Hey Mikal,

First off, thanks for putting together this library. I've been using it to develop some low-cost oceanographic buoys and it has been absolutely indispensable.

However, I did run into some problems using the IridiumSBD library/RockBLOCK with Software Serial. I was having an issue where messages would send successfully some of the time, but other times it would return ISBD\_PROTOCOL\_ERROR and not send.

After issuing an isbd.sendReceiveSBDText command, I would often see something like this in the terminal window\*:

>>AT+SWB=Hello!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

<<AT+SWB=HeŠŠŠŠŠŠŠŠŠŠŠµŠŠŠŠŠŠŠŠŠŠŠŠŠæŠŠŠŠŠŠŠŠŠŠŠŠŠØž¦!!!!!!!!!!!!!

(\*ASCII characters aren't exact, so please don't try to dig into the 1's and 0's of bit transmission. The point is that the response sort of looks like the message, but has errors.)

After some analysis, it turns out the commands were being issued to the RockBLOCK perfectly fine, but the code would get hung up when the RockBLOCK was responding and the Arduino/SBD library was listening for the AT Response. The Arduino/SBD library would frequently misread bits, but often times 'catch up' by the end of the message and was still able to parse the end of line character, allowing the code to continue. Other times, if the last character was one of the ones misread, it would just hang up and not send the command. (This was really a bugger because IF the message sent at all, the actual transmitted message was always correct with no errors since the RockBLOCK was always seeing the command properly. It was just whether the Arduino/SBD library was able to see the AT Response properly or not.)

Turns out there is a fairly well known problem of software serial not being as reliable as hardware serial because of the time delays and background processing for software serial (makes sense now), but it did cause me some consternation. (See the AltSoftwareSerial Library webpage for more specifics on the reliability issue.)

I have been using an Arduino MEGA, so I was able to easily solve the problem by switching the RockBLOCK's RX and TX lines to Serial1 (hardware serial) lines on the MEGA.

There may be a way to rework the loop that parses the AT response similar to what @Plumb did or build in something so that bits are not lost in the software serial transmission, but it is beyond my powers as a programmer. Since that is the only spot in the code it seems to have issues with software serial, I presume it can be done. Until that problem can be resolved, I’d recommend anyone using the library to use hardware serial pins. It would be nice in the long term to have a patch in the library, so hardware serial isn’t necessary though.

Hopefully that input is helpful for development. Thanks again for all your efforts.

DanL