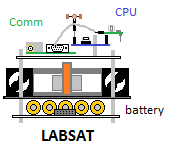
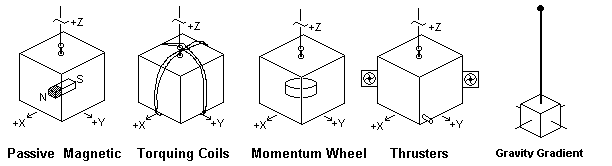
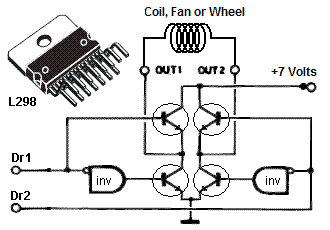
**EA-467 ATTITUDE CONTROL - B-dot LAB** (rev a) 13 Dec 2017

Moved questions to the end – post-lab… and separated Manual-tabular test from RF-telemetry test

**Introduction:**  This is the third part of the Attitude Determination and Control series. The first portion predicted and then measured the spacecraft’s moments of inertia. The second lab developed a sun sensor and magnetometer for attitude estimation. In this lab an ADCS board has been added to the Labsat that contains magnetorquing coils, a momentum wheel and a pair of thrusters (fans). You will develop Bdot control methods to eliminate initial rotation rates and then in the next lab, use thruster control to align to the sun (sun tracking).



**Part G. Passive Magnetic Control:** This is the simplest form of attitude control. Our original PCSAT1 launched in 2001 and RAFT in 2006 used a passive magnetic control system for a slow spin for ***thermal control***. PCSAT1’s permanent magnet was aligned along its Z axis. Answer additional questions about this method in the Post Lab

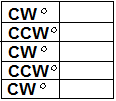


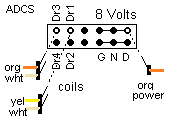
**ADCS System Driver:** For many of the ADCS methods we need power circuits to drive the coils, motors, or fans in either direction. A common circuit for this purpose is called an “H” bridge because of its apparent schematic structure. Your ADCS board contains an L298 dual “H” bridge driver chip to take the 0 and 1 signals from the CPU and use them to turn on power to a circuit in either direction as shown to the right (with a coil load). The transistors are turned on in diagonally opposite pairs to energize the load one polarity or the other using the HIGH or LOW commands in your CPU on the two driver pins Dr1 or Dr2 to activate the coil (or fan or motor) either way. The chip actually has two independent drivers and the second one uses inputs Dr3 and Dr4 to drive a second output pair such as for the second magnetorquing coil.

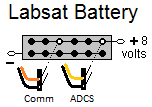
**Part H: Bdot Control with Torquing Coils:** One of the simplest yet common control laws is the Bdot control law. It uses a magnetometer to measure the local magnetic field in each of the three axes and then compute their rates. It then energizes the spacecraft torque coils to counteract the angular rates. This is used to de-spin a spacecraft or to establish a spin of a desired rate.

Measure the area dimensions of the X coil for use in the Post Lab \_\_\_\_\_ by \_\_\_\_\_\_\_.

The Torque Coil and Bdot control are relatively weak torques, so we will make ***with and without*** comparisons of the damping of spacecraft rotation hanging on a string. Just timing the rotations will not do, because the period of movement back and forth is based more on the harmonic oscillator physics of the string and not on the angular rates. What is damped by the Bdot control, is the instantaneous rate of rotation, and thus the total angular difference traveled on each oscillation. This can be subjectively observed by the angle of the +X pointer above the compass rose.

**1) First, test with no B-dot Control.** Before connecting your LABsat to the PC, hang it on the string and start from neutral with 3 turns Counter Clock Wise to “0” over your compass rose. Start the stopwatch and release very gently (no torques). Count the number of turns (x 360 deg) plus the final angle to the stop on every iteration of the decaying oscillation for the first five velocity nulls. Record the total angle traversed and the time for each half period.

**2) Setup for B-dot:** First, make sure you have the same LABSat that you had in the last lab so you can use your magnetometer calibration constants. Configure the ADCS connection block as shown in the sketch to the right. Plug the Org/Wht X coil and Yel/Wht Y coil wires into Dr3/Dr4 and Dr1/Dr2. Plug in the single Orange ADCS power to the 8 volt power bus. The CPU (Arduino) should have a yellow/black power cord to the 8 volt Battery tray tap (in the next step) Do not confuse this with the yellow/black going to the reaction wheel.

**3) Battery Power Setup:** To fly this LABsat you must power it from the LABsat battery (not the USB power). This battery has several plugin locations on its power block as shown here. The bottom row is ground and the top row is +8v. The power plugs are not symmetric, be sure to plug them in with the black wire down as shown here. Plug in the yellow/black pins from the Arduino ADCS system. Later you will plug in the orange/black line from the Comms board.

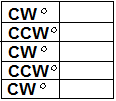
**4) Download the latest program MicroMagBdotArduino16h** from Blackboard and immediately resave to your drive with a new filename. Change the initial lines to your XYZ cal magnetometer constants as you determined in the last lab. The **main loop** is where ***you will implement the B-dot control law*** in the four commented-out lines. The other routines will have already been written to ***read*** the magnetometer, to ***drive*** the coils and to ***downlink*** telemetry. Look over these other routines to gain a full understanding of what they are doing.

**5) Modify the code to implement Bdot Control:**

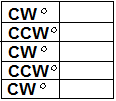
**Rates:** The first two missing code lines compute the actual angular rates in X and Y. Without a gyro, we can compute the rate as simply the difference in angle position from the last sample by subtracting the old value of each component (i.e. Xold) from the new component (i.e. X). Notice how we get the old values at the end of the LOOP where we save the previous value of X and Y in variables named Xold and Yold. This difference over the time of the loop (650 ms), gives a Bdot measurement over this increment of time.

***The second pair of missing code lines implement the B\*dot Control Law for the X and Y axis.*** You will calculate a Magnetic Moment (Mx) based on a Gain K times Bdot (computed as above) for the X and Y axis. The gain factor has been found to work best at about 0.3 for this lab to give a Magnetic Moment in the range of 1 to 10 for the expected rates. Of course, you also need to add a minus sign to the calculation of Mx to make the control torque act in the opposite direction from the rate.

**6) Torque Coil Driver:** Rather than use a linear amplifier and analog driver for the coils, we use the much more efficient on/off digital pulse technique. The average torque will be proportional to Mx or My by varying the duty-cycle according to that input. A simple FOR loop runs 10 times per cycle through the Bdot loop. The values of Mx and My (between 0 and 10) determine how many times out of 10 the given coil is pulsed ON. See the PulseCoils function at the bottom of the code. Notice that you will get a maximum torque with M = 10 which results in 10 pulses in 10 cycles, but half the torque (half as many pulses) if M=5 for example. The Gain factor K is used to scale your Bdot algorithm to give torques in the range 0-10 for the expected rates. To save time, rather than iterating to find the best K value, assume the best control value of K is 0.3 as it already is in the code. Confirm that your telemetry is updating every 650 ms or so.

**7)** **Test your code:** You should be able to disconnect the USB cable and the Arduino will continue on battery power as configured in step 3 above. TheLED on the upper right of each coil shows the axis that ***seeks*** north when energized.Do a quick sloppy verision of the step-1 spin testwithout data. Observe if the proper coil (and LED) is energized relative to local North to counteract the string spin, that is, energized on the west side when spinning CCW and on the east side when going CW. You can also see the magnitude of the coil torque by observing the duty-cycle modulation of the coil (and LED). The gain is about right when the LED’s are ***near-fully-ON*** per pulse cycle when the spin is at the max rates and orthogonal to North and moving at a reasonable rate. At slow rates the pulse width of the LED’s should be less.

**Manual Flight Test:** Now with Bdot code running, re-do the spin test of step 1 and record the degrees covered in each half period. This is only 5 data points but you should see the decay in rates by comparing this test with Bdot to the step 1 test without.

**MOTOR Test:** Reconnect the USB cable and, reverse the SIGN of the Bdot Control code lines from negative to positive, reconnect your coils as in step 2, and repeat a spin test. What do you expect will happen? What do you observe?

**TELEMETRY Flight Test:** If there is time, you can also configure the telemetry on your LABcube to transmit to a ground station receiver and capture the angles and rates during the test. Unfortunately, all of the LABsats share the same telemetry channel, so we will use the Link budget to our advantage. By placing your own receiver only 1 foot from your labcube, how much more power will your link provide than the link to the other nearest LABcube (assume 8 feet away)? \_\_\_\_\_\_dB. (remember 1/R^2 path loss). Given that the FM Capture effect of FM modulation only takes about 10 dB stronger than other signals to capture the receiver, what is your calculated link margin \_\_\_\_\_ dB to your nearest neighbor?

**A) Prepare for transmitting.** Now plug in the org/blk power wires from the Comms module to the Labsat battery as in step 3. Prepare to connect the Arduino telemetry data output white wire to the COMMS board TX input line (pin 2) shown here. Execute PuTTY at 9600 baud on your right hand PC with your radio ground station receiver connected to its serial port and momentarily confirm your LABsat transmits data by briefly touching the white telemetry wire to the COMM module pin 2. Keep the COMMS off when not in use to avoid interference to the telemetry channel.

**B) Again Hang your satellite on the string** (neutralized). Rotate it 3 turns Counter Clock Wise and stop above “0” on the compass rose as before. When the frequency is clear, plugin the COMMS and very gently release with no torques. If your putty is full of telemetry before you start the test, you should re-start Putty so the data begins about when you release the satellite. After the 5 reversals, copy/Paste the Putty data into a .TXT file with NOTEPAD. Save this for importing into Excel post lab.

**C) Repeat the test without Bdot control**. Begin a new capture file in PuTTY. Repeat step 1 but now with the power to the coils dis-connected at the Coill Driver chip. Again, wind three times CCW and release at the same time you start your Putty capture. You will compare this data in the post lab analysis to see if the Bdot control law damped the rotation rates on the string. Save the capture file.

**Post Lab:**

a) PCSAT’s forced spin is for thermal balance. Its orbit is nearly polar and 100 minutes. How many flips in attitude will occur for each orbit of the spacecraft?\_\_\_\_ At this rotation rate, and when spinning in the plane of the sun, what is the duration of each sun exposure on a side of the cube shaped satellite? \_\_\_\_\_. This exposure is over +/- 90 degrees normal to that face. Given that the average of a cosine over this angle is 50%, what is the effective Sun exposure of this face per orbit? \_\_\_\_\_\_. If PCSAT-1 had a Z axis antenna and this passive control method, approximately what attitude (antenna polarization) will an observer on the ground at the equator see of PCSAT-1 passing to the east or west? \_\_\_\_\_\_ What will an observer at mid-latitude see with the satellite to the north or south? \_\_\_\_\_\_ What about an observer at the poles? \_\_\_\_\_\_\_

b) Your LabSat has two orthogonal 20 ohm coils wound from 100 turns of #30 wire operated from the 8 volt bus. Calculate the current in the coil when energized. \_\_\_\_\_\_\_ From your measurement of the X coil dimensions, compute the Magnetic Dipole of the coil \_\_\_\_\_\_\_\_\_\_? (use formula for the magnetic dipole of a current loop).

c) Compare the tabular decay of the manually recorded spin tests from steps 1 and 7. Is it obvious tht the Bdot control law reduced the rates, and therefore the total angular distance covered in the same amount of time?

d) Import the two Putty capture files in .TXT format. Use the import wizard to eliminate unneeded columns and then plot the X and Y coordinates over time for each file and then compare the results. You can also plot just the X position and the Mx rates to see that they should be in quadrature. That is, when the coil is facing North or South with maximum Field, then the Mx is zero since the rate is zero. And when X is approaching zero, it is moving at the maximum rate and so Mx will be maximum. If there is extraneous data points at the start of your file before you actually released the satellite, delete them so your data begins at the time of release.

e) Was the Bdot control effective at damping the rate of rotation? Discuss the analysis of your data and discuss the effect of Bdot on the LABsat.

**Lab Report:** Combine this lab with the two previous lab exercises for your ADCS report.You will write one lab report covering the three different parts of the ADCS lab.