1 Get idea

2 Research on for to achieve goal

3 Research about components

4 Research about components

5 Research about components and circuit

6 Start creating circuit diagram

7 Creating further circuit diagrams

8 Research about where to purchase

9 Additional research about where to purchase

10 Obtain quote from Communica

* Wait
* Wait
* Wait

14 Obtain Quote

20ste – Receive parts from Communica. Test GPS – Success!

Test GSM – Boot running… Suspect insufficient power, recommended is 4v

21ste – Looking for buck converter online as well as multimeter

23ste – Make lipo battery as power source from old cellphone battery in order to eliminate buck converter

Test GSM – success

28ste – get idea to move to modular design which will pen GPS to be used for other aplications

30ste – Start paper work

Write engineering goals and categories on paper

Beginning research from sources gathered

6/02:

Write code to reply on certain SMS messages

Did to much testing ran out of airtime on simcard!

Created code to respond to specific sms’s like ‘help’ and ‘location’

----- explain--------

7/2

Tried adding GPS info to code. Ended up with error. Adafruit gps and Gsm both use same library causing errors (software serial) while compiling. Switched to TinyGPS++ still got errors. Finally found out That I needed AltSoftSerial as I can’t use 2 instances of software serial.

After much trial and error I have gotten the device to output the location

10/02

Revised the code to send a link to google maps if the command ‘location’ is sent to the device:

void sendGPS(){

sms.beginSMS(senderNumber);

sms.print("Location: "); //Sends location

sms.print(gps.location.lat(), 6);

sms.print(", ");

sms.print(gps.location.lng(), 6);

sms.print("\nGoogle maps: "); //Sends link to Google Maps

sms.print("https://www.google.com/maps/place/");

sms.print(gps.location.lat(), 6);

sms.print(",");

sms.print(gps.location.lng(), 6);

sms.print("/");

sms.endSMS();

}

Wrote the status and OFF commands’ code but the first try failed, I modified the code and it should work but the first try of the code spammed a bunch of messages and used all the airtime on the simcard. I wil have to buy more tomorrow. Here is the code:

else if(msg.indexOf('STATUS') >= 0){

sendStat();

}

else if(msg.indexOf('OFF') >= 0){

gpsPower = False;

digitalWrite(gpsVIN, LOW);

sms.beginSMS(senderNumber);

sms.print("GPS is now off!");

sms.endSMS();

}

And:

void sendStat(){

sms.beginSMS(senderNumber);

sms.print("Bike Tracker Status: ");

if(gpsPower){

sms.print("ON");

sms.print("\nSatellites: ");

sms.print(gps.satellites.value());

}else{

sms.print("OFF, send command 'on' to reboot the gps");

}

sms.endSMS();

}

03/03

Brought new airtime and tested the code written on 10th Feb. It did not work as the microcontroller ran out of space for variables. I shortened all the sentences to preserve space, but the code would not work as expected and continuously sent SMS’s. I thus decided to rewrite the SMS commands, removing the status and adding the amount of satellites in the location SMS. I reprogramed the ON/OFF feature of the GPS and made it so that when of is send the GPS is turned off and when on is sent it turns back on. Unfortunately, this lead to stability issues as the microcontroller once again ran out of space. This forced me to reprogram this function again. Sending off still revokes power from the GPS but sending location will return power to the GPS module. This code turned out great as sending location after the GPS was switched off would send the previously know location and will update the data once it has acquired 3 or more satellites. The complete updated code on the next page.

I also worked on the case for the final PCB. By making adjustments to the design. This idea sketches are also on the next pages.

05/03

Made final tweaks and simplifications to the software and think that it is about time to make a new hardware prototype that gets rid of the Arduino and opts for a atmega 328P-PU without an external clock to preserve power and make the device ready for testing and designing the third and final prototype, the self-contained PCB.

Also need to purchase more airtime as while testing I ran out of it. Do not think that running out of airtime will be a problem in the final build as users will not query the device several dozen times a day.

08/03

Tried testing the GPS inside a car but ran into errors in the software after the device was sent an advert by Vodacom. I noticed that this would be a possible problem in a real-world scenario, so I reprogrammed the device to only respond to SMSs starting with an ‘#’ symbol. Now the device can determine which SMSs are commands and which are not and only reply to ones that are commands. This is hopefully the last alterations for this prototype. Next, I will start working on the next prototype. For this I will need a new apparatus: a multi-meter, to measure the amount of amp draw. This apparatus is needed to determine the current draw of the device, so that I can calculate how long the battery will last. The goal of the next prototype as stated in a previous entry is to minimize the power consumption of the device and get rid of the Arduino in favor of an ATMEGA 328P-PU IC.