
RA3000-040M

IMPORTANT:

Please note that there are not implied warranties that the goods shall be merchantable or that they are fit for a particular purpose.

Submitted By: Groschopp.com

712-722-4135
gro@groschopp.com

GROSCHOPP
ELECTRIC MOTORS • GEARMOTORS • MOTOR SETS

420 15th Street
Sioux Center, IA 51250-2100
Voice: 712-722-4135

gro@groschopp.com
www.groschopp.com
Fax: 712-722-1445

MOTORTEC℠

HP Rating 0.106

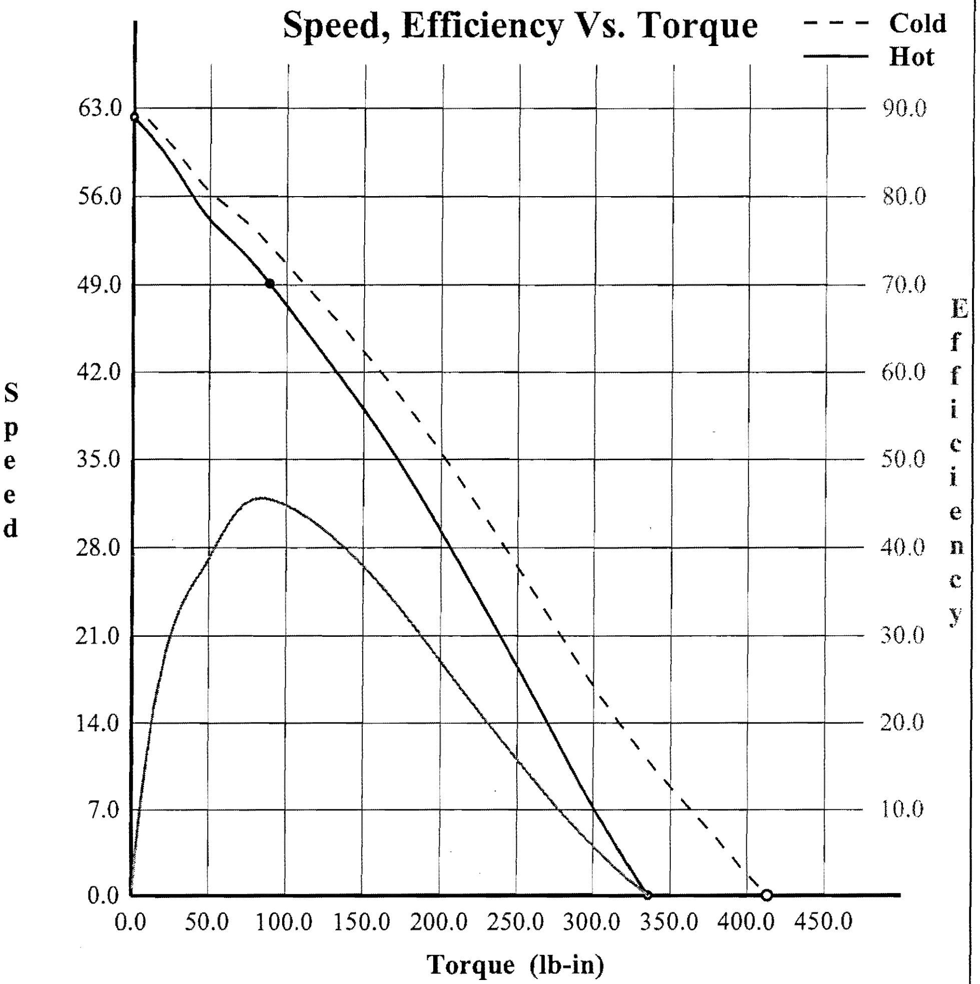
DATA SHEET

Jan 19, 2005

Model # PM8018-RA4020M

Customer: GG-Web 41304
Application:
Quote:

Motor: PM 8018 12 v DC
2 Poles 950 rpm
Gearbox: RA 4000M 20:1 Ratio



Rating:
Speed: 49.1 rpm
Torque: 88.7 lb-in
Current: 10.2 amps
Output: 52 watts
Output: 0.0691 HP

Duty Cycle:
On: Continuous
Off:

Efficiency:
Gearbox: 65.2 %
Motor: 69.7 %
System: 45.5 %

Start/Stall Conditions:
Current: 41.79 amps
Torque: 413.48 lb-in

$ 436.00

Gear Ratio
1: 2 1/2

Catalog No: 41304
PM8018-205-012
RA4000-020M

Notes:

IMPORTANT:
Please note that there are not implied warranties that the goods shall be merchantable or that they are fit for a particular purpose.

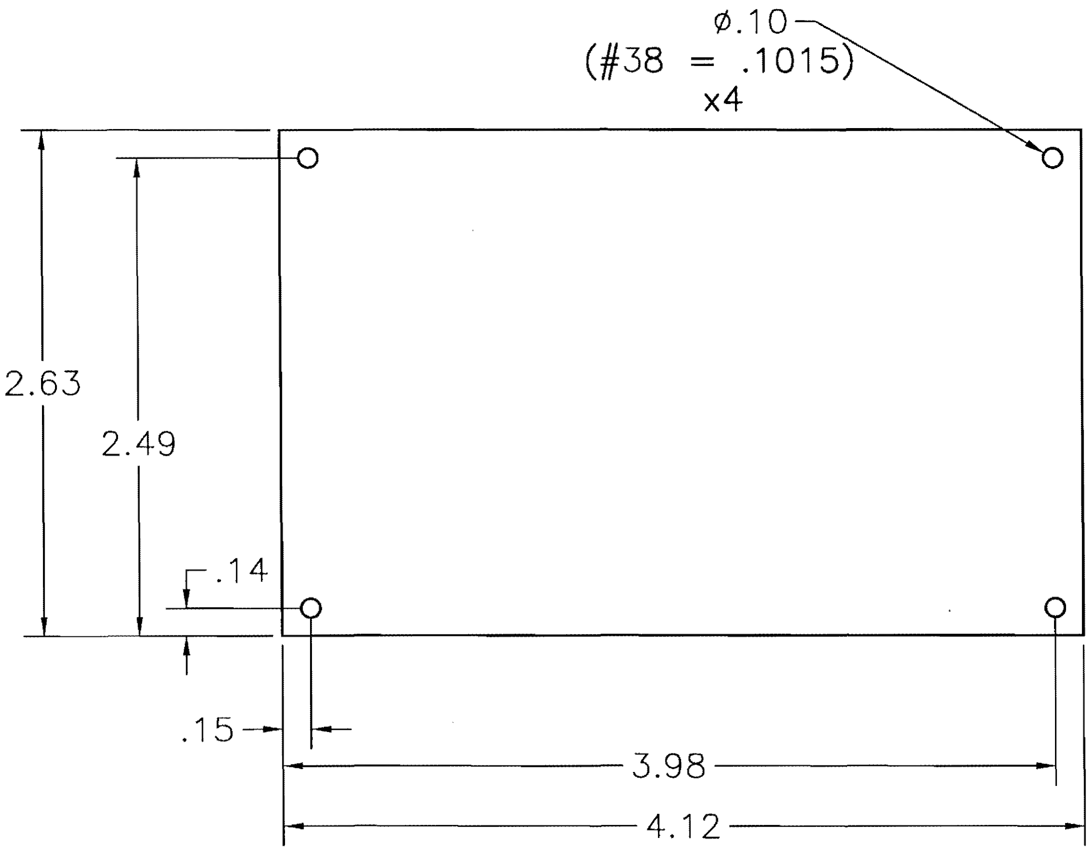
Submitted By: Groschopp.com

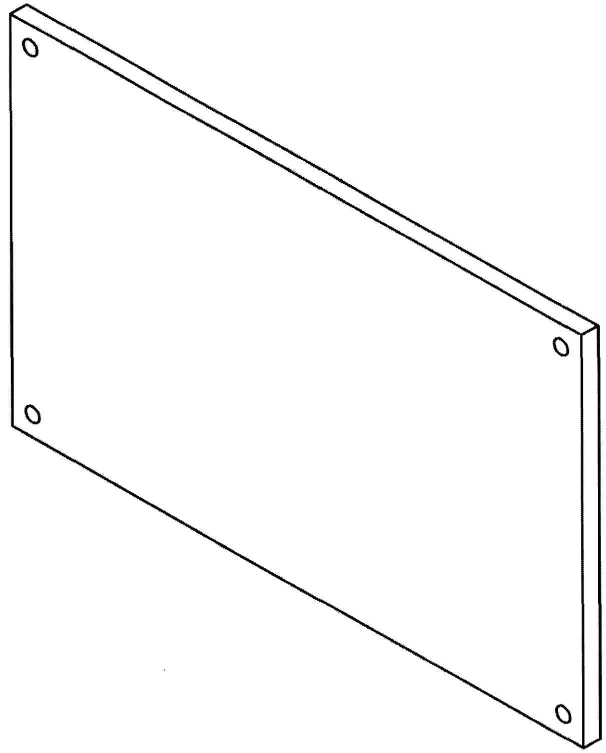
712-722-4135
gro@groschopp.com

Mechanical Desktop

Appendix 15

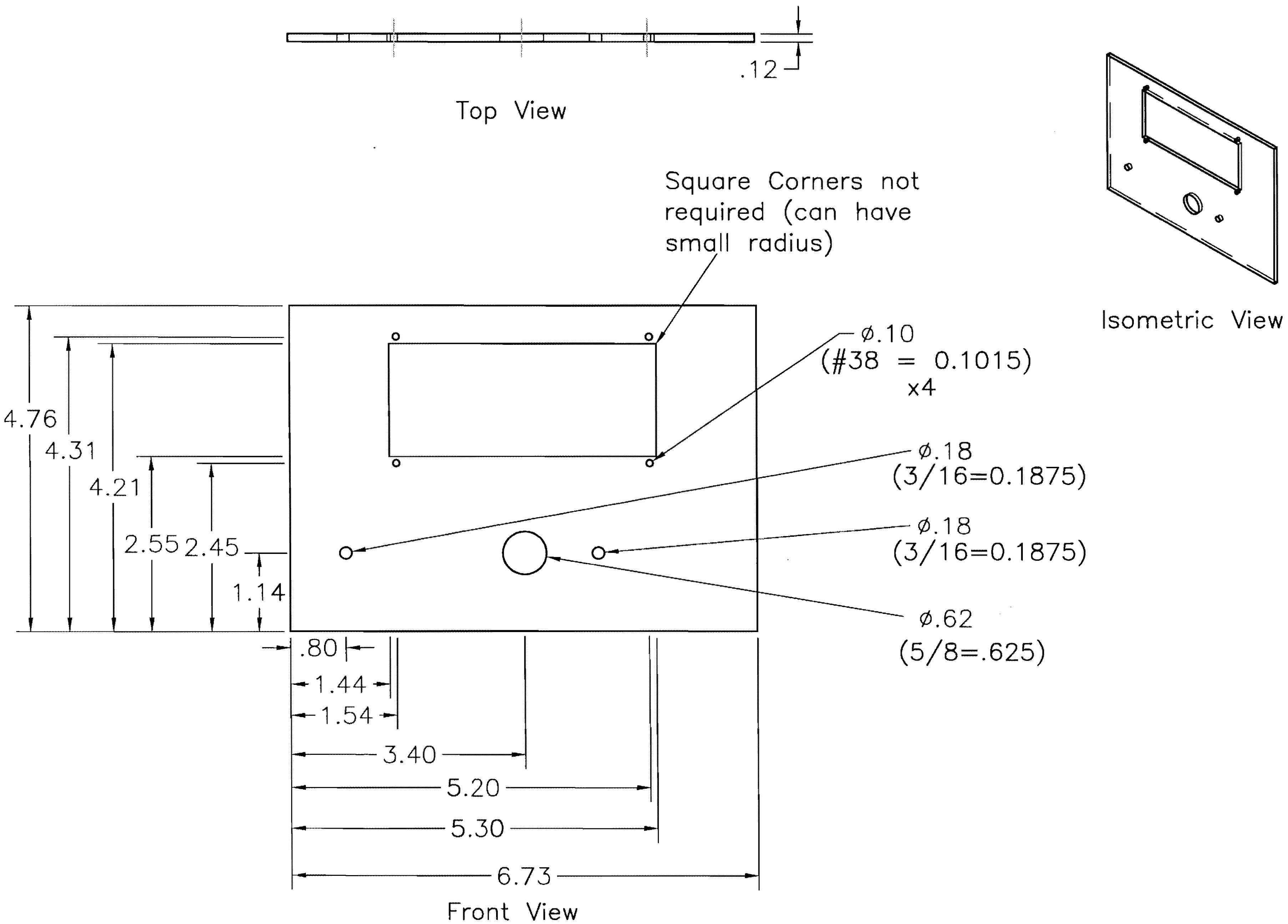
LCD Covering





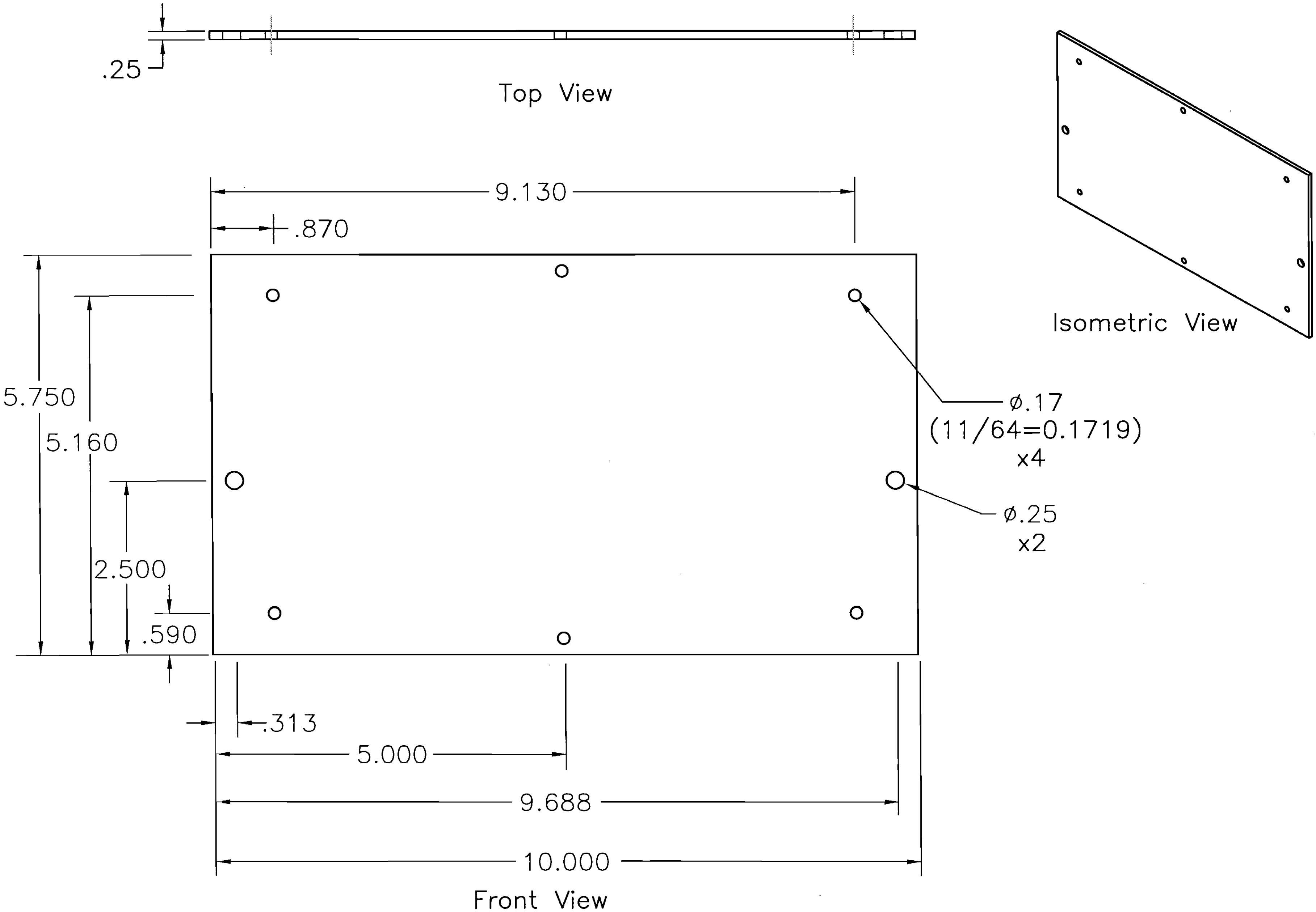
Name: SARGE Material: Lexan Units: Inches Scale: 1:1 Date: 3/8/05

LCD Casing



Name: SARGE   Material: Polycarbonate   Units: Inches   Scale: .5:1   Date: 3/8/05

NEMA 4 Box Mount



Name: SARGE Material: PVC Units: Inches Scale: .5:1 Date: 3/15/05

Tipping Moment

Appendix 16

	wheelbase	track width rear	Wt front	Wt left rear	Wt right rear	Wt
sprayer empty	0.47	0.5	236.332	191.741	196.200	624.273
sprayer (4 gal)	0.47	0.5	334.432	240.791	227.414	802.636
sprayer (6 gal)	0.47	0.5	372.334	236.332	249.709	858.375
sprayer (8 gal)	0.47	0.5	416.925	258.627	254.168	929.720
sprayer (10 gal)	0.47	0.5	459.286	267.545	276.464	1003.295
sprayer (12 gal)	0.47	0.5	508.336	276.464	294.300	1079.100
sprayer (14 gal)	0.47	0.5	557.386	298.759	294.300	1150.445

	Experimental
	° Lateral slope
sprayer empty	
sprayer (4 gal)	
sprayer (6 gal)	
sprayer (8 gal)	
sprayer (10 gal)	13.5
sprayer (12 gal)	15
sprayer (14 gal)	17

Will not drive on slopes greater than 5 degrees due to motors

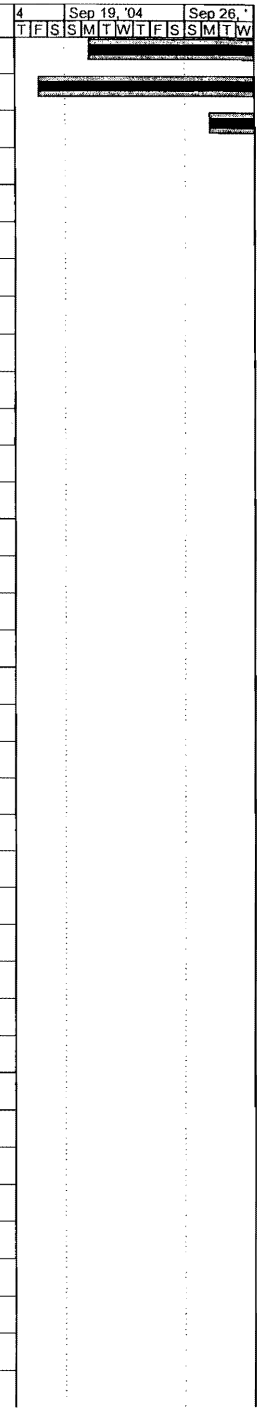
Wt' LR	**Wt'RR**	Wt'F	**n**	**Rear Radius**	**Front Rad**	Δ radius	L'	tanλ1
258.627	276.464	89.182	0.270	0.130	0.130	0.000	0.449	17.330
347.809	350.039	104.789	0.270	0.130	0.130	0.000	0.449	17.330
387.941	365.645	104.789	0.270	0.130	0.130	0.000	0.449	17.330
414.695	392.400	122.625	0.270	0.130	0.130	0.000	0.449	17.330
414.695	445.909	142.691	0.270	0.130	0.130	0.000	0.449	17.330
468.205	441.450	169.445	0.270	0.130	0.130	0.000	0.449	17.330
499.418	441.450	209.577	0.270	0.130	0.130	0.000	0.449	17.330

tanλ2	tanλ	track width	x	z	° long	° lat	Y	% Slope
0.000	17.330	0.500	0.178	0.355	26.618	35.151	0.250	50.11504
0.000	17.330	0.500	0.196	0.431	24.438	30.119	0.250	45.44229
0.000	17.330	0.500	0.204	0.469	23.473	28.036	0.250	43.42515
0.000	17.330	0.500	0.211	0.477	23.848	27.670	0.250	44.20525
0.000	17.330	0.500	0.215	0.475	24.355	27.744	0.250	45.26735
0.000	17.330	0.500	0.221	0.473	25.082	27.857	0.250	46.80556
0.000	17.330	0.500	0.228	0.455	26.568	28.767	0.250	50.00594

Microsoft Project

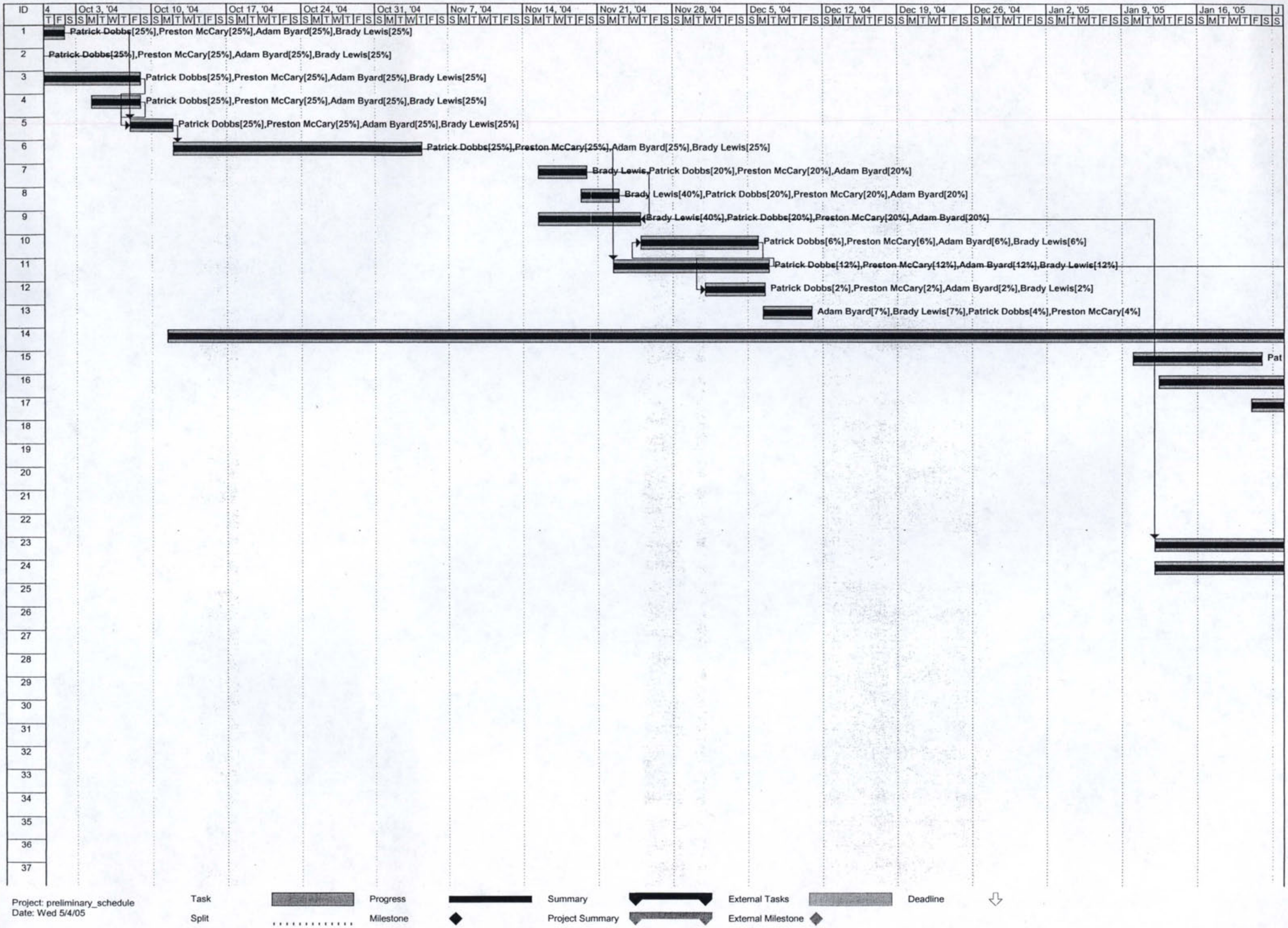
Appendix 17

ID		Task Name	Duration	Start	Finish	% Complete	Predecessors	Resource Names
1	✓	Proposal - Literature Review	13.65 days?	Mon 9/20/04	Fri 10/1/04	100%		Patrick Dobbs[25%],Preston McCary[25%],Adam Byard[25%],Brady Lewis[25%]
2	✓	Proposal - Problem/Needs Statement	14.6 days?	Fri 9/17/04	Wed 9/29/04	100%		Patrick Dobbs[25%],Preston McCary[25%],Adam Byard[25%],Brady Lewis[25%]
3	✓	Proposal - Engineering Analysis	13.65 days?	Mon 9/27/04	Fri 10/8/04	100%		Patrick Dobbs[25%],Preston McCary[25%],Adam Byard[25%],Brady Lewis[25%]
4	✓	Proposal - Health/Safety & Environmental Impact	5.5 days?	Mon 10/4/04	Fri 10/8/04	100%		Patrick Dobbs[25%],Preston McCary[25%],Adam Byard[25%],Brady Lewis[25%]
5	✓	Proposal 1st Draft	4.05 days?	Fri 10/8/04	Mon 10/11/04	100%	2,3,4,1	Patrick Dobbs[25%],Preston McCary[25%],Adam Byard[25%],Brady Lewis[25%]
6	✓	Proposal 2nd Draft	23 days?	Mon 10/11/04	Thu 11/4/04	100%	5	Patrick Dobbs[25%],Preston McCary[25%],Adam Byard[25%],Brady Lewis[25%]
7	✓	Microcontroller - Read from Inductor Circuit	5.5 days?	Mon 11/15/04	Fri 11/19/04	100%		Brady Lewis,Patrick Dobbs[20%],Preston McCary[20%],Adam Byard[20%]
8	✓	Microcontroller - PWM Signal	4.05 days?	Fri 11/19/04	Mon 11/22/04	100%		Brady Lewis[40%],Patrick Dobbs[20%],Preston McCary[20%],Adam Byard[20%]
9	✓	Microcontroller - Preliminary Guidance Controller	11.25 days?	Mon 11/15/04	Wed 11/24/04	100%	7FF,8FF	Brady Lewis[40%],Patrick Dobbs[20%],Preston McCary[20%],Adam Byard[20%]
10	✓	Preliminary Presentation	6.75 days?	Wed 11/24/04	Sun 12/5/04	100%	11	Patrick Dobbs[6%],Preston McCary[6%],Adam Byard[6%],Brady Lewis[6%]
11	✓	Proposal Final Draft	11.25 days?	Mon 11/22/04	Mon 12/6/04	100%	6	Patrick Dobbs[12%],Preston McCary[12%],Adam Byard[12%],Brady Lewis[12%]
12	✓	Faculty Presentation	6.25 days?	Wed 12/1/04	Mon 12/6/04	100%	10	Patrick Dobbs[2%],Preston McCary[2%],Adam Byard[2%],Brady Lewis[2%]
13	✓	Modify Existing Frame	5.5 days?	Mon 12/6/04	Fri 12/10/04	100%		Adam Byard[7%],Brady Lewis[7%],Patrick Dobbs[4%],Preston McCary[4%]
14	✓	Reevaluate Current Design	124.04 days?	Mon 10/11/04	Thu 3/10/05	100%		Adam Byard[40%],Patrick Dobbs[20%],Preston McCary[20%],Brady Lewis[20%]
15	✓	Correct Design Proposal	7.9 days?	Mon 1/10/05	Fri 1/21/05	100%		Patrick Dobbs[40%],Adam Byard[20%],Preston McCary[20%],Brady Lewis[20%]
16	✓	Microcontroller - Speed Sensors & Cruise Control	23 days?	Wed 1/12/05	Thu 2/3/05	100%		Brady Lewis[40%],Patrick Dobbs[20%],Preston McCary[20%],Adam Byard[20%]
17	✓	Microcontroller - Filter Inductor Circuit	15.8 days?	Fri 1/21/05	Thu 2/3/05	100%		Brady Lewis[40%],Patrick Dobbs[20%],Preston McCary[20%],Adam Byard[20%]
18	✓	Microcontroller - Battery Measuring	7.65 days?	Fri 1/28/05	Thu 2/3/05	100%		Brady Lewis[40%],Patrick Dobbs[20%],Preston McCary[20%],Adam Byard[20%]
19	✓	Microcontroller - Spray Valve Control	3.1 days?	Wed 2/2/05	Fri 2/4/05	100%		Brady Lewis[40%],Patrick Dobbs[20%],Preston McCary[20%],Adam Byard[20%]
20	✓	Microcontroller - Remote Status Indicator	1.9 days?	Thu 2/3/05	Fri 2/4/05	100%		Brady Lewis[40%],Patrick Dobbs[20%],Preston McCary[20%],Adam Byard[20%]
21	✓	Microcontroller - User Interface (LCD)	85.9 days?	Mon 2/14/05	Sat 8/6/05	100%		Brady Lewis[40%],Patrick Dobbs[20%],Preston McCary[20%],Adam Byard[20%]
22	✓	Microcontroller - Joystick Control	43.15 days?	Thu 2/10/05	Tue 3/22/05	100%		Brady Lewis[40%],Patrick Dobbs[20%],Preston McCary[20%],Adam Byard[20%]
23	✓	Microcontroller - Finalize Guidance Controller	109.4 days?	Wed 1/12/05	Fri 4/29/05	100%	16FF,9	Brady Lewis[40%],Patrick Dobbs[20%],Preston McCary[20%],Adam Byard[20%]
24	✓	Microcontroller - Finalize Master Controller	109.4 days?	Wed 1/12/05	Fri 4/29/05	100%	18FF,19FF,17FF,21FF,22FF	Brady Lewis[40%],Patrick Dobbs[20%],Preston McCary[20%],Adam Byard[20%]
25	✓	Microcontroller - Convert Breadboards to Circuit Boards	42.2 days?	Mon 2/14/05	Thu 3/31/05	100%	23,24	Brady Lewis[40%],Patrick Dobbs[20%],Preston McCary[20%],Adam Byard[20%]
26	✓	Construction - Build Prototype	18.45 days?	Mon 3/7/05	Thu 3/31/05	100%	25FF,29FF	Patrick Dobbs[25%],Preston McCary[25%],Adam Byard[25%],Brady Lewis[25%]
27	✓	Testing - Lay Greenhouse Wire	33.3 days?	Thu 1/27/05	Thu 2/24/05	100%		Patrick Dobbs[30%],Adam Byard[30%],Preston McCary[20%],Brady Lewis[20%]
28	✓	Testing - Energy Balance	17 days?	Mon 2/14/05	Mon 2/28/05	100%		Patrick Dobbs[40%],Preston McCary[20%],Adam Byard[20%],Brady Lewis[20%]
29	✓	Construction - Select New Battery	18.2 days?	Mon 2/14/05	Tue 3/1/05	100%		Patrick Dobbs[40%],Preston McCary[20%],Adam Byard[20%],Brady Lewis[20%]
30	✓	Testing - Calculate New Center of Gravity	83.68 days?	Tue 2/1/05	Fri 4/22/05	100%		Preston McCary[40%],Patrick Dobbs[20%],Adam Byard[20%],Brady Lewis[20%]
31	✓	Testing - Flowrate Measurements	83.65 days?	Tue 2/1/05	Fri 4/22/05	100%		Preston McCary[40%],Patrick Dobbs[20%],Adam Byard[20%],Brady Lewis[20%]
32	✓	EURCA - Abstract Due	10.05 days?	Tue 3/1/05	Wed 3/9/05	100%		Patrick Dobbs[40%],Preston McCary[20%],Brady Lewis[20%],Adam Byard[20%]
33	✓	Testing - Cost Analysis	11.35 days?	Mon 4/18/05	Fri 4/29/05	100%		Patrick Dobbs[40%],Preston McCary[20%],Adam Byard[20%],Brady Lewis[20%]
34	✓	Testing & Modification	56.2 days?	Tue 3/1/05	Sun 5/1/05	100%	26FF,28FF,30FF,31FF	Patrick Dobbs[25%],Preston McCary[25%],Adam Byard[25%],Brady Lewis[25%]
35	✓	Presentation - EURCA	1.9 days?	Thu 3/31/05	Fri 4/1/05	100%	11,26FF	Patrick Dobbs[25%],Preston McCary[25%],Adam Byard[25%],Brady Lewis[25%]
36	✓	Final Report	10.98 days?	Mon 4/18/05	Tue 5/3/05	100%	11,26FF,34FF	Patrick Dobbs[25%],Preston McCary[25%],Adam Byard[25%],Brady Lewis[25%]
37		Presentation - ASAE Convention	4.05 days?	Sun 7/17/05	Wed 7/20/05	0%	36	Patrick Dobbs[25%],Preston McCary[25%],Adam Byard[25%],Brady Lewis[25%]

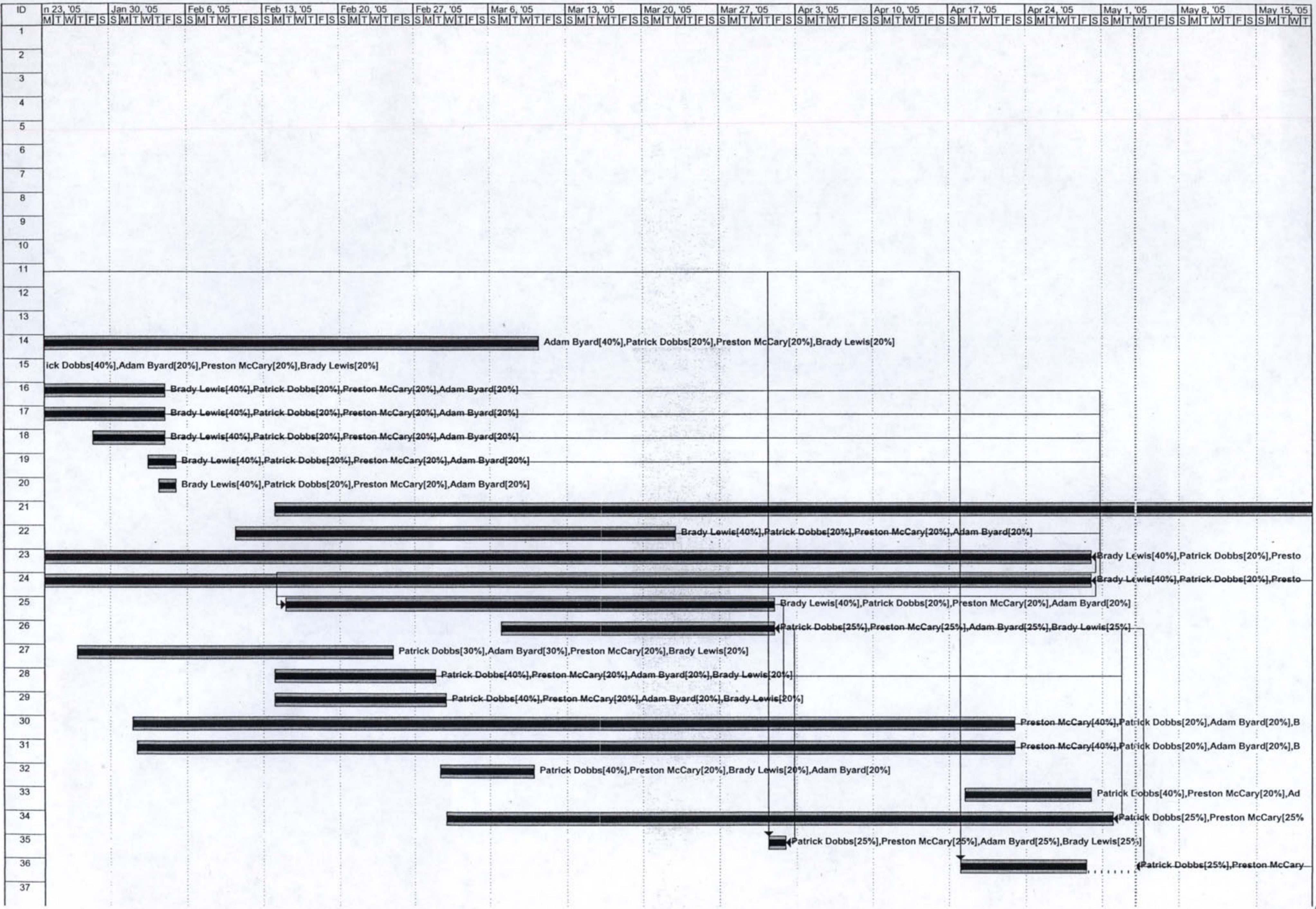


Project: preliminary_schedule
Date: Wed 5/4/05

Task | Progress | Summary | External Tasks | Deadline
Split | Milestone | Project Summary | External Milestone

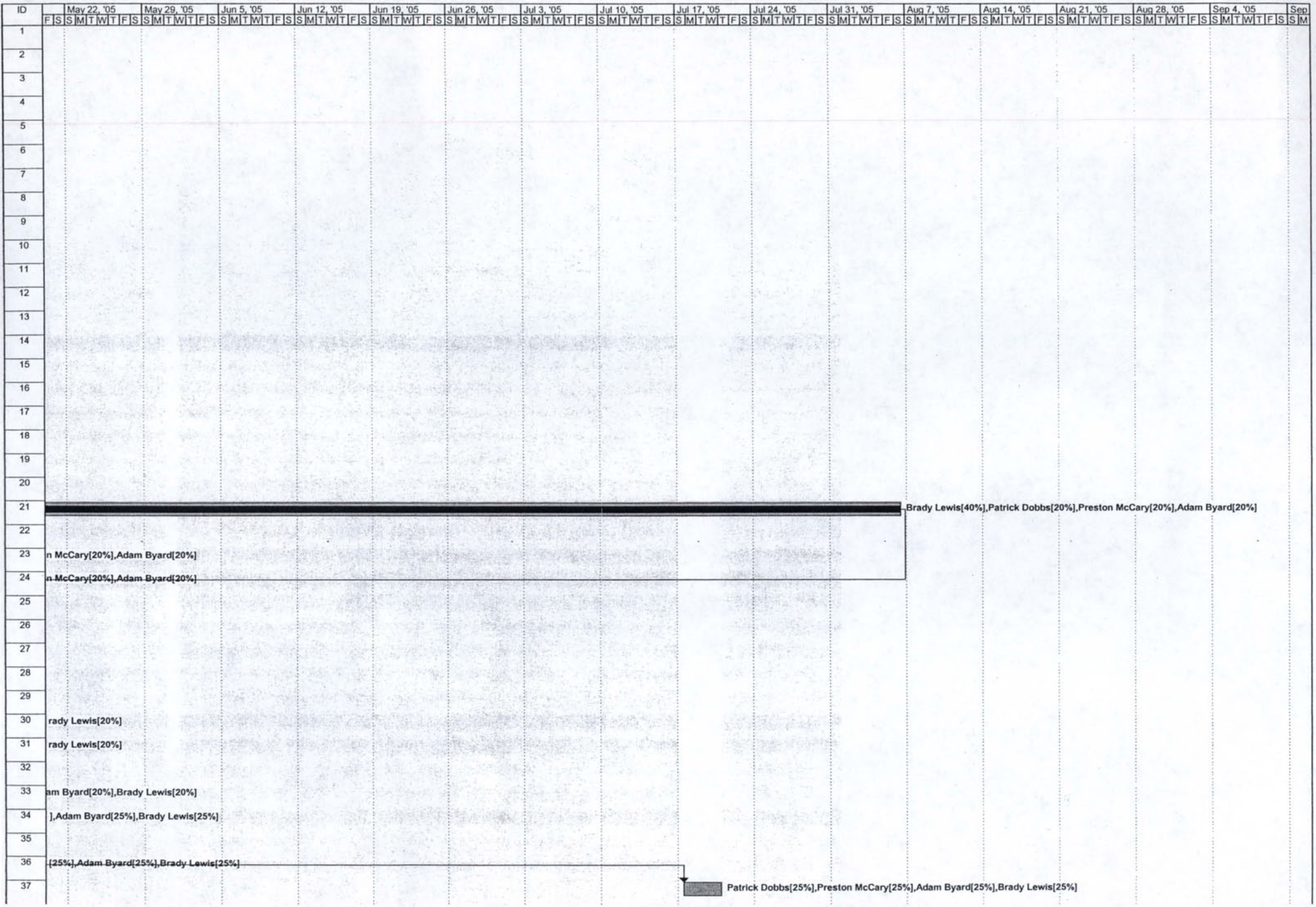


Project: preliminary_schedule
Date: Wed 5/4/05



Project: preliminary_schedule
Date: Wed 5/4/05

Task | Progress | Summary | External Tasks | Deadline
Split | Milestone | Project Summary | External Milestone



Project: preliminary_schedule
Date: Wed 5/4/05

Task | Progress | Summary | External Tasks | Deadline
Split | Milestone | Project Summary | External Milestone

Recommendations

Appendix 18

Recommendations:

Processor
The guidance system could operate more effectively using a faster processor with more memory. When developing a proportional control routine for steering the robot, voltage and distance data are needed for each new greenhouse. The new data is needed to account for wire type, depth of wire, and greenhouse floor material. This process could be automated with a new processor.

Safety
The team recommends that four light emitting diodes (LEDs) be mounted on top of the boom so that the producer could visually check the robot from outside the greenhouse. LEDs should be used instead of incandescent bulbs because they use less current.

A device needs to be placed on the front and rear doors so that if anyone enters the greenhouse it will shut of the power to the oscillator circuit; the robot will in turn stop spraying. This device will keep people from inadvertently being sprayed if they walk into the greenhouse while the robot is operating.

Motors
It was discovered by the current design team that the motors do not have enough power to drive with a full tank. The team suggests that 12 volt, Groschopp PM 8018 RA 4000M, 0.169 hp motors with a right angle gearbox with a 20:1 ratio be used. These motors have an output speed of 49.1 rpm and a torque of 88.7 in-lb. The pulleys connecting the motors to the axles need to be changed from a 1:1 ratio to a 2.5:1 ratio; this will reduce the speed of the tires and increase the torque.

Frame
The team also recommends that the front of the frame be altered so that the front tire can spin freely. This should be done by widening the front of the frame.