**1 - I2C Sensor Reads & SD Card Write**

Sensors:

* Accelerometer
* Compass
* Gyroscope
* (Altimeter)
  + Code commented out due to unresolved issues
* Motion sensor
  + Simple digitalPin read, not I2C

Shields:

* SD Card

Notes:

Program works fine when hooked up to hardware serial (computer terminal output). When battery used (7.4V), gyroscope data not acquired (0, 0, 0), or program simply fails at header write. I think a 12V input source would ameliorate this issue.

**2 - Camera (JPG acquisition) & SD Card Write**

Sensors:

* Camera board
  + Also has motion sensing capability

Shields:

* SD Card

**3 – Receive SMS Using GPRS Shield**

Sensors:

* None

Shields

* GPRS

Notes:

Because Arduino Uno only allows one hardware serial I/O to communicate at a time, GPRS shield communicates with Arduino using SoftwareSerial library.

It should be noted that SoftwareSerial.h has a built-in buffer size of 64 bytes, which I modified to 256 bytes without a significant impact on compiled program size. This had to be done, because given the size of the message header, etc, messages were always truncated before even reaching the content body.

This program illustrates one of two strategies for controlling the GPRS. Here, we setup the GPRS software serial to read output of the Arduino hardware serial. In this way, we can write AT commands in plain text to the Arduino hardware serial and the GPRS will capture everything that is printed and process them as they would be if one were manually inputting AT commands into a terminal window.

The second strategy involved the use of the built-in GSM.h library within Arduino. It shortens commands by defining all sorts of functions, but I don’t think many of them will be extremely useful for our purposes. This library was designed for use with a different GPRS shield, so to make it work, it will require a lot of backend manipulation of the library files. I’ve tried it with some success, but the first strategy just seems much easier to me.

**4 – LCD Display of Plain Text Written in Serial Monitor**

Sensors:

* None

Shields:

* LCD Display

Notes:

LCD shield requires use of digital pins 4, 5, 6, 7, 8, 9 for input and overlaps with GPRS shield when stacked in the traditional configuration. One way around this will be to mount one of the two shields on a PCB and manually route the pins to other available digital pins on the Arduino. In practice, this will likely need to be done anyways, as the LCD will need to be positioned in the prototype enclosure in a manner that allows viewing and interaction by the user. This position likely wouldn’t be optimal for the other shields/components if they were all stacked together.