ECE 445

Fall 2012

Design Review

10/4/12

Robotic Microphone Stand

Team #24

Freddy Alejandro Gomez

Dennis Yuan

Prof. Andrew Carl Singer

TA: Igor Fedorov

Table of Contents

Introductions

Title: Robotic Microphone Stand

Mark Rubel from Pogo Studio, a local music producer, needs a robotic microphone stand which can be adjusted remotely. Mark's ideal stand has full translational motion in the x, y, and z directions, pan and tilt of a mounted microphone, and the ability to restore the microphone position to a previously stored configuration. It also includes video feedback and a sensor to prevent the stand from bumping into another object. All of this functionality is controlled wirelessly by somebody with a controller or by a laptop.

We decided to pursue this project, because we felt that Mark Rubel had a very real problem that we could tackle given the constraints of this class. Mark's reasons for desiring a robotic microphone stand were so compelling, in fact, that two other groups are attempting their own versions of this project. First, the audio memory of the average person is extremely short. This makes it hard to directly compare the two sounds, since by the time a producer walks into the booth, adjusts the microphone, and walks back out, his memory of what the previous sound was is no longer accurate. Second, anyone at home with a good microphone can record, produce, and distribute a song on the internet. The professionals have to evolve to attract new clients, and part of that evolution is being one step ahead in the recording environment. Also, the cool factor of a robotic microphone, gimmicky as that may seem, adds to the experience of using a professional producing service. We do not plan to implement all of Mark's requested features, but a selection that we through were central to the core of the idea.

Objectives

Intended Functions

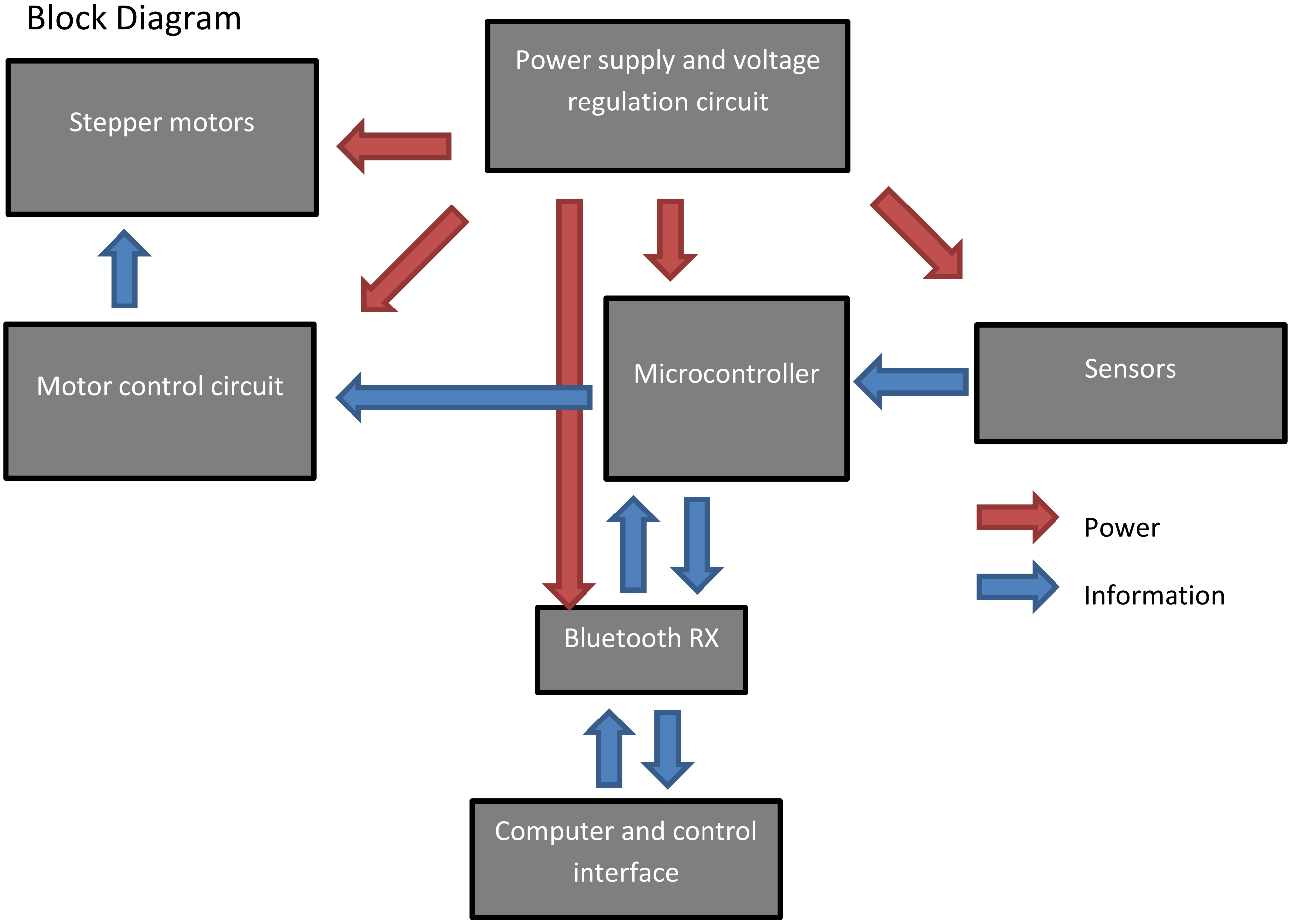
- Translational movement of a microphone 2 feet along the x axis
- Translational movement of a microphone 2 feet along the y axis
- Translational movement of a microphone 4 feet along the z axis
- Remote controlled via computer program
- Ability to store and reconfigure to stored x, y, and z positions via locally stored text file

Benefits

- Remote control of microphone position
- Protection of personnel from potentially dangerous recording environment
- Far less expensive than similar products

Design

Block Diagram



Structural Overview

The base of the stand will consist of a frame with 2 linear actuators to give motion in the x and y directions. A third linear actuator will be mounted on top of the first two to allow for movement in the z direction. The desired movement range will be 2 feet in the x and y directions, and 4 feet in the z direction. This means that the frame must be bigger to facilitate this amount of movement, which we will leave to the discretion of the machine shop. The entire thing will be placed on top of a platform that houses the commercial power supply.

The PCB will be mounted next to the platform that mounts the vertical linear actuator and have wires going through the top of the bottom platform to draw power. There will be sensors mounted at the edges of that platform, which will be switched when the platform nears the limits of movement. Two switches will be placed at the top and bottom of the vertically positioned linear actuator for the same purpose.

Block Description and Design Methodology

Power Supply and Voltage Regulation

The commercial AC-DC converter pulls power from a standard wall socket and outputs 5V DC. One switching regulator circuit steps down the 5V down to 3.3V DC. Three switching regulator circuits step down 5V DC to 2.7V DC, one regulator for each motor. The 2.7V lines are required to supply power to the motors, and require a higher current switching regulator than the other two outputs. Each motor draws up to 1A per phase for a maximum of 2A per motor. All three motors combined could draw up to 6A. The 5V output from the power supply will power the ATmega328 and act as a pull-up voltage for the sensors. The current drawn by the ATmega328 varies with the surrounding circuitry, but we make a conservative estimate of max 250mA. The 5V output is stepped down to 3.3V to power the Bluetooth module. The Bluetooth draws 30mA on average, but can draw up to 100mA when transmitting. An upper estimate for max power drawn = 2.7*6+5*.25+3.3*.1 = 17.78W. The commercial power supply will have a rating of 35W.

In **Figure 1** we show the circuit that will step 5V down to 2.7V. The components surrounding the LM2596 Step-Down Switching Regulator IC are based off of the example design in the datasheet.[1] The only parameter that differs is the R_2, since the example in the application notes is specifically for a circuit that steps down to 5V. Instead R_2 is chosen to be 1.2kΩ for V_{OUT} = 2.7V. We need this particular IC, because it can provide up to 3A of current, and the other IC that we use for the 3.3V regulation is not rated for high current. We use three of this one IC instead of a single higher current IC for to ease of use of this specific IC. This also allows us to treat each IC-motor pair as a separate block, simplifying verification.

In **Figure 2** we show the circuit for stepping down 5V to 3.3V. It uses the TL497 Switching Voltage Regulator IC. The TL497 has step up, step down, and inverting functionality, but we are using it as a step down regulator. The surrounding circuitry is selected and based off the design equations in the application notes of the data sheet for a step down regulator configuration.[2] I_O is 100mA, the max draw of the Bluetooth module in transmit mode, and we pick L=220µHfor convenience. Max voltage ripple is picked to be .05V, and the resulting C_o is a convenient 100µF.

Stepper Motors and Linear Actuators

These motors are used to provide translational motion through the linear actuators by spinning the screw rods. These rods will be threaded such that a full rotation will result in a linear distance of 1/10 of an inch moved. A total of three motors will provide degrees of freedom in the x, y, and z directions. Each of the three motors is controlled independently, and each has their own motor control IC and power regulator circuit. In full-step mode, there are 200 steps per full rotation, which will provide a theoretical linear resolution of 5 mils.

Motor Control Circuits

The motor control circuit helps us simplify the control of the stepper motors as shown in **Figure 3**. It not only translates the control signals into the appropriate high power waveforms, but it also saves us the use of many I/O pins on the ATmega328. For motor control we use a breakout board manufactured by Pololu that uses the A4988 Stepper Motor controller IC.[3] The ICs are driven in full-step mode, which should provide sufficiently accurate control as described in the stepper motors block description.

The ATmega328 on the main board will be the inputs to the motor control ICs. The ATmega328 will send three signals to this circuit: a step signal, a direction signal, and a reset signal. Each of the three stepper motor controllers will receive separate signals. The step is what signals the motor when to spin, and the direction signal indicates in which direction the motor should spin. The reset signal will be used to reset the phase of the stepper motor in the case that the linear actuators are moved to the limits of the frame.

Bluetooth RX

This module receives instructions sent wirelessly by the laptop, assuming the laptop has Bluetooth built in. All of the implementation of the Bluetooth communication is encapsulated between the RN-41 Bluetooth IC and the laptop using the Serial Port Profile (SPP) protocol. [4] The microcontroller sees this wireless link as a serial connection between the laptop and itself, and we can treat the data coming into the ATmega328 as such.

The implementation of the IC is powered by a 3.3V supply. There is a reset button that is active low, so we attach a normally open button switch to the reset pin with a pull-up resistor. There are two status LEDs, one to indicate that a connection has been established and one to indicate data is being transferred. These will be crucial for testing and verification. The pins and connections are shown in **Figure 4.**

There is also a circuit for translating the 5V output from the microcontroller to a 3.3V logic input to the RN-41 Bluetooth chip. The diode is reversed biased when the microcontroller output is high, and is set to 3.3V by the pullup resistor. When the microcontroller output is low, the RX input should see the forward voltage of the diode. We utilize a schottky diode for low forward voltage to ensure that the RN-41 will see a logical zero. [5] A simulation of this logic voltage translator circuit is performed in Agilent ADS and shown in **Figure 9**. It is shown that even as the DC voltage is swept above 3.3V, the output voltage does not go significantly above 3.3V.

Sensor Elements

Bump sensors signal the microcontroller when to prevent the motors from running. The sensors are normally open, but close when the linear actuators reach the bounds of the frame. This prevents damage to both the frame and the stepper motors. These sensors will also be used to calibrate the stand upon startup. When the stand is first powered up or on reset, the stand will find the origin by moving until all three bump sensors are switched closed. This eliminates the need to use rotary encoders to track the location, and instead the location can be stored as variables in the control software.

For the schematic of the sensor elements, see **Figure 5**. The circuit includes the switch along with a debouncing circuit. When closed by pressing down on a lever, the top input to the debouncer is grounded and pulls up the voltage seen by the microcontroller I/O high. When the lever is released and returned to its normal state, the pull up resistor pulls forces the output of the debouncing circuit low. The switch will be connected to the PCB containing the debouncer and microcontroller by a pair of twisted wires.

Microcontroller

We use an ATmega328 as the brains of our robotic stand, with the schematic shown in **Figure 6**. [6] It is powered by 5V from the power supply module. It will receive control signals from the laptop via Bluetooth. Commands will be received wirelessly and passed to the microcontroller for processing from the Bluetooth RX. The surrounding circuitry features a reset button with pull-up resistor and external crystal oscillator for increased clock speed from 8MHz to 16 MHz.

Upon receiving a move command, the microcontroller will send out signals from the various I/O pins to the motor control circuit. To store a position, the ATmega328 will send positional data back to the Bluetooth RX to the laptop. To restore the stand to a position, the laptop will send positional data to the microcontroller, and the microcontroller will signal the motors to move until that position is reached. The microcontroller also directly receives data from the sensors to override user inputs. We want to have as much processing done in the microcontroller as possible to minimize the amount of communication we send over the wireless channel. This will make testing easier and verification much simpler.

Computer and Control Interface

The user will enter control signals through a laptop, which we assume for the purposes of this project will have Bluetooth built in. If a particular laptop does not have Bluetooth, a simple USB dongle can be purchased. The Bluetooth will be used to broadcast wirelessly to the Bluetooth RX using the SPP profile, emulating a serial connection between the laptop and the microcontroller. An interface will be provided so the user can move the microphone and also store and return to preset configurations. The stored configurations will be saved and written to a local text file. Not only does this allow the user to save the coordinates, but it also allows the user to edit the saved coordinates at any time.

Requirements and Verification

Power Supply and Voltage Regulation

Requirement	Verification
1) Commercial power supply will deliver 5V 2) Output of switching regulator to power Bluetooth will deliver 3.3V at up to 100mA 3) Output of individual switching regulator to power stepper motors will deliver 2.7V at up to 2A	1) Hook up oscilloscope to output of the commercial supply. Plug in commercial supply to wall socket and turn on supply. a) Measure 5V output on scope. 2) Use the bench DC generator as a 5 Volt input voltage for 3.3V step down circuit. Output should be 3.3V. a) Load the 3.3V output with a 33 ohm load and measure that the current is 100 mA 3) Use the bench DC generator as a 5 Volt input voltage for 2.7V step down circuit. Output should be 2.7 Volts a) Load the 2.7V output with a 1.3 ohm load and measure that the current is 2 Amps b) Repeat for each of the three 2.7V regulator circuits

Motor control circuit

Requirement	Verification
1) Control motor movement step by step 2) Must turn both clockwise and counter clockwise	1) Set DIR to high. Connect output leads to oscilloscope OUT1A with OUT1B to one channel and OUT2A with OUT2B to a second channel, with B's being grounds. Connect STEP lead to a debounced switch so that you can toggle the input to STEP. a) Toggle switch on and off manually and observe the output on the oscilloscope to verify that each step corresponds with a rising edge. Repeat 20 times 2) Set DIR to high. Connect the output leads to an oscilloscope OUT1A with OUT1B to one channel and OUT2A with OUT2B to a second channel, with B's being grounds. Connect the STEP lead to a function generator and send a square pulse with a period of 5us. a) Observe that the output signals are 2 square waves with OUT2 having a 90 degree phase lag. b) Connect DIR to ground. Observe that the output signals are 2 square waves with OUT1 having a 90 degree phase lag.

Stepper Motors

Requirement	Verification
1) Motor should not skip steps, should give a step every time it's instructed to 2) One revolution should match lead screw spec for linear motion 3) Move both clockwise and counterclockwise 4) Actual linear distance traveled to theoretical distance traveled should have no more than 1% error	1) Set DIR to high. Run test program from microcontroller to give a square wave of period 5us with 1600 cycles as an input to the STEP pin of the motor drivers so the motor will give 200 steps. a) The motor should move one revolution. 2) Move a full revolution using test program from above a) Measure the linear distance traveled on the lead screw with ruler. it should be 1/10" (2.45 mm) 3) Repeat above tests, but set DIR low for opposite direction of rotation. 4) See Tolerance Analysis

Bluetooth RX

Requirement	Verification
1) Needs to establish connection with computer 2) Needs to be able to receive data to computer 3) Needs to be able to send data back to computer 4) Data transfer should be accurate 5) Should communicate through a wall from at least 5 meters away	1) Establish connection with computer. Connection status LED indicator should light up, showing that connection has been made 2) After establishing connection with computer, send characters in the terminal interface. Data status LED indicator should be flashing, showing that data has been received by the Bluetooth module 3) Program the microcontroller to send arbitrary location data to the computer, and echo the received data to the screen. Compare for correctness and repeat 20 times 4) See verification for Computer and Control Interface block 5) Place computer at least 5 meters away through a wall and repeat all above test procedures

Computer and Control Interface

Requirement	Verification
1) Store and read data from computer memory 2) Send accurate serial data to the microcontroller over Bluetooth	1) Store data in a .txt file by running test code from Visual Studio and verify by opening the file a) Edit the file and load it back into the program to verify reading data 2) After receiving inputs keyboard, echo transmitted data on the screen before they're sent through Bluetooth to make sure they are read correctly a) Verify data is sent by checking connection and data status LEDs on Bluetooth RX module b) Verify data is correctly sent by reading output of Bluetooth RX module on logic analyzer

Sensor Elements

Requirement	Verification
1) Pushing down the lever on the pumper switch should change the output from low to high. 2) Debouncing circuit should minimize undesired, rapid switching between switching of high and low states	1) Construct the circuit as shown in the schematic and supply a voltage to the pull up resistors. Connect an oscilloscope to the output of the debouncer circuit. The voltage should show high. a) Press and hold the lever down. The voltage should now show low on the oscilloscope. b) Let go of the lever. The voltage should now again read high on the oscilloscope 2) Test the debouncing circuit by directly putting a square wave with period of 1ms in place of switch input. a) Observe on the oscilloscope that there is no rapid bouncing between states in between input switches

Microcontroller

Requirement	Verification
1) Properly send and receive signals from the Bluetooth RX block 2) Properly send signals to motor control block 3) Properly receiving signals from sensor block	1) See verification for Computer and Control interface block 2) Place microcontroller in Arduino development board. Use LEDs to read I/O pins corresponding to motor control outputs. Run test program that sends a square wave of period 1 second to STEP output, checks for clockwise and counterclockwise operation, and phase reset functionality. a) Verify for each set of motor controls for x,y and z axis motor control outputs 3) Connect debounced bumpers to microcontroller and run microcontroller test program. a) Motor control output signals should indicate phase reset when they receive a signal from their respective bumpers

Ethical Issues

1. to accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment;

All wireless communications devices we use in this project, the Class 1 Bluetooth module in particular, are certified by the FCC. [7]

3. to be honest and realistic in stating claims or estimates based on available data;

We will not overstate the accuracy or abilities of this robotic stand. There will be some amount of error involved in the distance the linear actuator will move, whether it be from imperfect lead screw, imperfect motors, or imperfect controls. We will find this error through the tolerance analysis and be upfront with the results to Mark from Pogo Studios and the 445 staff.

6. to maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations;

We realize that the mechanical implementation of the linear action in the robotic microphone stand will be primarily lead by the experts in the machine shop to manufacture the frame and linear actuator mechanism. Attempting to do these ourselves without the proper safety training would not only result in a less polished product, but would risk injury to ourselves and the machine shop staff. However, we will be applying our engineering training in circuits, communications and controls to the best of our ability in all the other aspects of the design and production.

7. to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others;

Working in a group of two, we will provide honest criticism of our contributions to the project and do our best to correct technical errors. We will also properly credit the contributions from the staff in the machine shop and any other staff or faculty we collect advice from.

Tolerance Analysis

The most critical component of the microphone stand is the stepper motor control. We must be able to gauge how fine it can resolve movements in the x, y, and z directions. Also, we need to be able to minimize the error between the expected distance traveled and the actual distance traveled.

The theoretical resolution, as stated in the stepper motor block description, is 5 mils. This linear resolution is smaller than two tenths of a millimeter. We have to test if we can actually achieve this by measuring the distance between the platform and the frame before and after a single step. If we are able to achieve such a small resolution, it may be advantageous to change the lead screw to trade linear resolution for movement speed in the linear direction. Using lead screws with different thread inclines will allow us to make this tradeoff as we move ahead with the project.

The second part to the tolerance analysis is measuring the error between the expected distance and actual distance traveled. We can measure this by writing a program to tell the motor to move to take a large, predetermined number of steps. We can calculate the theoretical distance it should travel by multiplying the number of steps taken by the distance each full rotation travels, divided by 200.

$$theoretical\ distance\ traveled = \frac{distance\ per\ full\ screw\ rotation * number\ of\ steps}{200}$$

We should then compare this to the actual measured distance traveled. By making the motor take a large number of steps, the absolute error is maximized so we can realistically measure it with a meter or ruler. We are aiming for a relative error of less than 1%.

$$error = \frac{|actual\ distance\ traveled - theoretical\ distance\ traveled|}{theoretical\ distance\ traveled} * 100$$

Parts List, Cost, and Schedule

part	quantity	price	total	usage
motor block				
Stepper Motor: Bipolar, 200 Steps/Rev, 35x36mm, 2.7V, 1000mA	3	$15.95	$47.85	motor
motor control block				
A4988 breakout board	3	$12.95	$38.85	IC for driving the stepper motor
P1.00KFCT-ND	6	$0.1	$0.6	1k resistors for A4988
sensor block				
Snap-Action Switch with 16.7mm Lever: 3-Pin, SPDT, 5A x6	6	$0.75	$4.5	bump sensor
P10.0KFCT-ND	6	$0.1	$0.6	10k resistor for pull-up
power supply block				
LS35-5	1	$19	$19	power supply 5V, 35W
LM2596	3	$1.78	$5.34	high current switching regulator for 2.7V supply
1N5822	3	$0.46	$1.38	diode for LM2596
DO5040H−683ML	3	$1.62	$4.86	68uH inductor for LM2596
P5155-ND	3	$0.41	$1.23	470uF cap for LM2596
P5183-ND	3	$0.44	$1.32	220uF cap for LM2596
P1.00KFCT-ND	3	$0.1	$0.3	1k resistors for LM2596
P1.20KFCT-ND	3	$0.1	$0.3	1.2k resistors for LM2596
TL497ACN	1	$2.02	$2.02	low current switching regulator for 3.3V supply
P1.20KFCT-ND	1	$0.1	$0.1	1.2k resistors for TL497A
P2.10KFCT-ND	1	$0.1	$0.1	2.1k resistors for TL497A
P5182-ND	1	$0.33	$0.33	100uF cap for TL497A
P12395-ND	2	$0.51	$1.02	150uF cap for TL497A
595-1458-1-ND	1	$1.11	$1.11	220uH inductor for TL497A
PWR263S-35-2R50J-ND	1	$3.25	$3.25	2.5 resistor for TL497A
Bluetooth RX block				
Bluetooth SMD Module - RN-41	1	$24.95	$24.95	Bluetooth RX
PB-126	1	$0.33	$0.33	NO pushbutton
P1.00KFCT-ND	5	$0.1	$0.5	1k resistors for RN41
HLMP3507	2	$0.15	$0.3	status LEDs for RN41
P220ECT-ND	2	$0.1	$0.2	220 ohm resistor for status LEDs
microcontroller block				
ATmega328	1	$2.88	$2.88	microprocessor
PB-126	1	$0.33	$0.33	NO pushbutton
520-HCA1600-SX	2	$1.18	$2.36	16MHz crystal oscillator
1C25Z5U223M050B	2	$0.15	$0.3	22pf caps for crystal oscillator
P10.0KFCT-ND	1	$0.1	$0.1	10k resistor for pull-up
PCB fabrication	1	$35	$35	PCB
PARTS TOTAL			**$201.31**	

Labor	**wage**	**time**	**total**	
Dennis	$35/hr.	160hr	$14000	
Alejandro	$35/hr.	160hr	$14000	
LABOR TOTAL			**$28000**	
GRAND TOTAL			**$28201.31**	

Week of	Dennis	Alejandro
8/26	Brainstorm ideas, find partner	Brainstorm ideas, find partner
9/2	Brainstorm ideas, find partner	Brainstorm ideas, find partner
9/9 – Approved RFA	Brainstorm ideas, find partner	Brainstorm ideas, find partner
9/16 – Proposals Due	Pick microcontroller, research Bluetooth	Discuss mechanical design with machine shop
9/23 – sign up for Design review	Pick Bluetooth/wireless implementation	Motor control circuit, pick motors
9/30 – Design review	Set up microcontroller environment	Finalize design with machine shop, order parts
10/7	Breadboard microcontroller and breadboard Bluetooth RX	Breadboard and test sensors, motor control circuit
10/14	Design and submit PCB v1	Assemble structure, test motors
10/21	Breadboard and test power supply and regulation	Program translational movement control
10/28 – Individual progress review	Assemble parts onto first PCB, testing	Program movement control and startup routine
11/4 – mock presentation signup, mock-up Demos	Design and submit PCB v2	Program store and recall functionality
11/11	Testing	Design UI
11/18 – Thanksgiving break	Design and Submit PCB v3 (if needed), prepare demo	Design UI
11/25 – presentation signup	Presentation and final paper	Optimization
12/2 - Demos	Presentation and final paper	Optimization
12/9 – Presentations, final paper	Presentation and final paper	Optimization

Figure 1, 5V to 2.7V step down switching regulators. Taken from application notes [1]

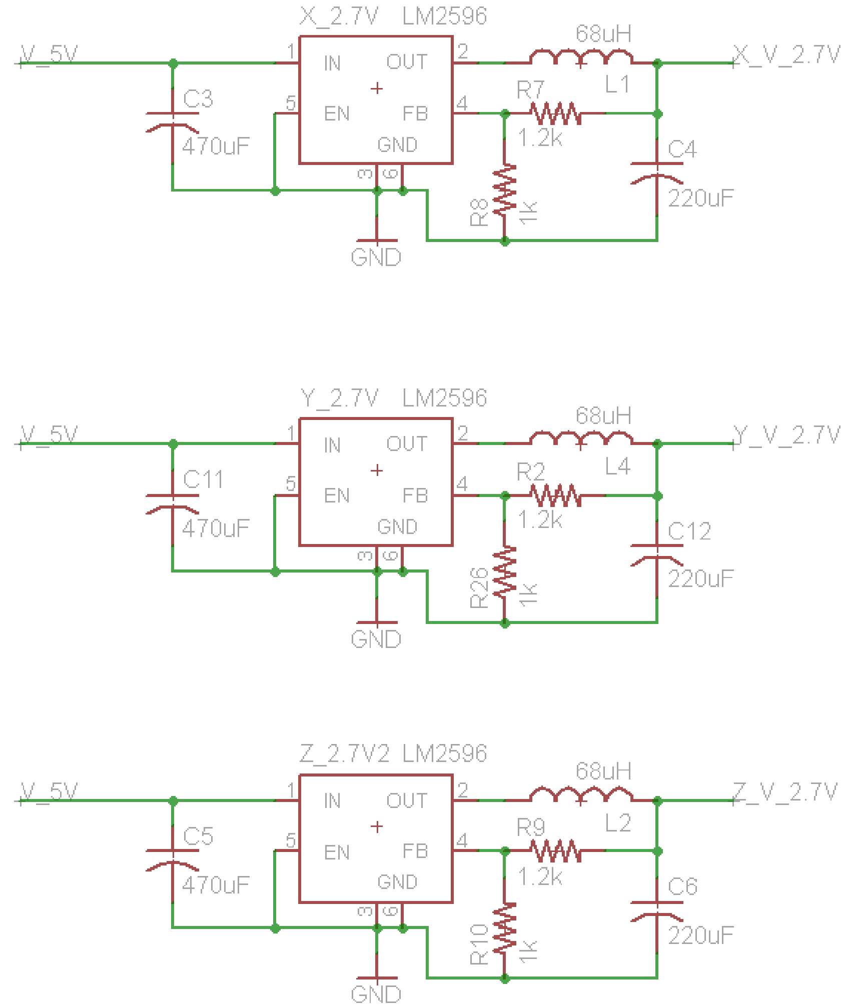


Figure 2, 5V to 3.3V step down switching regulator. Taken from application notes [2]

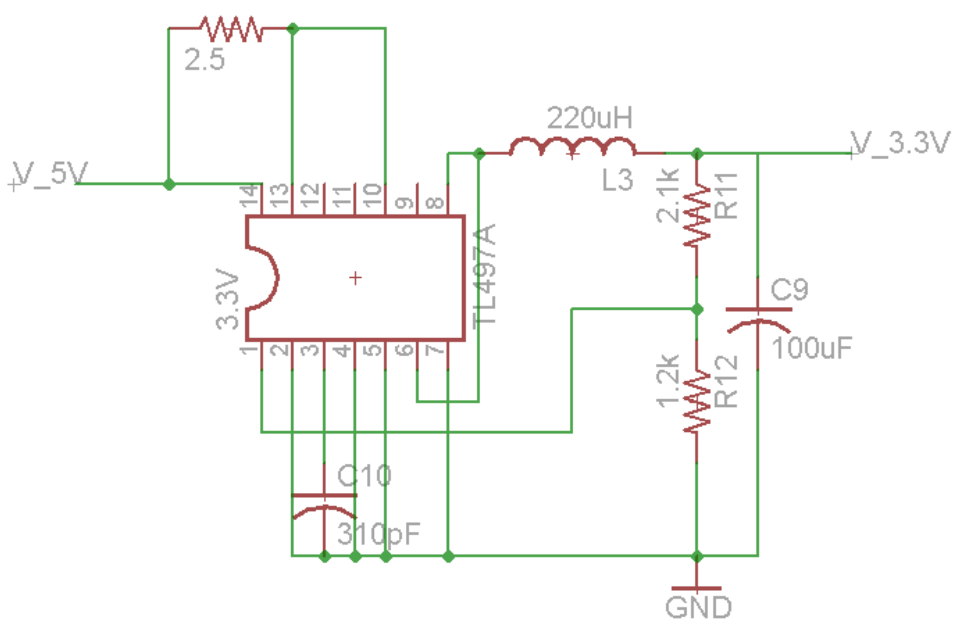


Figure 3, Stepper motor control circuit

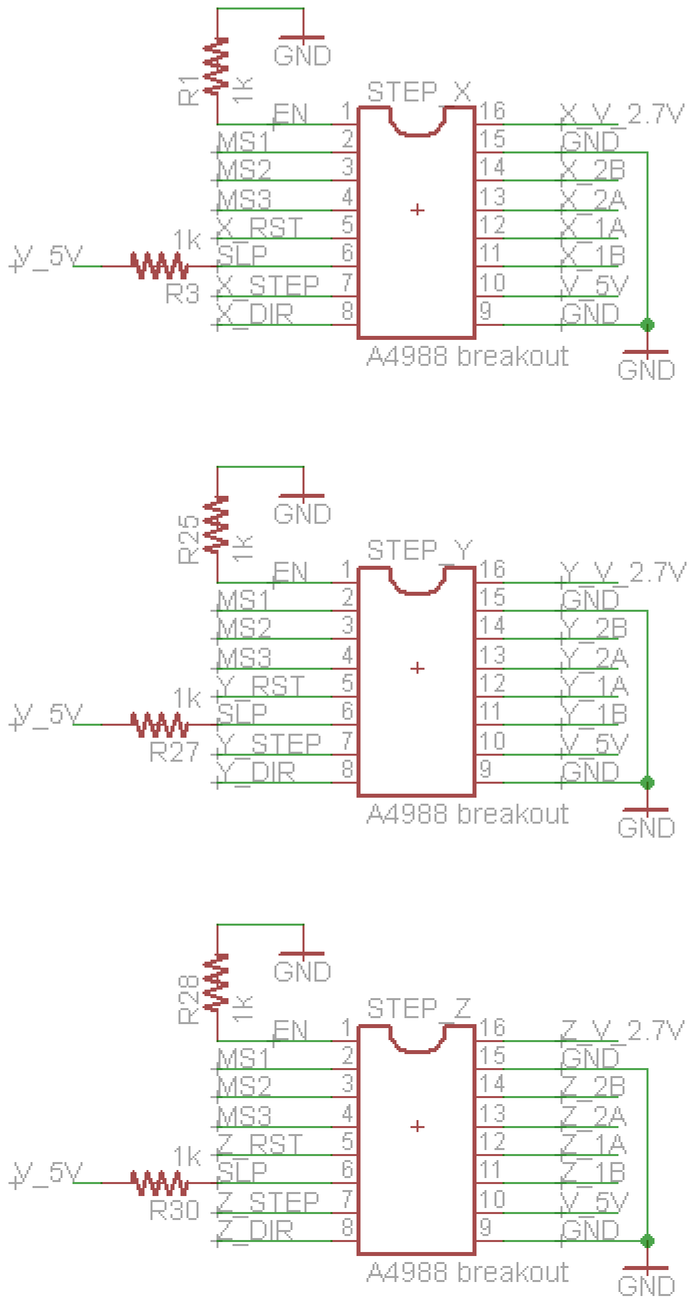


Figure 4, the RN-41 Bluetooth module

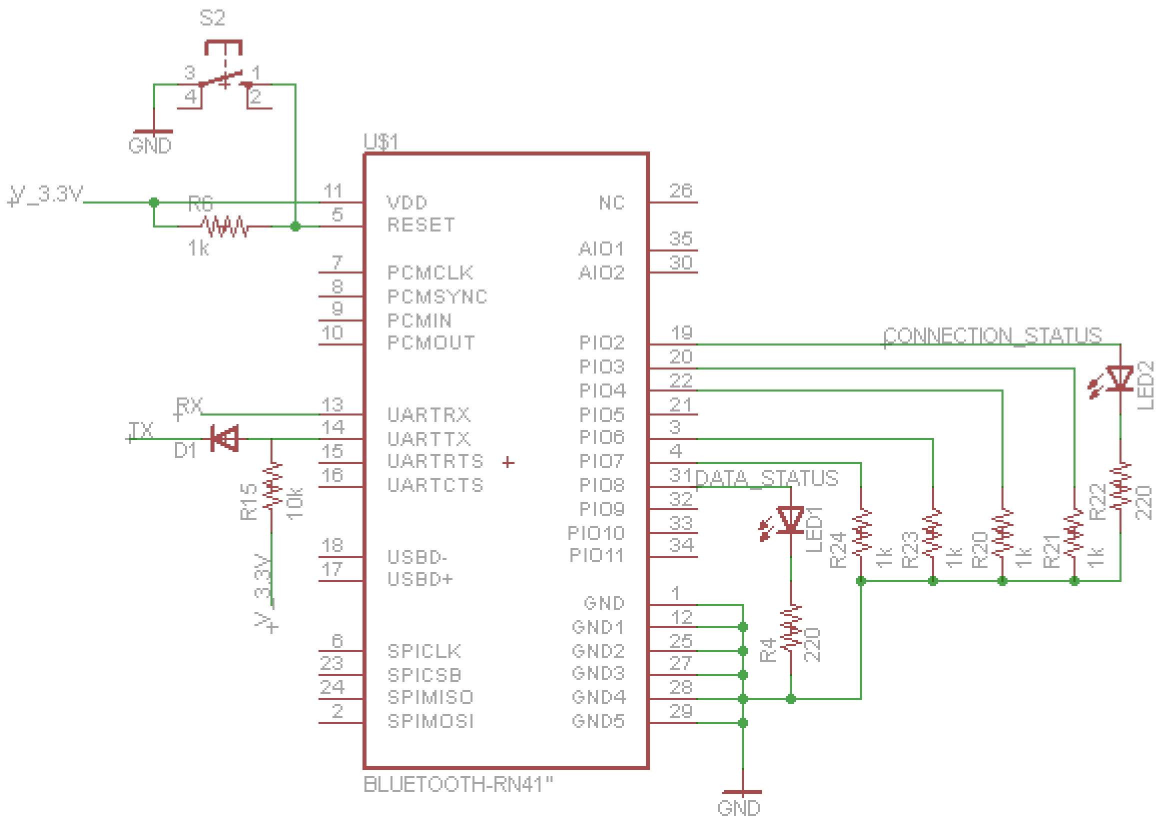


Figure 5, Bump sensor elements with debouncer. One of three identical circuits shown

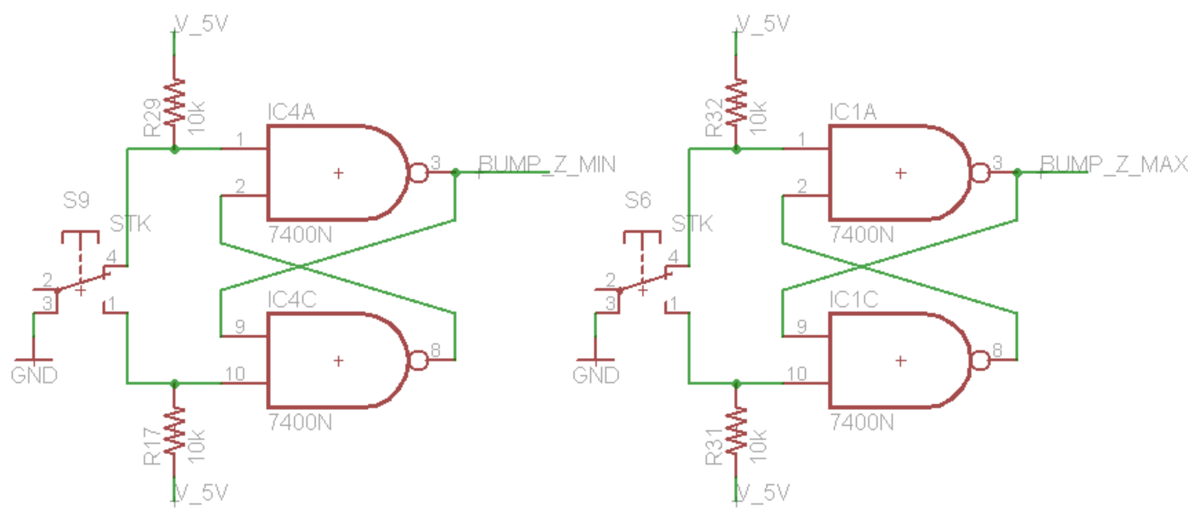


Figure 6, ATmega328 microcontroller

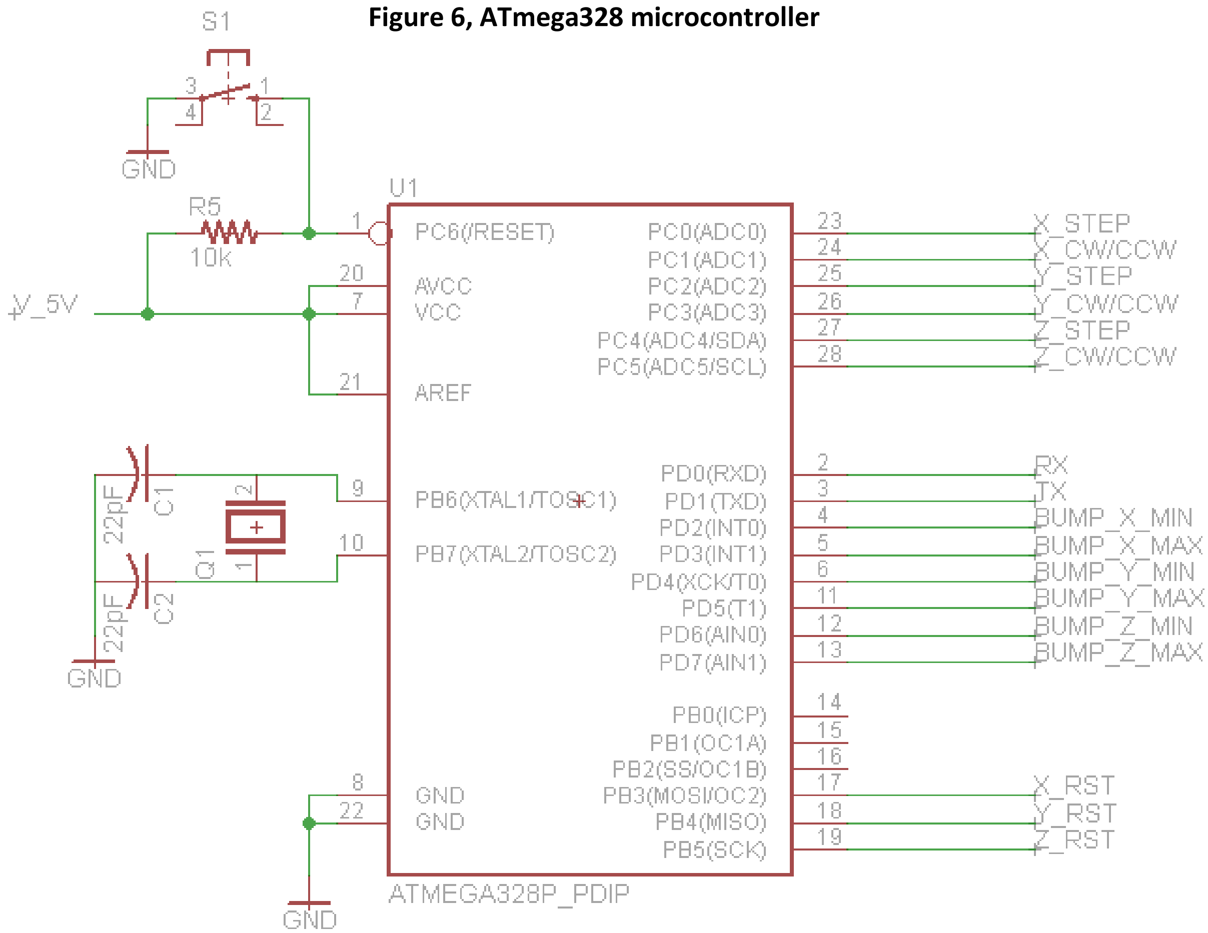


Figure 7, the microcontroller software flowchart

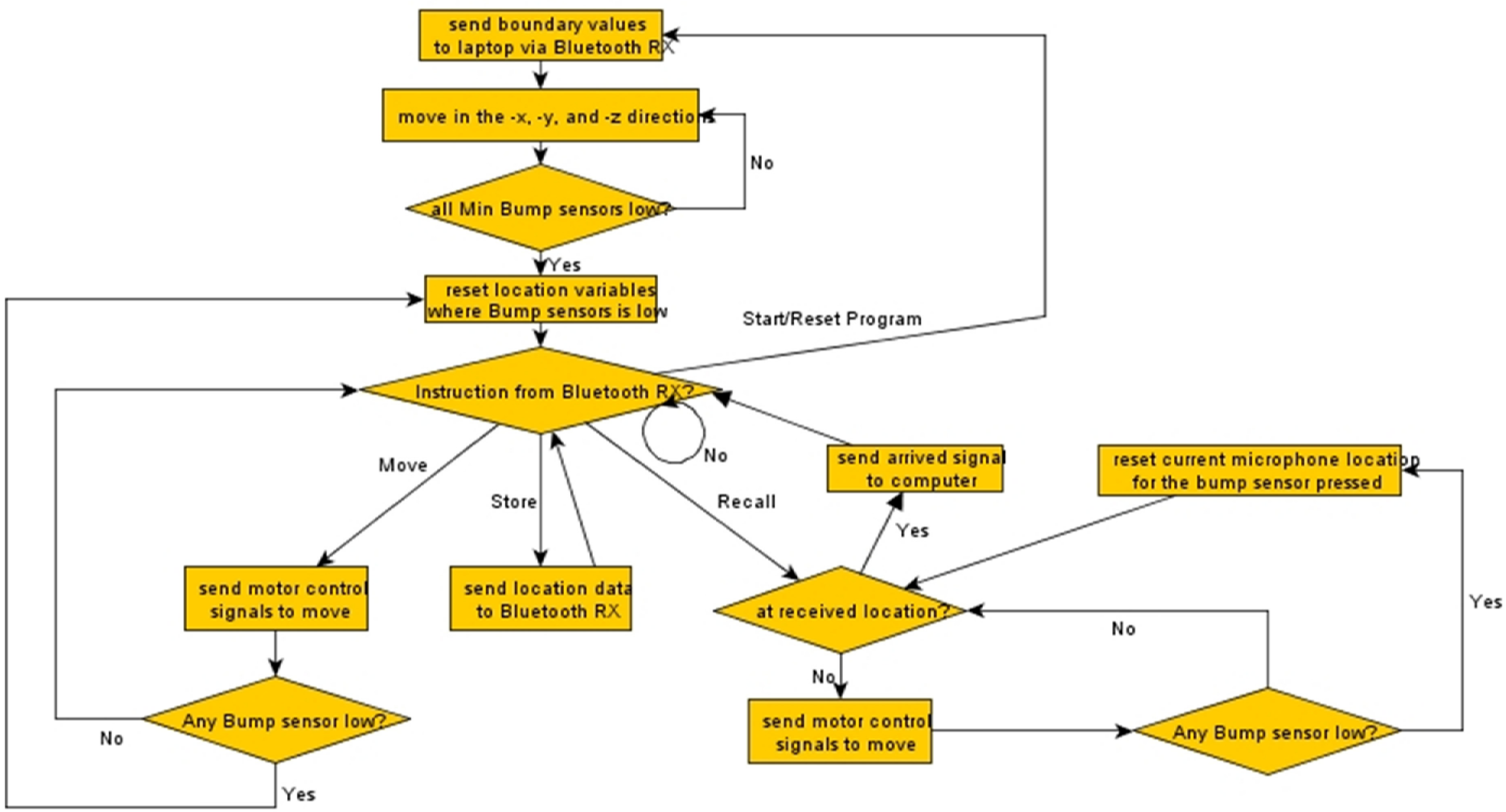


Figure 8, the software flowchart for the client software

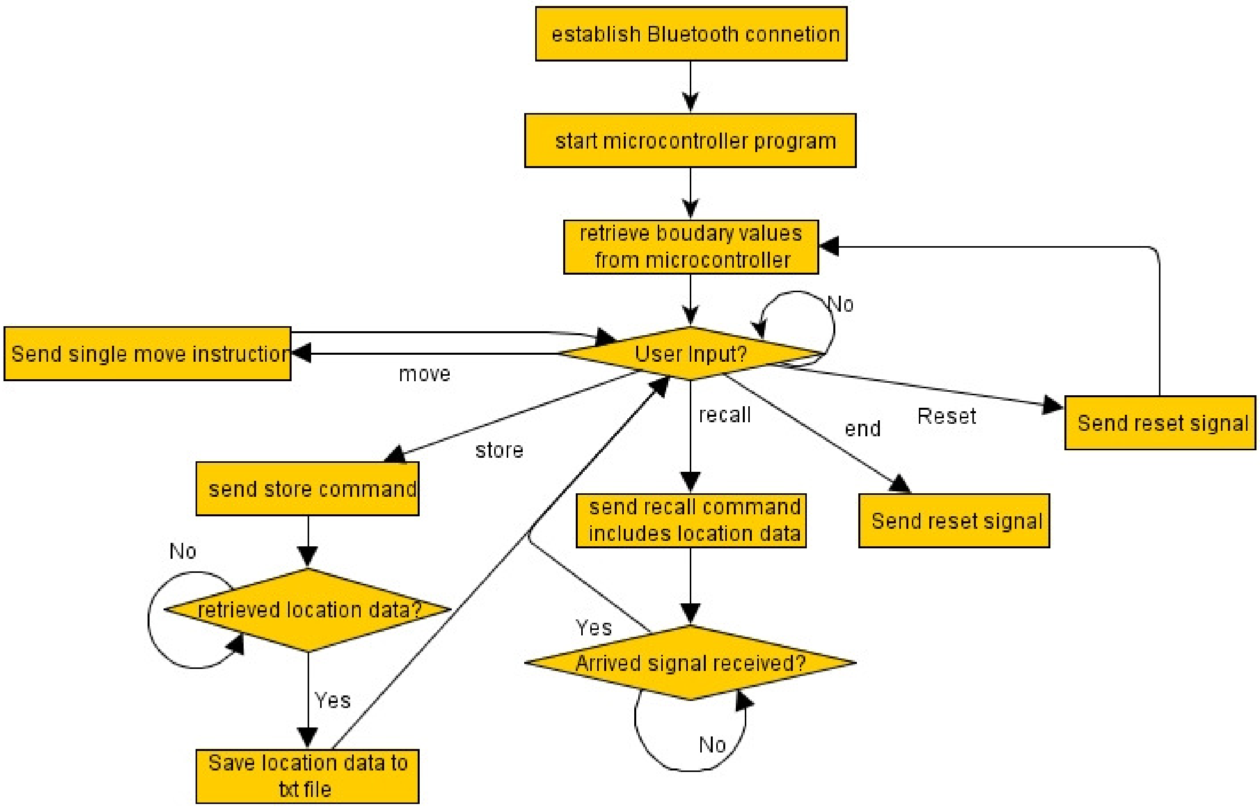
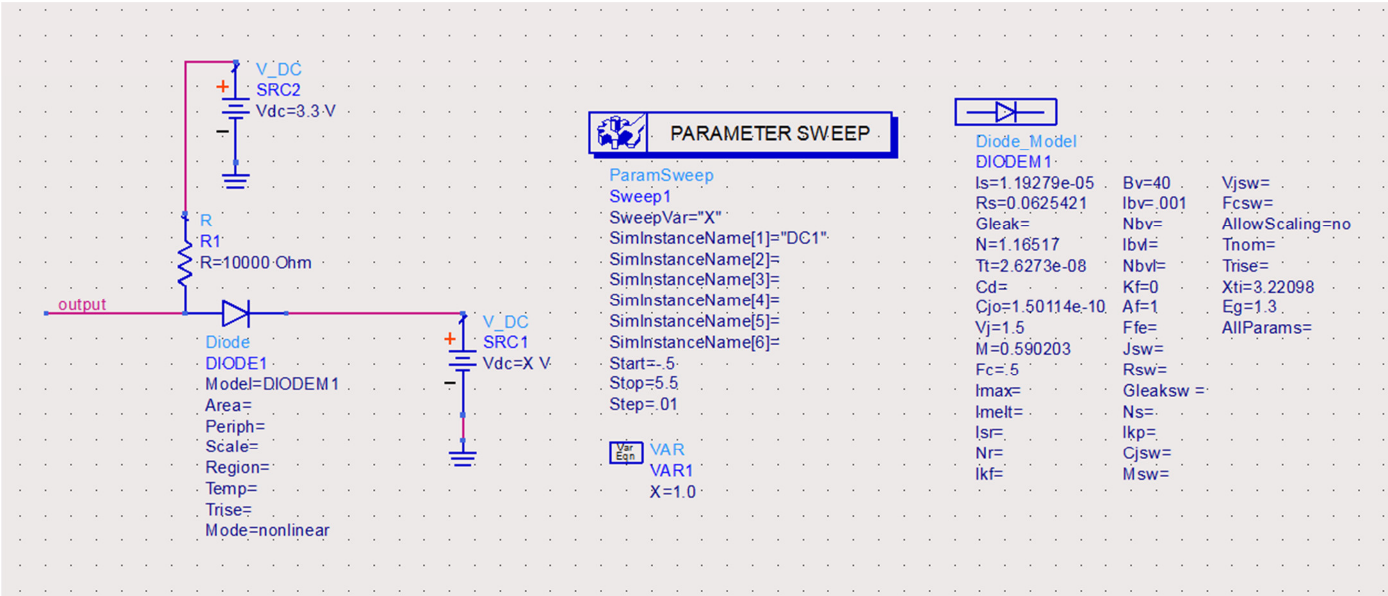
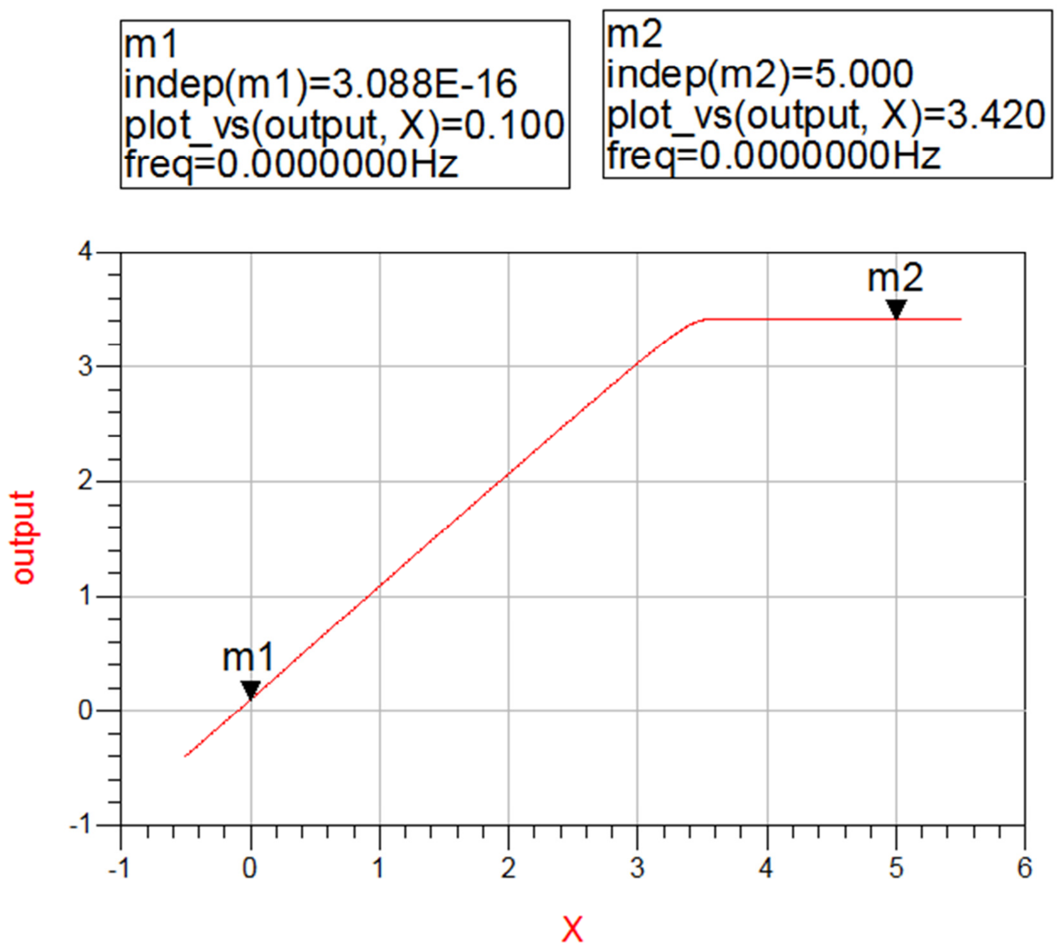


Figure 9, a simulation in ADS of the logic voltage translator circuit with schematic and plot





References

1. On Semiconductor, LM2596 3.0A, Step-Down Switching Regulator. [Online]. Available: http://www.onsemi.com/pub_link/Collateral/LM2596-D.PDF
2. Texas Instruments, TL297A 500-mA Peak Step-up, Step-down, Inverting Switching Voltage Regulator. [Online]. Available: http://www.ti.com/lit/ds/symlink/tl497a.pdf
3. Allegro Microsystems, Inc., A4988 DMOS Microstepping Driver with Translator and Overcurrent Protection. [Online]. Available: http://www.pololu.com/file/download/a4988_DMOS_microstepping_driver_with_translator.pdf?file_id=0J450
4. Bluetooth Special Interest Group, Serial Port Profile (SPP). [Online]. Available: https://www.bluetooth.org/Building/HowTechnologyWorks/ProfilesAndProtocols/SPP.htm
5. Sparkfun, Sensor Interfacing. [Online]. Available: http://www.sparkfun.com/tutorials/65
6. Atmel, 8-bit AVR® Microcontroller with 4/8/16/32k Bytes In-System Programmable Flash Datasheet. [Online]. Available: http://www.atmel.com/Images/doc8161.pdf
7. Roving Networks, RN-41/RN041-N Class 1 Bluetooth Module. [Online]. Available: http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Wireless/Bluetooth/rn-41-ds-v3.3r.pdf