GPS + Inertial Sensor Fusion

Senior Project Proposal

Aleksey Lykov, William Tarpley, Anton Volkov

Advisors:
Dr. In Soo Ahn, Dr. Yufeng Lu

Date:
November 26, 2013

Project Summary

The objective of this project is to combine inertial and GPS sensors via I2C on the Raspberry Pi single board computer for the purpose of maintaining accurate positioning in areas without GPS or in the event of losing a GPS signal. This will be accomplished with heavy post processing, including Kalman filtering of the inertial measurements (accelerometer, gyroscope data) combined with drift reduction using magnetometer data, and finally through sensor fusion with GPS data.

Detailed Project Description

An inertial navigation system (INS) or inertial measurement unit (IMU) is a form of dead reckoning navigation system that uses a combination of accelerometer and gyroscope sensors working in concert to detect displacement relative to a starting point. The system measures both linear accelerations given by its three-axis accelerometer and angular velocity changes from the gyroscope measurements. World referenced-frame acceleration data can then be integrated to calculate the velocity and position of the sensors over time, but because the INS can only measure motion in relation to a starting location, the initial position must be supplied by some outside system (in the case of this project, a Global Position Sensor). Additionally, to compensate for the drift in the inertial navigation system caused by various defects in inertial sensors, this outside reference (GPS) must be polled occasionally to correct for the position error.

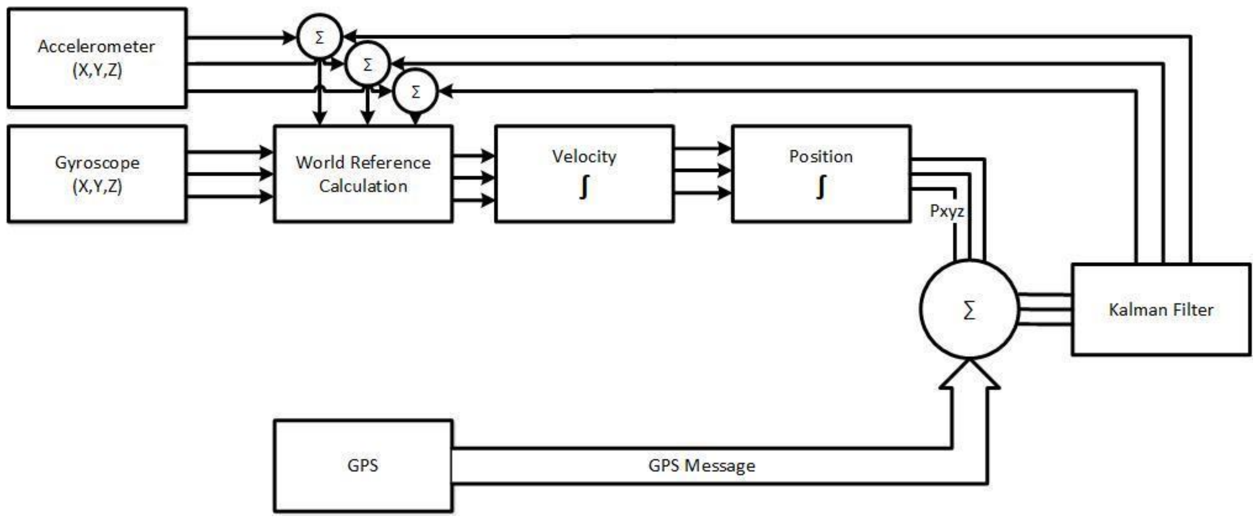
Using a Raspberry Pi microcomputer as the base system and an MPU 9150 IMU, an inertial navigation system will be developed. Kalman Filtering and GPS will be used to complete a “strapdown solution” – a closed-loop system which can self-correct for error [1].

The main sensor used for the project is the MPU 9150 Inertial Measurement Unit (IMU). This chip sends out nine axes of data: x-acceleration, y-acceleration, z-acceleration, yaw, pitch, roll, and three axes dedicated to magnetometer data. The three-dimensional acceleration and gyroscope data is used for data acquisition. Additionally, the chip has the capability to output the quaternion as well as real-world adjusted data which combines the data from the gyroscope and accelerometer to get real-time accelerations for yaw, pitch and roll. If needed, the magnetometer can be used to help calibrate the gyroscope to compensate for some of the drift.

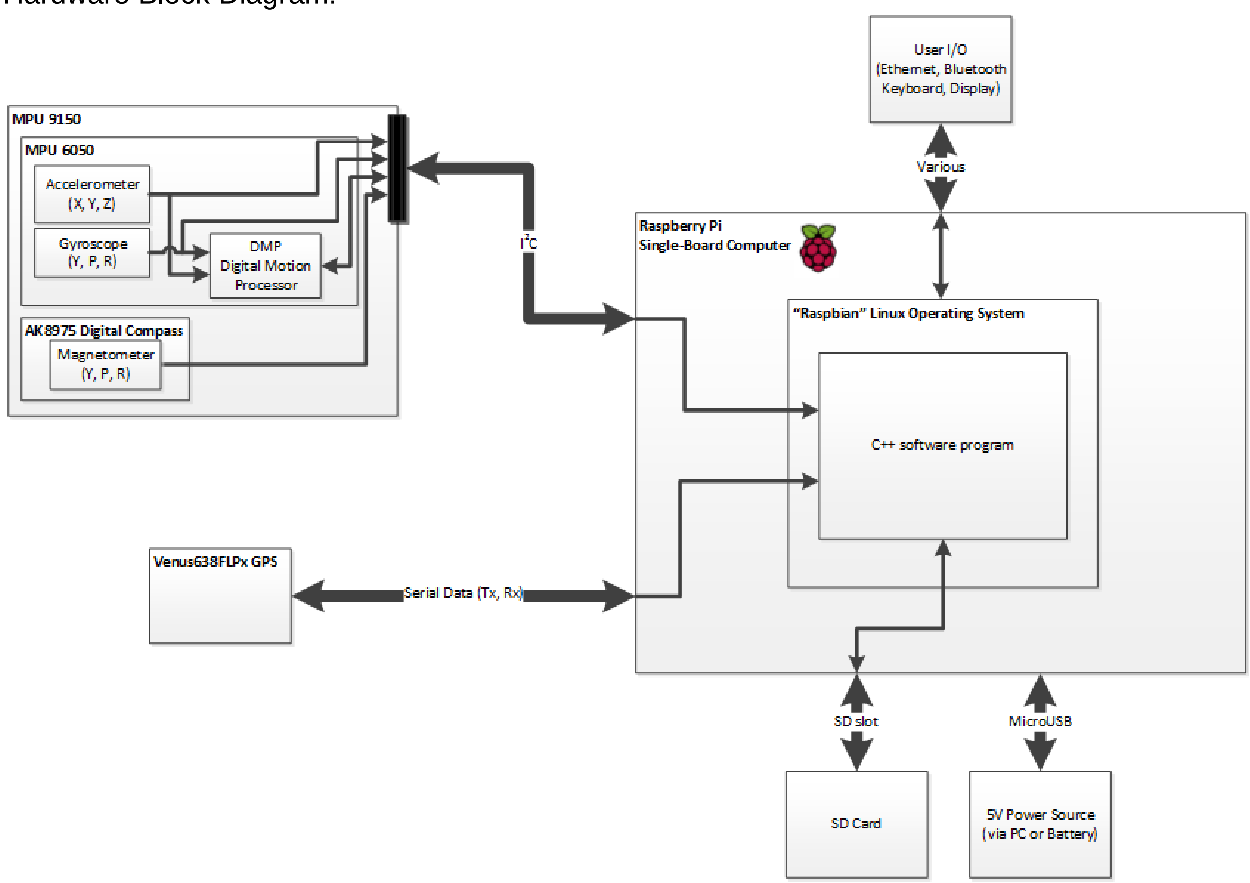
A GPS unit will be used to give the program its initial position and to correct for error through the operation of a Kalman Filter.

The heart of the system will be the Raspberry Pi microcomputer. It runs C or C++ code for data acquisition and data processing. The IMU and the GPS are semi-permanently attached to the I/O pins of the Pi, which read in data from both devices after which the data is time stamped and saved to SD card. Eventually, the Pi will also perform real time filtering of the data instead of just data acquisition in order to calculate current system position.

Signal Block Diagram:



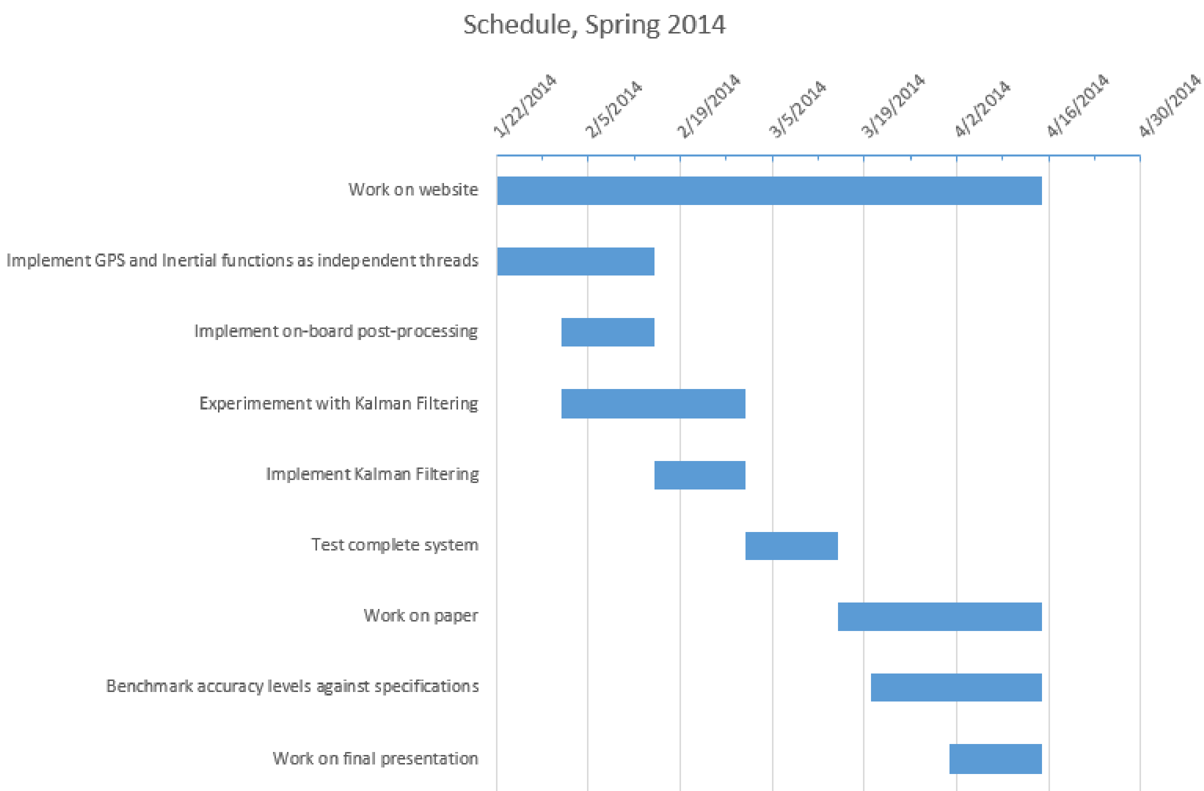
Hardware Block Diagram:



Functional Requirements for System Operation

- System shall be ready to start navigation 30 seconds after being started
  - 30 seconds to acquire GPS satellite position lock while outdoors
  - 20 seconds to simultaneously stabilize gyroscope
- System shall rely primarily on inertial measurements, using GPS external positioning input only to correct for drift [2]
- After one minute of operation without GPS data, the system shall report a position within either 5% or 1 meter of the true displacement, whichever is greater
  - If the user travels 100 meters in one minute, the system shall report between 95 and 105 meters total displacement
  - If the user travels 0 meters in one minute, the system shall report a distance “travelled” – a drift – of less than 1 meter
- While maintaining a GPS signal, the system will report a position with a degree of accuracy with less than 5% error (if 100 kilometers are travelled in one hour, the system will report between 95 kilometers and 105 kilometers travelled)
- If the system loses a steady GPS signal, it shall alert the user via the display interface
- The system shall initially save all acceleration and GPS data to text file, but the final product will be able to integrate into velocity and position data (and also correct for errors in real time)
- The system shall re-acquire a GPS signal if it is lost within 30 seconds of entering an unobstructed outdoor environment [5]
- The system shall operate continuously (without any crashes or lapses in data acquisition) on battery for a minimum of 2 hours
- Maximum acceleration: ±2g, ±4g, ±8g, or ±16g selectable by user [3]
- Maximum angular rate: ±250 degrees/second at max resolution [3]
- During operation, the system shall display the following information to the user via LCD:
  - Whether a GPS lock exists and the last known GPS coordinates
  - The displacement from start [m, X/Y/Z]
  - The “odometer” distance from start [m]
  - The current velocity [m/s]
  - The current acceleration [m/s^2]
- Additionally, the system shall record the following information to text file on SD during operation. The system’s best timestamped estimates for X, Y, and Z acceleration, velocity, and position from start
- The system will maintain a post-processed position update rate of at least 5 Hz
  - Acceleration measurements shall be taken at a minimum of 100 Hz [4]
  - GPS measurements shall be taken at a minimum of 1 Hz when GPS signal is present [5]

Schedule



Bibliography

[1] D. H. Titterton, and J. L. Weston. Strapdown Inertial Navigation Technology. Stevenage: Institution of Electrical Engineers, 2004.

[2] Rockwell Collins. GuS Integrated Sensor Suite [Brochure]. 2012 Warrenton, VA. Internet: http://www.rockwellcollins.com/~/media/Files/Unsecure/Products/Product%20Brochures/Controls/Flight%20Controls/Athena%20GuS/Athena%20GuS%20data%20sheet.aspx, 2012 [11/03/2013].

[3] Sparkfun. 9 Degrees of Freedom - MPU-9150 Breakout. Internet: https://www.sparkfun.com/products/11486, 05/14/2012 [11/03/2013].

[4] Invensense. MPU-9150 Product Specification. Internet: http://invensense.com/mems/gyro/documents/PS-MPU-9150A-00v4_3.pdf, 09/18/2013 [11/03/2013].

[5] Sparkfun. Venus GPS with SMA Connector. Internet: https://www.sparkfun.com/products/11058, 01/25/2011 [11/03/2013].

Equipment List

1. Raspberry Pi Revision A 256MB Single-Board Computer
2. Sparkfun MPU-9150 Breakout Board
3. Sparkfun Venus GPS with SMA Connector
4. Sparkfun Antenna GPS Embedded SMA