

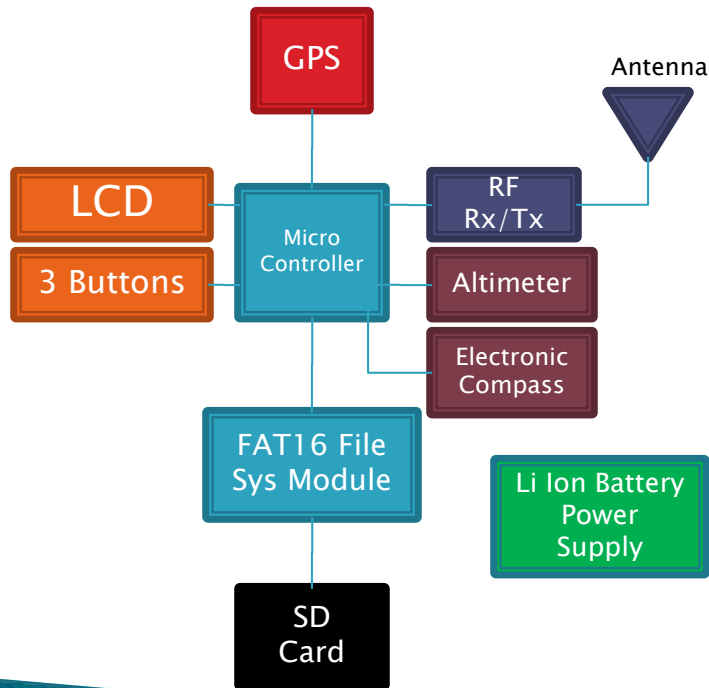
Tandem GPS System with Remote User Tracking

Brad Leppert
Eric Hudson

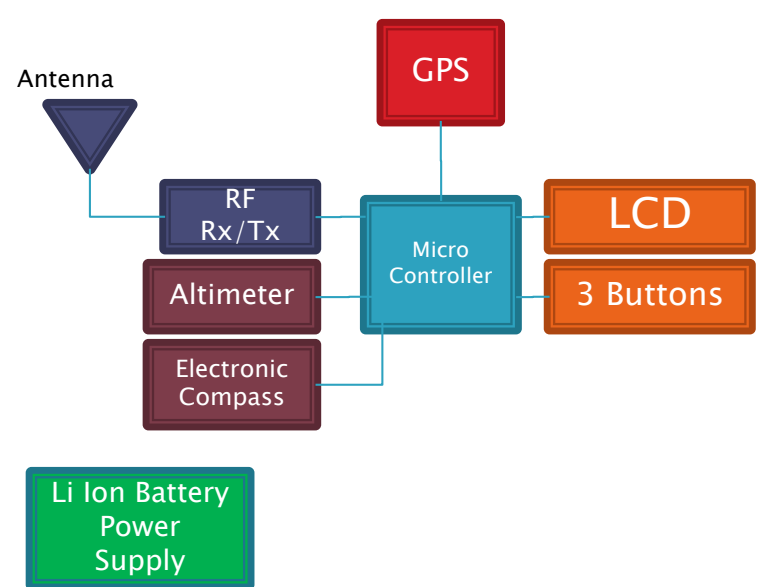


Block Diagram

Unit 1



Unit 2



Major Parts

- **Micro Controller:** 2x Atmel Atmega644PA

- **GPS:** 2x PCTEL 5012D-CE GPS modules



- **Radios:** 2x Free Wave MM2 radios



- **Electronic Compass:** 2x PNI V2XE Compass modules



- **Switch Debouncers:** 4x MAXIM 6817 2Ch switch debouncers

- **LCD:** 2x Sparkfun Parallel Graphic LCDs 128x64

- **SD Card Module:** 1x Comfile COM5 SD/MMC FAT16 file system module

- **Barometer:** 2x Intersema MS5534C Digital Pressure Sensors

Custom sentence

- ▶ FWR sentence: for transmission and easy parsing over the Free Wave Radios

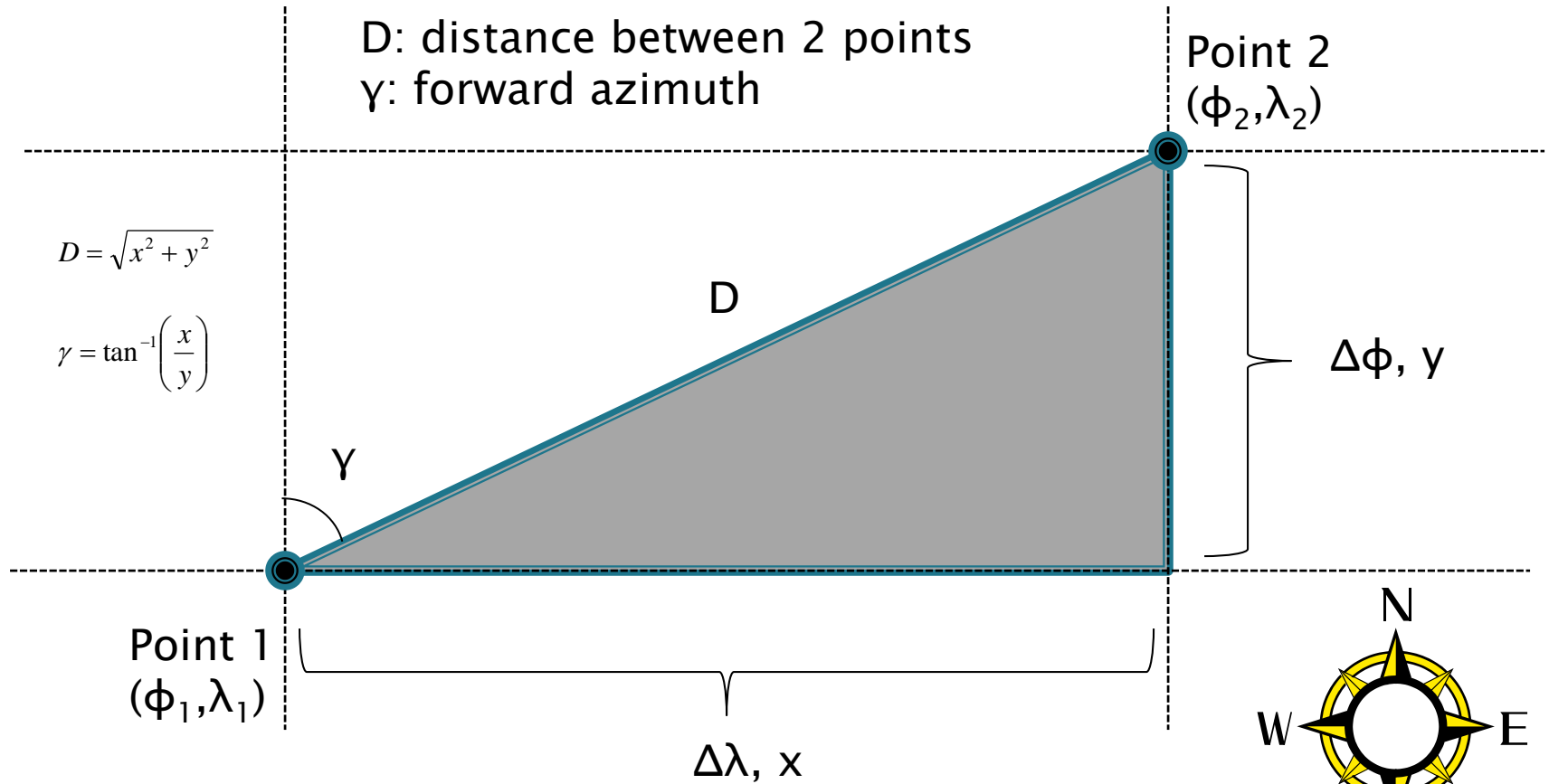
▶ *\$FWR, hhmmss.sss, dd mm.mmm H, ddd mm.mmm H, xxx.x, ddd.d, xxxxx, A,xx,xx <CR> <LF>*

▶ *1st char always \$, ¹Comma ²delimited ³data ⁴fields, ⁵ends w/ ⁶carriage ⁷return ⁸and ⁹line feed*

Data Fields:

- 1 = UTC of last position fix
- 2 = Latitude with spaces inserted and hemisphere included
- 3 = Longitude with spaces inserted and hemisphere included
- 4 = Speed Over Ground [mph]
- 5 = Track Angle [deg]
- 6 = Elevation [ft]
- 7 = Valid Lock
- 8 = Satellites in View
- 9 = Satellites in Use

Geographic Coordinates



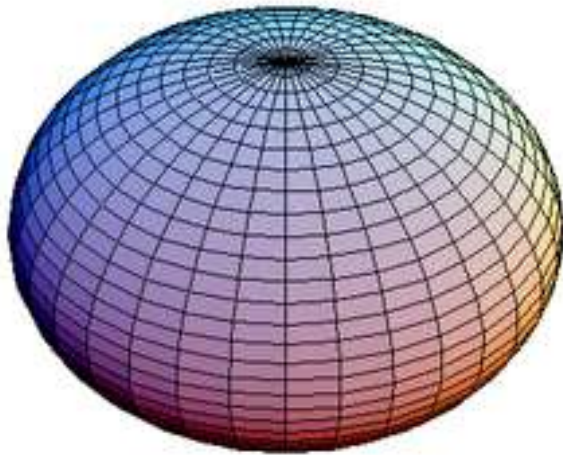
Have: $\Delta\lambda$ and $\Delta\phi$ [°]
Need: x and y [km]

Aside:

Spherical Form of Pythagorean Thm.

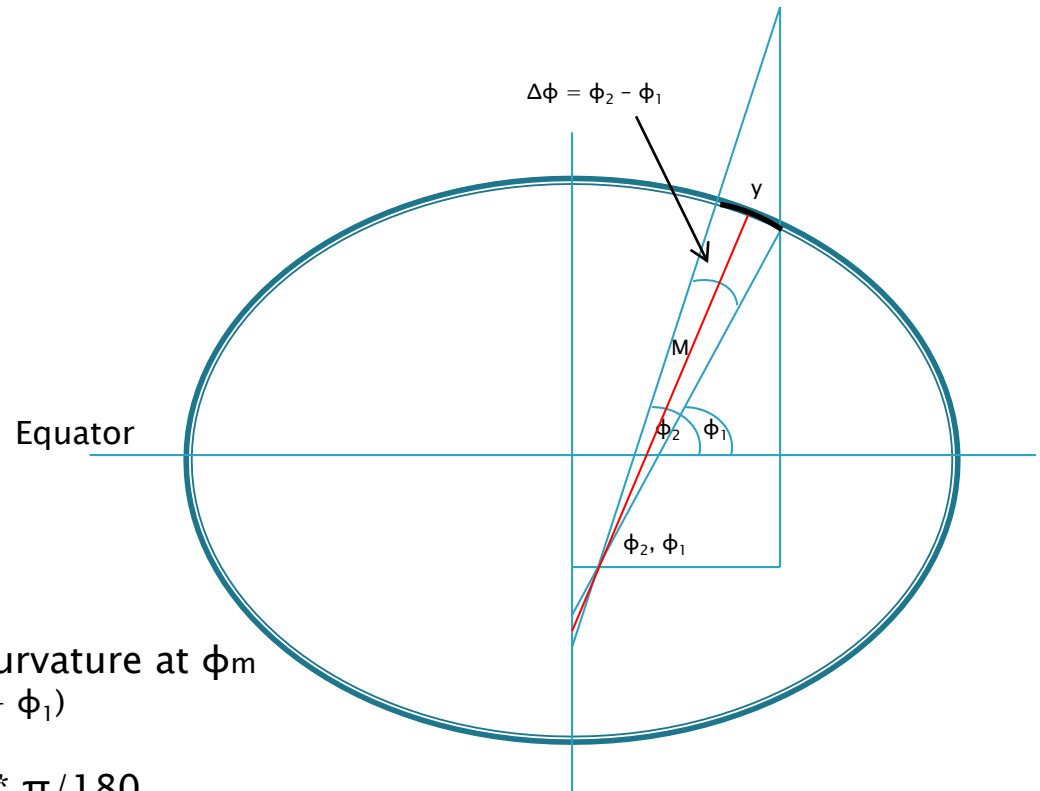
$$\cos\left(\frac{c}{R}\right) = \cos\left(\frac{a}{R}\right) \cos\left(\frac{b}{R}\right).$$

Geographic Coordinates



Oblate Spheroid

Elliptical Arc Length \approx Angle * Radius of Curvature



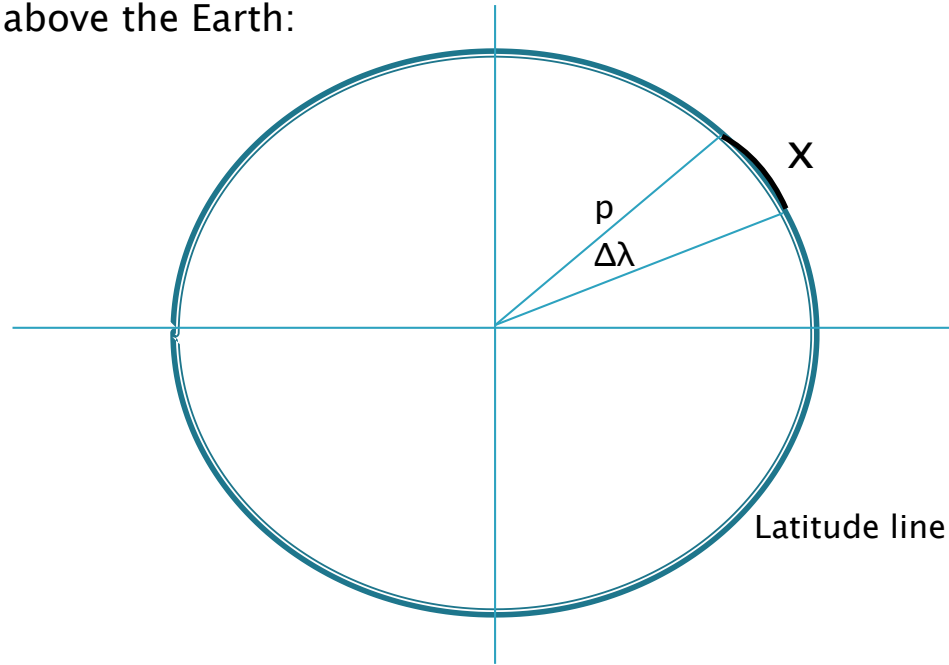
N: normal radius of curvature at ϕ_m

$$\phi_m = \frac{1}{2} (\phi_2 + \phi_1)$$

$$y \approx \Delta\phi * M(\phi_m) * \pi/180$$

Geographic Coordinates

Looking down from above the Earth:



$$x \approx \Delta\lambda * p$$

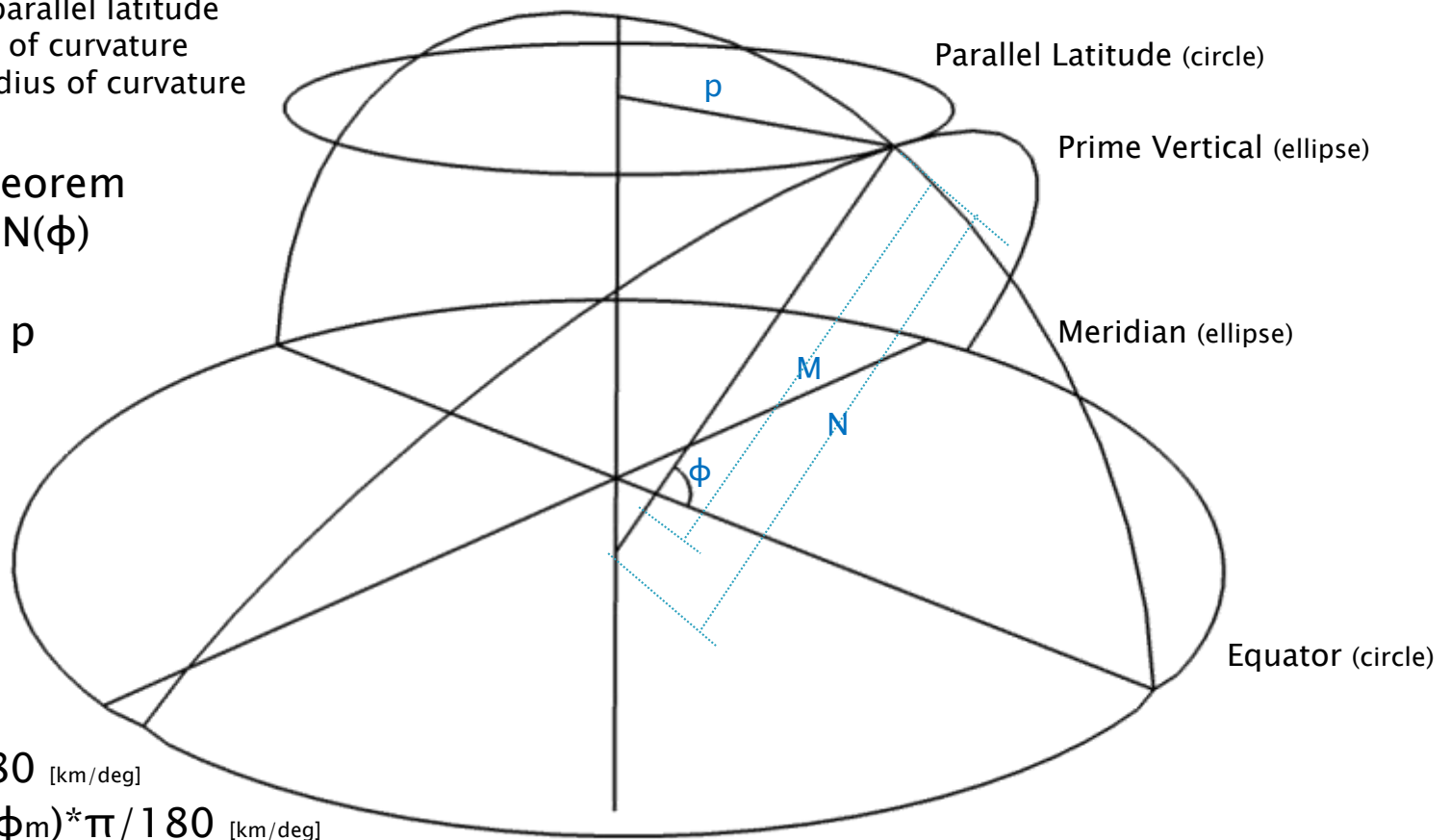
Geographic Coordinates

p : radius of the parallel latitude
 N : normal radius of curvature
 M : meridional radius of curvature

Meusnier's theorem

$$p = \cos(\phi) * N(\phi)$$

$$x \approx \Delta\lambda * p$$



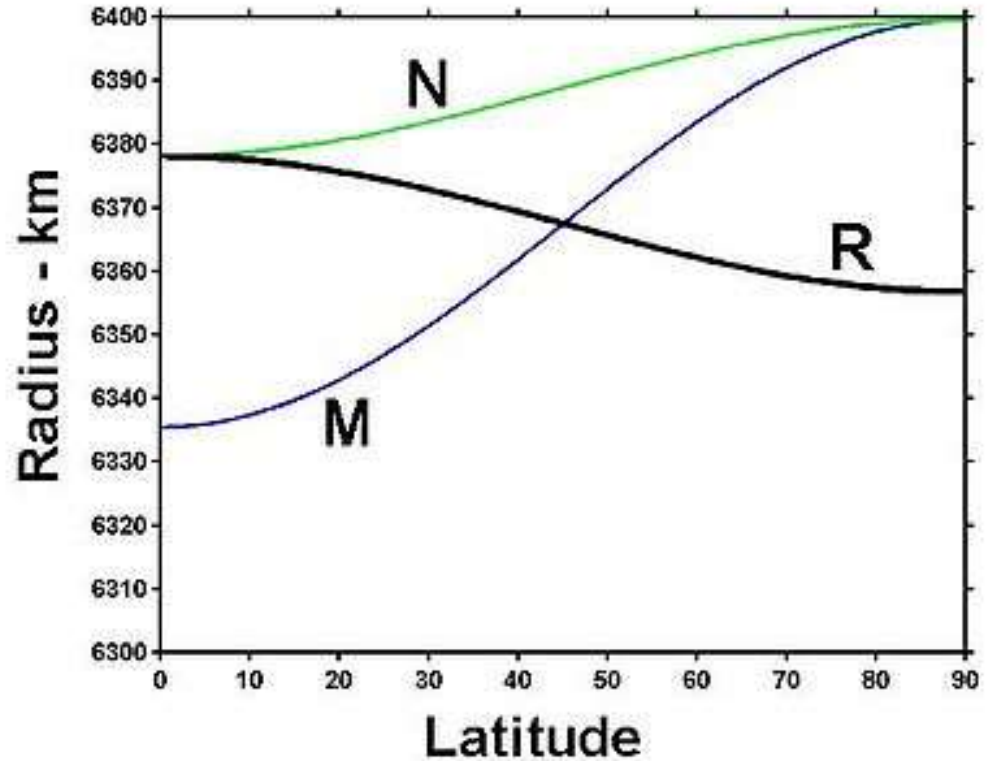
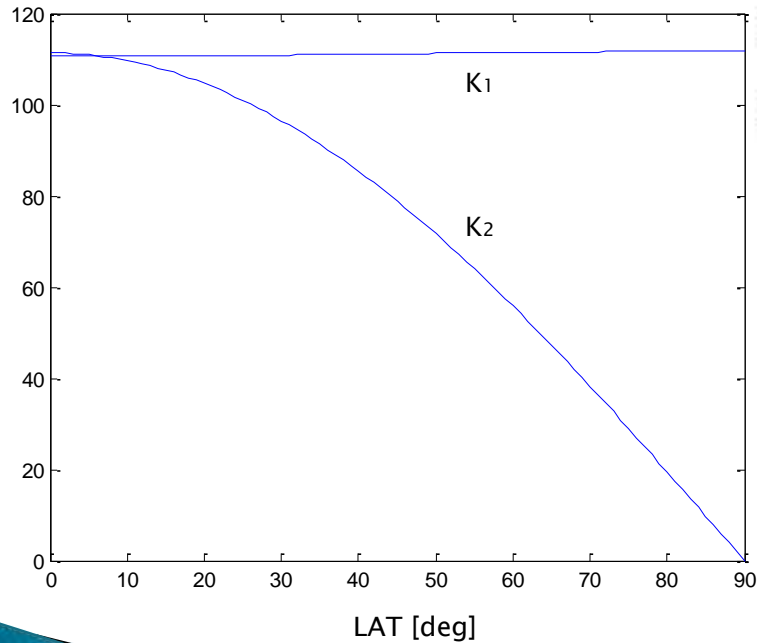
$$K_1 = M(\phi_m) * \pi / 180 \text{ [km/deg]}$$

$$K_2 = p = \cos(\phi) N(\phi_m) * \pi / 180 \text{ [km/deg]}$$

Geographic Coordinates

N: normal radius of curvature
M: meridional radius of curvature
R: Earth's radius

K₁ & K₂ vs LAT



Geographical Equations

▶ Geographic Distance between 2 points:

- Ellipsoidal Earth projected to a plane
- used by FCC for distances less than 475km

$$D = \sqrt{(K_1 \Delta\phi)^2 + (K_2 \Delta\lambda)^2} \text{ [km]}$$

$$\Delta\phi = \phi_2 - \phi_1 \text{ [Latitude degrees]}$$

$$\Delta\lambda = \lambda_1 - \lambda_2 \text{ [Longitude degrees]}$$

$$\phi_m = \frac{\phi_1 + \phi_2}{2} \text{ [degrees] (convert to radians for COSINE)}$$

$$K_1 = 111.13209 - 0.56605 \cos(2\phi_m) + 0.00120 \cos(4\phi_m) \text{ [km/degree LAT difference]}$$

$$K_2 = 111.41513 \cos(\phi_m) - 0.09455 \cos(3\phi_m) + 0.00012 \cos(5\phi_m) \text{ [km/degree LONG difference]}$$

- Google Earth: 2.52km Eq: 2.526km (PMBC1 to CROW)

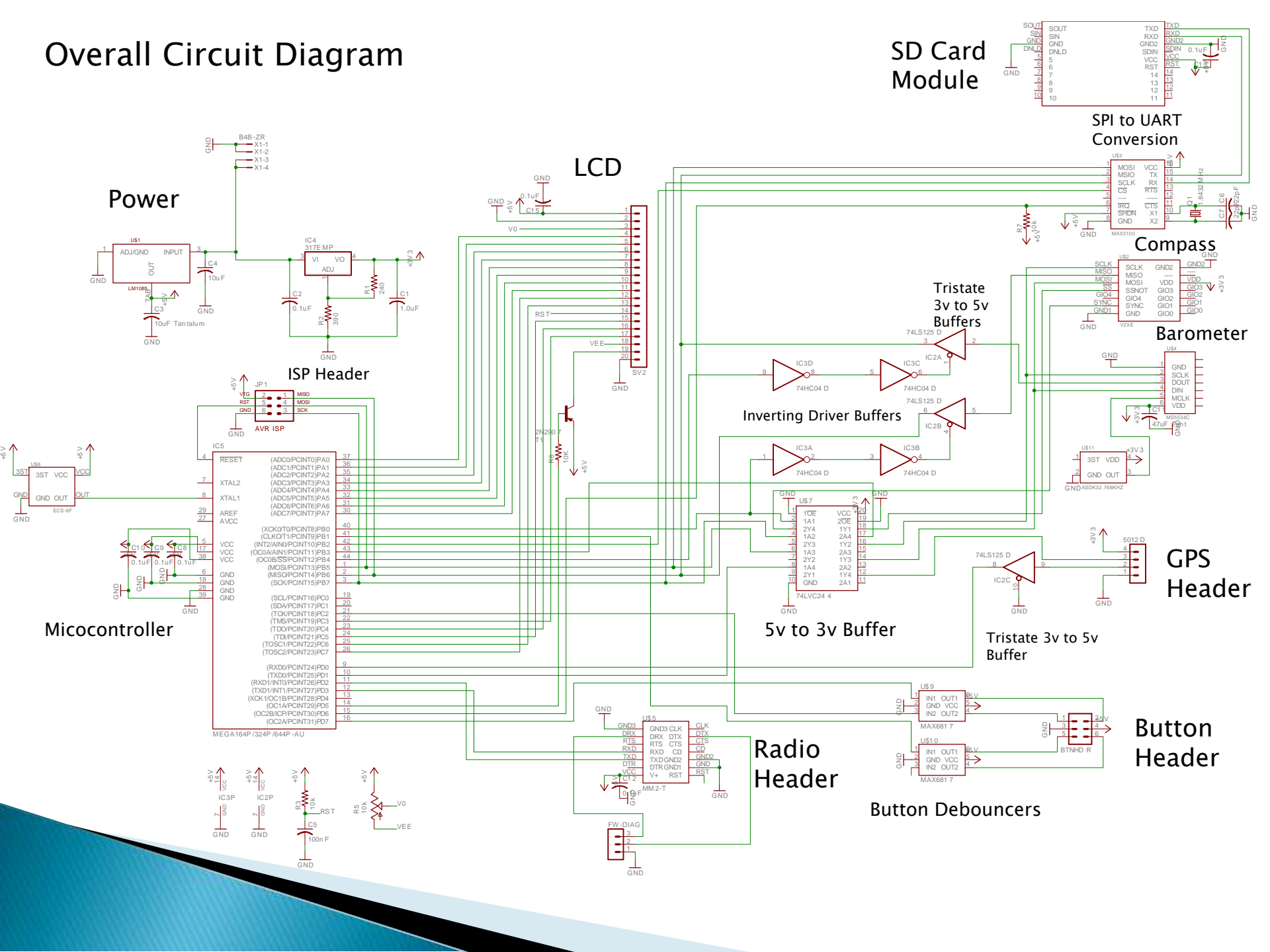
Geographical Equations

▶ Azimuth Between 2 Geographic Points

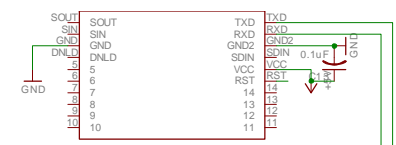
$$\gamma \approx \tan^{-1}\left(\frac{K_2\Delta\lambda}{K_1\Delta\phi}\right) * \frac{180}{\pi} [\text{deg}]$$

- Will use *atan2* function from *math.h* in winAVR compiler then use *fmod((γ +360),360)* *floating point modulus* to map the result from -180° to 180° ➡ 0° to 360°
- Google Earth: **150.39** Eq: **150.2970**
- AVR Studio Simulation: ~78500 Clock Cycles (~5.6ms @ 14MHz) for calculating D and γ .

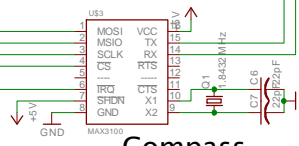
Overall Circuit Diagram



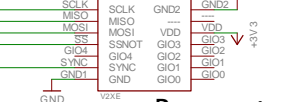
SD Card Module



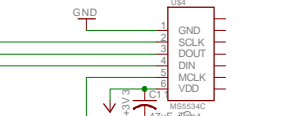
SPI to UART Conversion



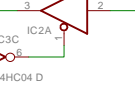
Compass



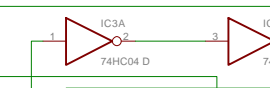
Barometer



Tristate 3v to 5v Buffers



Inverting Driver Buffers



5v to 3v Buffer



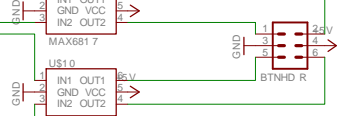
Tristate 3v to 5v Buffer



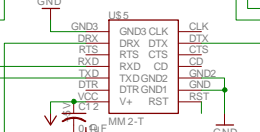
GPS Header



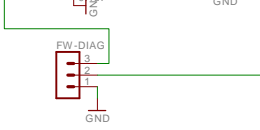
Button Header



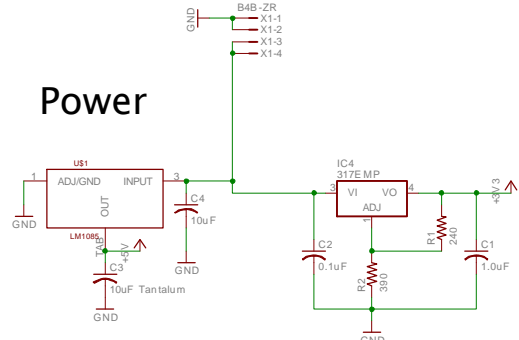
Radio Header



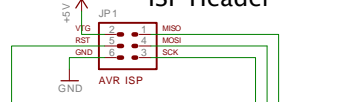
Button Debouncers



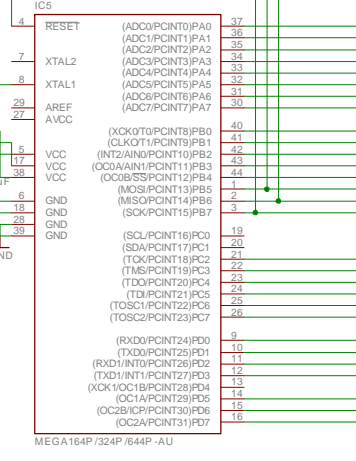
Power



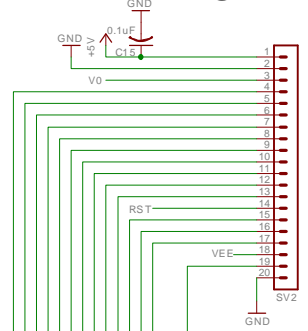
ISP Header



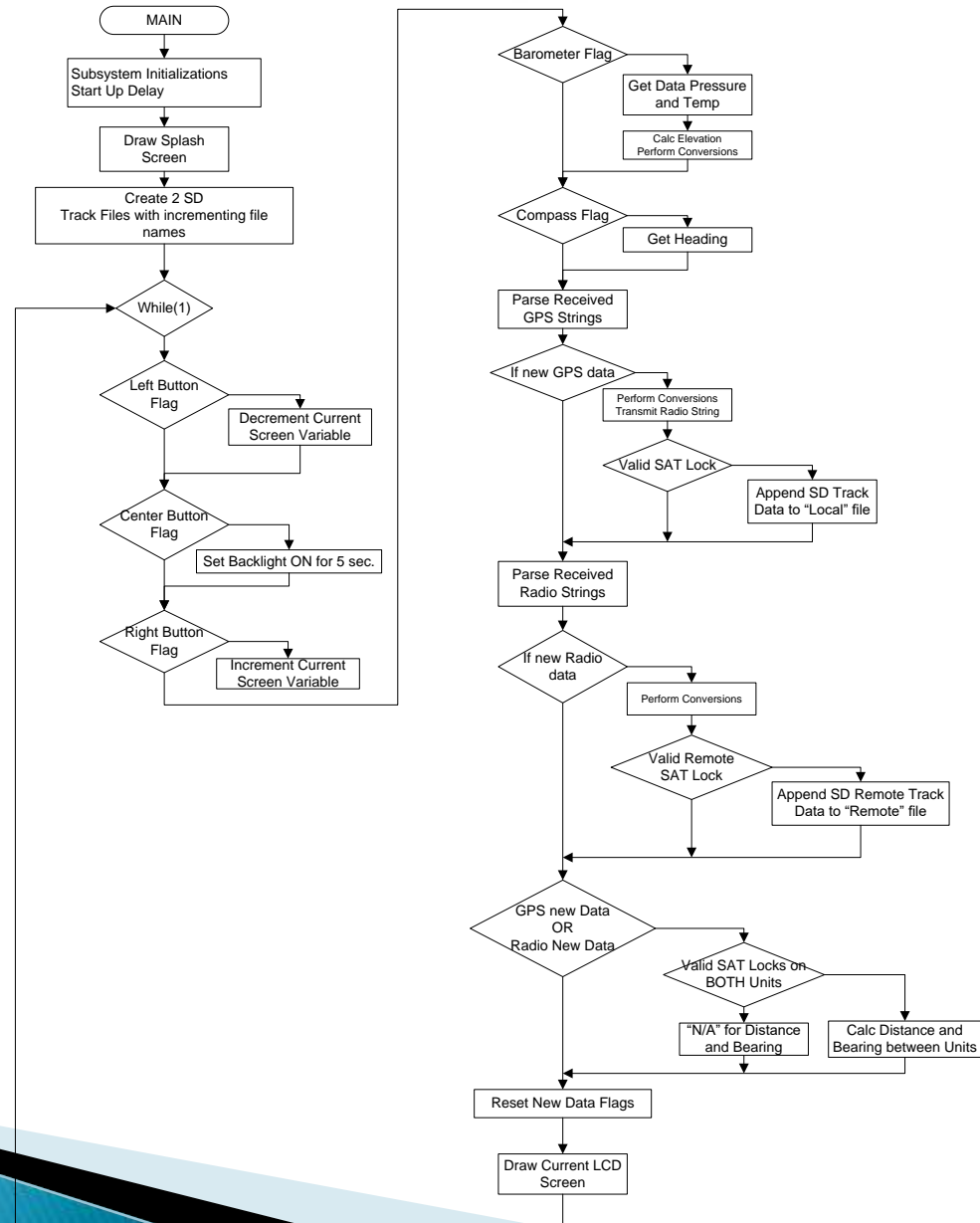
Microcontroller



LCD



UML Activity Diagram for MAIN



LCD Data Screens

```
** SAT LOCKS **  
LOCK                :: YES  
SATS IN USE        :: 05  
SATS IN VIEW       :: 11  
** REMOTE UNIT **  
LOCK                :: YES  
SATS IN USE        :: 06  
SATS IN VIEW       :: 10
```


LCD Data Screens



The image shows a green LCD screen with a grid of data. The labels on the left are TIME, LAT, LONG, ELEV, SPEED, TRACK, and TEMP. The data values are as follows:

Label	Value
TIME	235201
LAT	41 18.875 N
LONG	105 24.897 E
ELEV	204.5 [ft]
SPEED	0.0 [kts]
TRACK	105 [deg]
TEMP	72 [F]

LCD Data Screens



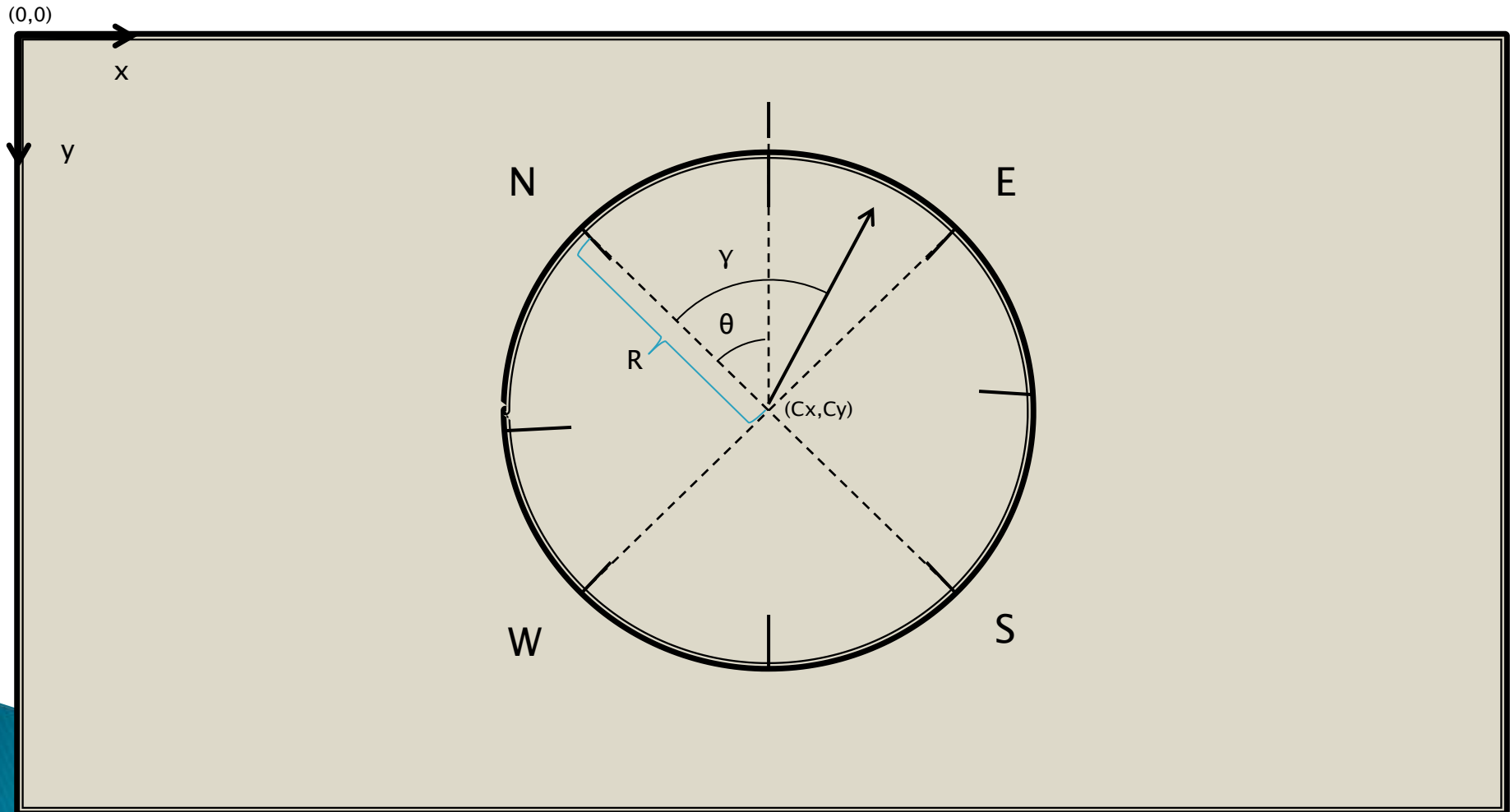
LCD Compass Geometry

θ : heading

γ : bearing to other unit

$$x = -r \sin(\theta - \gamma) + C_x$$

$$y = -r \cos(\theta - \gamma) + C_y$$



LCD Data Screens

```
REMOTE  UNIT  DATA
LAT      :: 41  18.867  N
LONG     :: 105 34.873  W
DIST     :: 33.35  [mi]
BEARING  :: 113  [deg]
ELEV     :: 7557  [ft]
TRACK    :: 92.7  [deg]
```



Power Usage

Current Draw

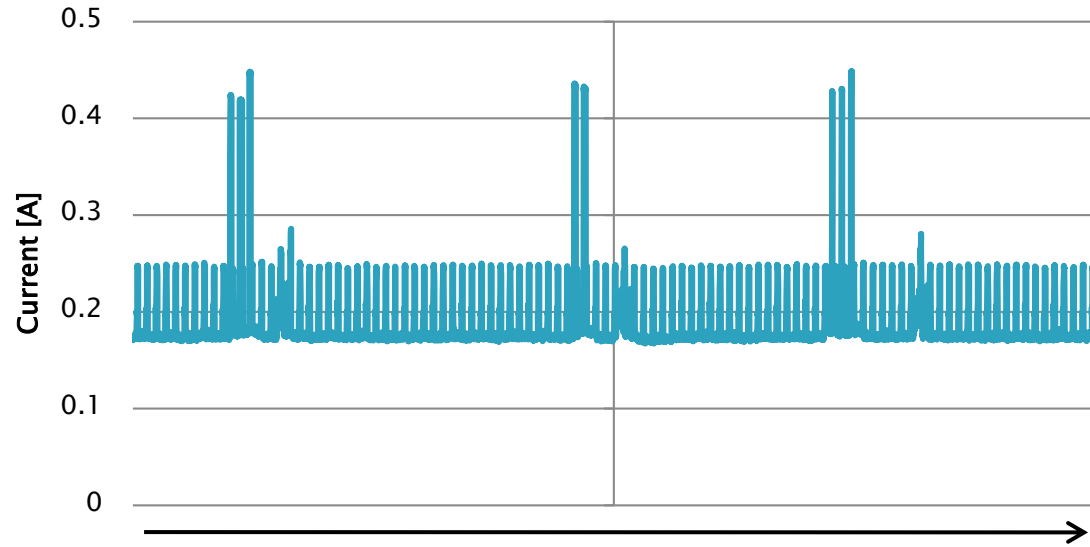
Average: 188 [mA]
RMS: 191 [mA]
Peak: 449 [mA]

Battery Life

2200mAh 7.4v Li-ion pack
->11.5 Hours

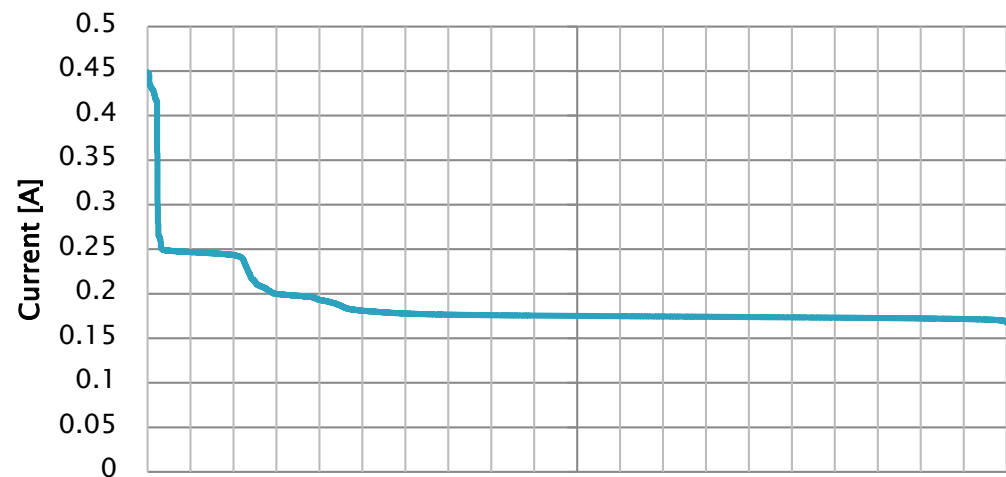


Current Draw



2 seconds elapsed

Sorted Current Samples



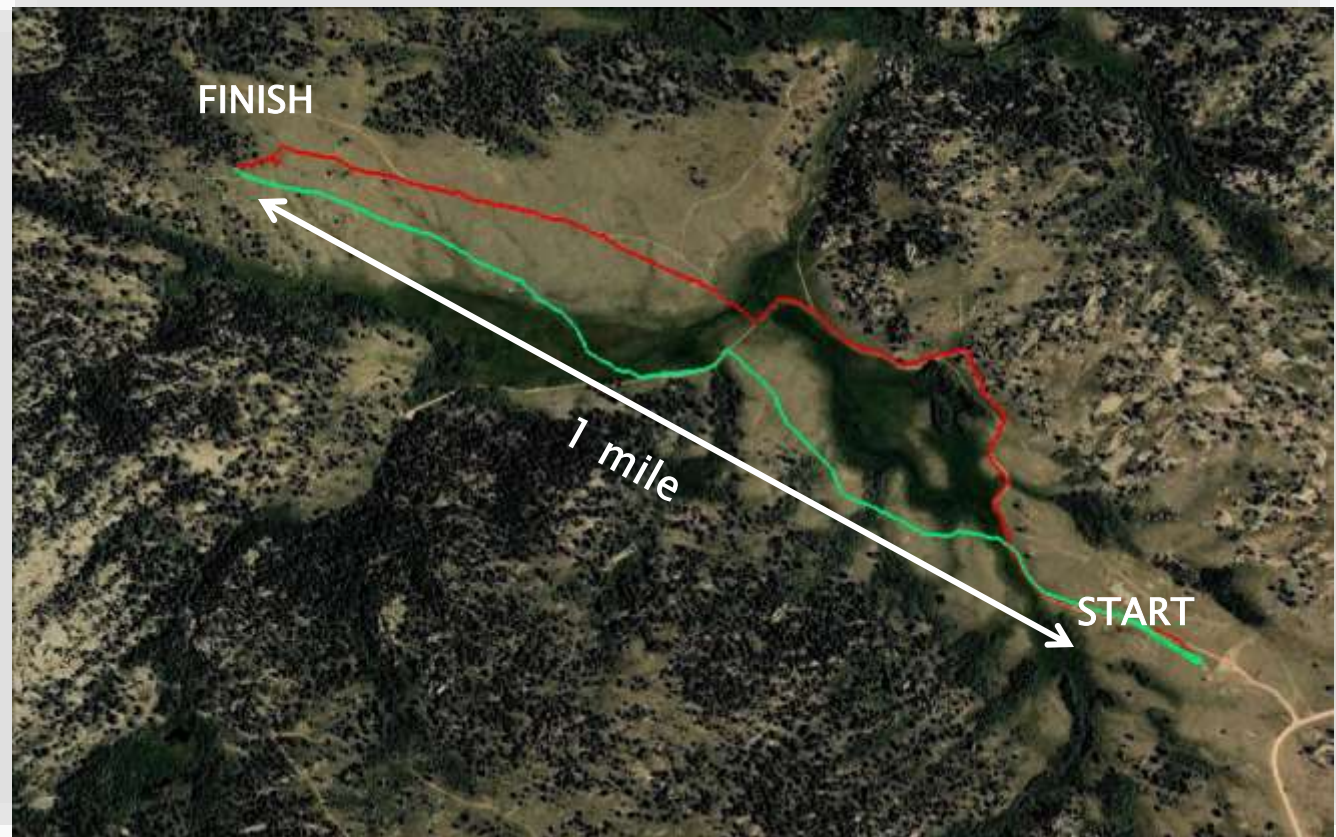
0.1 sec. per division

Field Test Results

Vedauwoo

Brad (red track) hiked out to a 1600m distance from Eric and hid behind a large rock and some trees.

Eric (green track) then successfully found Brad with only the aid of the GPS unit.



Questions

?

