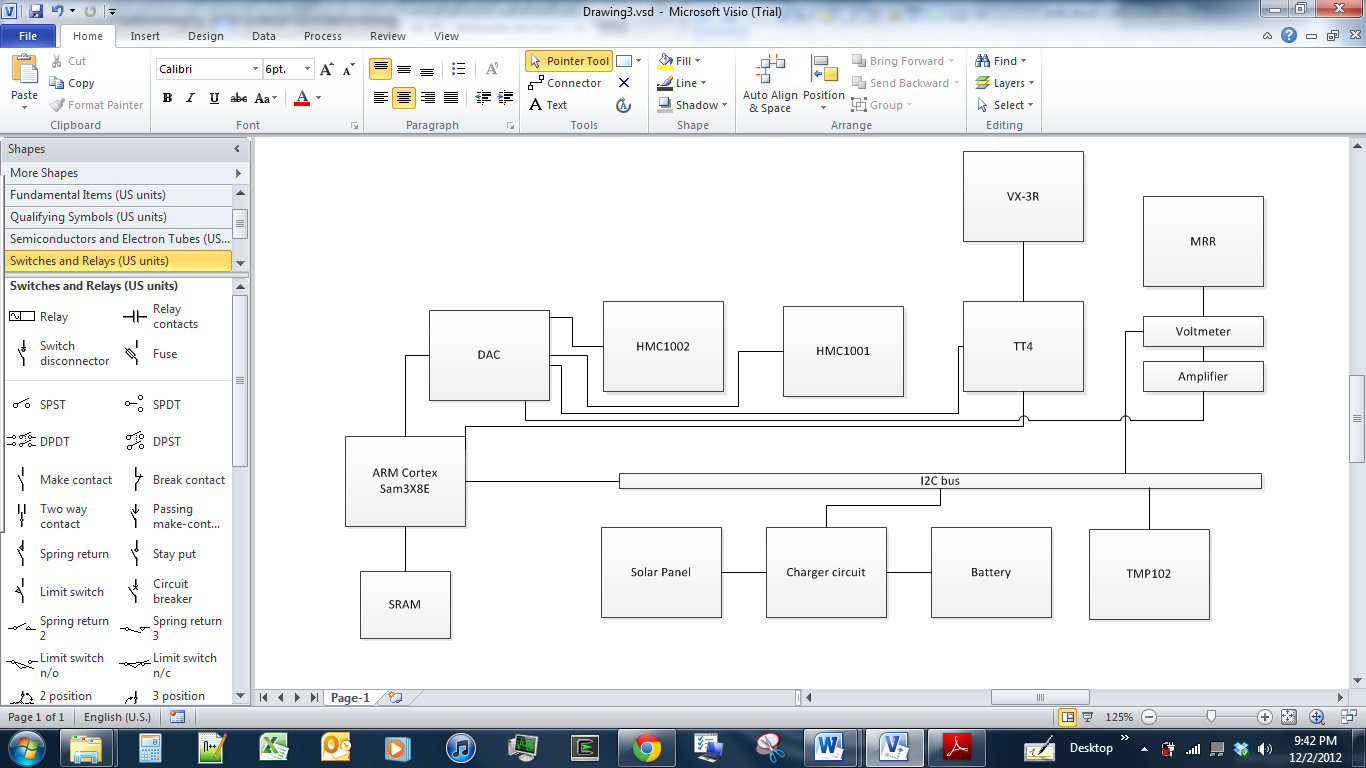
**Control system**



**Controller**

Our control system will be based on the Atmel SAM3X8E ARM Cortex-M3 CPU [1]. We chose this chip because the latest Arduino board, the Arduino Due [2] is based around it. Arduino is accessible and we have many open source references. The schematics and programming libraries are available to us so that functions such as the I2C capabilities are already made. This makes implementing it more efficient. This board also includes an onboard voltage regulator that we will use to create our 3.3 V rail. Like all chips in that family, since this chip operates at 3.3 volts it uses less power than some alternative Arduino boards. This board was also selected because it is clocked at 84 MHz.

We will have two main types of communication with our devices, analog and I2C. Analog will be handled with the DAC that comes with the Arduino. I2C will be handled with the built in I2C functions. This board supports multitasking and that will be useful to talk to both lines at the same time.

**DAC**

The DAC will be used to interface the magnetometers and radio driver. Each magnetometer has 4 analog inputs that will be connected. They also have one digital input each to enable and reset them. The digital inputs will be connected directly to the SAM3X8E digital inputs. The other device that has analog inputs will be the radio driver. We are using the TinyTrak4. The TinyTrak 4 has been successfully used for satellite communication before [3]. The University of Hawaii at Mānoa has a paper on this subject. The TinyTrak4 will then control the satellite radio. The TinyTrak uses 2 analog inputs.

**I2C**

Inter-Integrated circuit (I2C) is a two wire standard for communicating to devices. We chose it because it is easy to implement, uses only two wires, and many devices on the market already support it. The protocol works by having multiple addressed devices on the same bus. A master will control the clock line and send commands through the data line. When a command is sent to a certain device, it will include that device’s address. Once a command is sent to a device, it can take control of the data line and the master will read the input. This protocol allows up to 255 devices to be on one bus. We are using this because more devices can be added easily in the future if we decide to do so. Currently we are planning on having only a temperature sensor, voltmeter on the MRR, and the charging circuit connected to the I2C line. These will be used mostly for telemetry and calibration reasons. We have room to add more voltmeters on the line as well so that the satellite can have more information about itself.

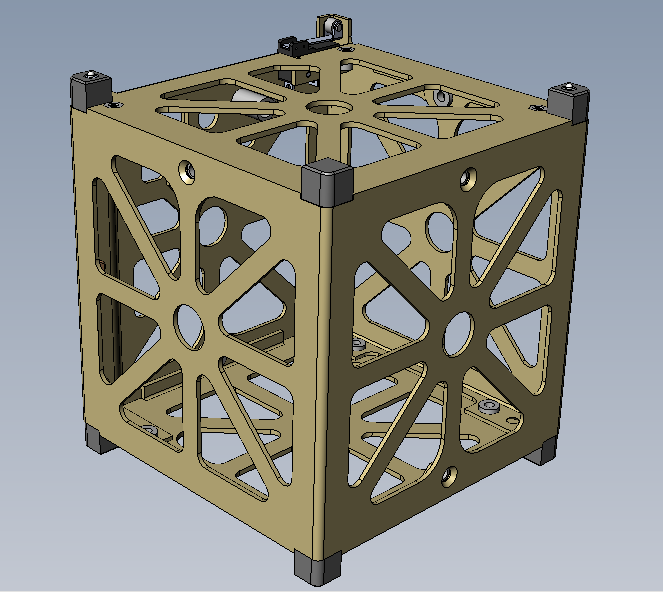
**Digital inputs**

Some devices will be directly connected to the SAM3X8E digital inputs. As mentioned before, some enables will be connected for the magnetometers. The TinyTrak4 controller will also have an enable for the radio. The DAC will be connected to the digital inputs also. The main device, the MRR, will be connected to the digital inputs to control the modulation of the device. The SAM3X8E has 50 digital inputs so we also have room to expand if we decide to add more devices.

**Function**

The CPU will be constantly checking the magnetometers and storing the data. When it has to transmit data it will use the temperature sensor to calibrate the MRR and will begin transmitting. If there are problems with the MRR, we have a backup radio that the processor will enable and use to communicate. The I2C bus will also control the charger circuit and a voltmeter on the MRR to collect telemetry data. This telemetry data will be used to calibrate the satellite, but can also be sent back to earth.

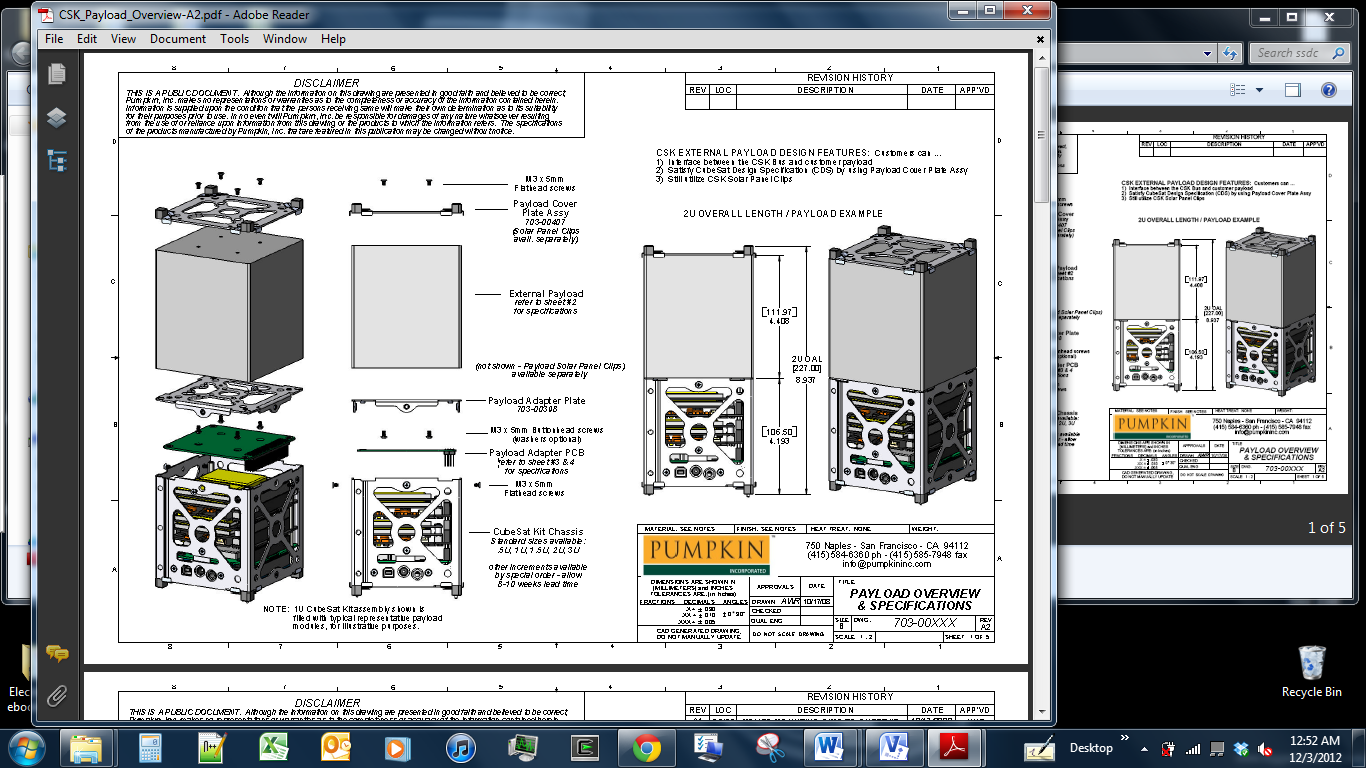
**Structures and mechanisms**



We will be using a frame made from Pumpkin Inc.[4] that was made for 1U satellites so it conforms to the limits of this competition. This satellite has 97.46 mm by 97.46 mm by 90.70 mm inner dimensions for our components and weighs 163 grams. We are using a ready-made frame to save time since it is already designed. We will attach four non deployable solar panels to the sides of the frame and have a stabilization coil on one of the sides.

**Alternatives**

While convenient, the Pumpkin frame does not have a hole for our MRR to be mounted. Since Pumpkin provides CAD models of all their products, we could modify one panel and add a hole for our MRR. We could even include a mounting bracket built in to that custom panel.

**Cover**

Pumpkin Inc. also sells a cover for this frame so that we can protect our satellite. This is one option we are considering. Although convenient, this has some drawbacks. The main one is that we would have to drill a hole for our MRR to be visible. This frame also adds more weight than carbon fiber would. A previous UF Funsat satellite has used carbon fiber frames (SWAMPSAT III, FUNSAT VI 2009). This is a cheaper alternative that we are most likely going to use.

**Alternative**

The CAD drawing from Pumpkin has a volume of 59966 mm3[5]. We found that the density of carbon fiber is 83 lbs/ft3[6]. Using this we find that:

Our cover, assuming a similar design to that of Pumpkin Inc.’s, should weigh 76.6 grams. Considering we will cut a hole out for the MRR, 76.6 is an over estimate of our weight.

The total estimated weight of our frame and cover is 239.6 grams with this option.

[1] ATMEL, “AT91SAM ARM-Based Flash MCU, SAM3X SAM3A series.” 28 May 2012. [Online]. Available:<http://www.atmel.com/Images/doc11057.pdf>. Accessed: [2 Dec. 2012].

[2] Arduino, “Arduino Due.” 23 Nov. 2012. [Online]. Available: <http://arduino.cc/en/Main/ArduinoBoardDue>. Accessed [26 Nov. 2012].

[3] Dugay, Dennis D., “Kumo A’O Cubesat Telecommunication Subsystem Integration and Testing.” [Online]. Available: <http://www.spacegrant.hawaii.edu/reports/18_FA07/DDugay_FA07.pdf> Accessed: 2 [Dec 2012].

[4] CubeSat Kit, “Begin your CubeSat Mission with the CubeSat Kit.” [Online]. Available: <http://www.cubesatkit.com/> Accessed: [2 Dec 2012].

[5] CubeSat Kit. “3D CAD design.” [Online]. Available: <http://www.cubesatkit.com/files/CAD/csk_1u_chassis_solid_revd.zip> Accessed:[2 Dec 2012].

[6] “Tube Properties.” [Online]. Available: <http://www.carbonfibertubeshop.com/tube%20properties.html> Accessed: [2 Dec 2012].