MetaSensor: Development of a Low-Cost, High Quality Attitude Heading Reference System

Gabriel Hugh Elkaim
Christopher C. Foster

BIOGRAPHY

Gabriel Elkaim received his B.S. degree in Mechanical/ Aerospace Engineering from Princeton University, Princeton NJ, in 1990, and both M.S. and Ph.D. degrees from Stanford University, Stanford CA, in Aeronautics and Astronautics, in 1995 and 2002 respectively. In 2003, he joined the faculty of the Computer Engineering department, in the Jack Baskin School of Engineering, at the University of California, Santa Cruz, Santa Cruz CA, as an Assistant Professor.

Christopher Foster received his B.S. degree in Computer Engineering from Santa Clara University, Santa Clara CA, in 2003 and is finishing his M.S. at University of California, Santa Cruz, Santa Cruz CA in Computer Engineering.

ABSTRACT

Current low-cost solutions to guidance, navigation, and control (GNC) of a small unmanned vehicle (underwater, ground, marine surface, or flight) still cost several thousands of dollars, and offer relatively poor performance at that price point. Tactical grade inertial navigation solutions (INS) run in the several tens of thousands of dollars, and often come with data restrictions due to arms treaties (ITAR). Given that navigation is, at its most basic, the enabling technology for unmanned systems, a lower cost solution becomes an enabling technology for the proliferation of these systems. The design philosophy is to reduce expenditures on sensors (which are expensive), and augment the system with better processing (which is cheap). We have embarked on the development of low-cost Attitude Heading Reference System (AHRS) that is based on MEMs technology sensors. The Metasensor consists of a three axis magnetometer, and three axis accelerometer, and a single axis rate gyroscope. Using novel calibration algorithms to correct for bias and scale factor errors, as well as non-linearities in the sensors, we are able to get superior performance using inexpensive sensors. The attitude is computed using a quaternion based estimation filter that linearizes the attitude in the Navigation frame. Our hardware/firmware combination achieves post-calibration results with standard deviations under 0.7 milli-g's and 0.6 milli-Gauss on the accelerometers and magnetometers, respectively. Combined with the quaternion attitude estimation algorithms, we are getting static performance with standard deviations of 0.12 degrees in pitch, 0.04 degrees in roll, and 0.08 degrees in yaw, with min to max deviations of less than 0.6 degrees on all axes. The Metasensor is more than just a low-cost AHRS, however. Follow-on work will turn the basic Metasensor into a full dead reckoning navigator capable of providing basic GNC subsystem functions to small UGVs for less than a thousand dollars.

1 INTRODUCTION

With the coming of the digital age there as been a drive to computerize everything. Computers are no longer just machines with keyboards, mice, and monitors, but now are embedded in cameras, cell phones, and music players. The next evolution is autonomous systems. An autonomous system is a computerized structure that removes humans from the loop by allowing a computer to take direct actions based on pre-programmed algorithms. Although autonomous systems in military applications bring up visions of robots taking over the world, there are many less frightening applications in use today. One example of an autonomous system that has implications in the transportation field is a vehicle that can navigate from one point to another while avoiding obstacles, such as what was required in a recent DARPA Grand Challenge [1].

One of the critical inputs to an autonomous system with a navigation component is attitude, or the orientation of the system with respect to a navigation reference frame. This paper presents the hardware and firmware required to form an inexpensive yet high quality sensor set sufficient to determine attitude. This hardware platform is flexible enough to accommodate future modifications, including additional hardware components and software algorithms, and has provisions for persistent data collection for post-analysis as well as communication hardware for the future output of live orientation data. Most importantly, this platform is low cost, allowing it to be used in a variety of cost sensitive applications.
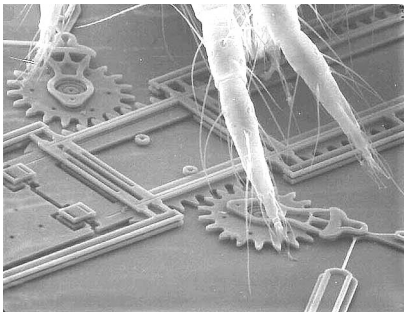
Figure 1. Spider Mite standing on gears of a MEMS motor mechanism courtesy of Sandia National Laboratories, SUMMiT^{TM}Technologies, www.mems.sandia.gov

1.1 Motivation

The three motivations for this project are cost, quality, and to a lesser extend, the use of new sensor technologies. As attitude determination has been a long standing problem, many solutions currently exist. Attitude hardware typically has a cost verses quality relationship: the higher the quality, the higher the cost. Our motivation is to create a high quality yet low cost attitude platform, something only possible through the use of new sensing technologies, a more accurate estimation algorithm, and more accurate calibrations.

Recently there has been a technological breakthrough in sensor construction that served as the inspiration for this project, namely MEMS devices and solid state magnetometers. Micro-Electro-Mechanical Systems, or MEMS are microscopic machines that allow for new approaches, especially in sensing the world. For an idea of scale, Figure 1 shows the legs of a Spider Mite on the gears of a MEMS motor mechanism. MEMS Accelerometers and MEMS Rate Gyroscopes are two technologies that have come from recent research that we use here for orientation measurements. Solid state magnetometers, also a relatively recent addition, provide a compact and accurate alternative to sensing magnetic fields.

These various motivations of high quality, low cost, and the use of new sensing technologies led to the formal specification of this project: To design and implement a hardware and firmware platform providing high quality readings from various sensors sufficient to provide quality attitude calculations while maintaining low cost and flexibility for both hardware and firmware expansion. The minimum requirements are high quality measurements of three axes of sensitivity to gravity and three axes of sensitivity to the earth's magnetic field at 100 Samples Per Second (SPS) with a means of persistent data storage and realtime data output. The additional goals are to include temperature and rate gyroscope inputs at the same quality and sample rate, as well as the connection for a GPS receiver. Eventually this hardware will provide the basis for a tightly coupled INS/GPS solution.

In the context of this project, high quality outputs at low cost translates to tactical grade outputs at consumer grade prices. Specifically, the long term goal is to be able to use this hardware to produce an Attitude Heading Reference System (AHRS) with errors less than $1.0°$ per hour for a retail price under $1,000.00. The hardware and firmware presented here is the first step in reaching this goal and has met both the minimum requirements and additional goals set forth in the formal specification.

1.2 Background

The three main approaches currently used in attitude determination are multiple antenna GPS, Rate Gyro Integration, Multiple Vector Measurements. GPS position and velocity errors and the assumption that the orientation of the object is identical to the direction of travel are two problems with using a single antenna GPS receiver to determine attitude. Differential GPS (DGPS) addresses this fist issue by using fixed receivers at know locations to broadcast current GPS errors to local DGPS receivers. The second problem can be addressed through the use of multiple antennas at various locations on the body of the object, the so-called Attitude GPS solution, however this approach requires expensive multiple antenna receivers and results in a low bandwidth. [2] and [4] both discuss applications of multiple GPS antennas to determine attitude, however these systems remain very expensive. Additionally, [10] notes that GPS is subject to potential interference and should not be relied upon solely for any critical application.

The second approach of using inertial sensors to determine attitude is along the lines of final goal of this hardware platform. An INS is the combination of an Inertial Measurement Unit (IMU) with the appropriate algorithms to calculate various values of interest for navigation such as position, attitude, and velocity. An IMU is a set of sensors that detect and record motion, typically made up of accelerometers and rate gyroscopes.

The hardware system presented in this thesis is designed so that it will be capable of becoming a low cost yet capable position sensor, however the current implementation is only that of an AHRS. An AHRS is a device that calculates and outputs attitude and heading but does not generate position or velocity. Although the final goal is a position/velocity sensor, we are currently only interested in what is required for an attitude solution, namely the three axes measurements of the earth's magnetic filed and three axes of measurements of gravity. Follow on work will extend this hardware solution to met its full potential as a position sensor.

2 THEORY
2.1 Basic Attitude

The attitude of an object is the orientation of the object's body fixed reference frame in terms of a navigation fixed reference frame. A reference frame is a coordinate system fixed to some object, and as their names imply, the

body fixed reference frame is fixed to the body of the object in question and the navigation fixed reference frame is fixed by our navigation reference. We have defined the body fixed reference frame with positive x-axis is towards the front of the hardware system, the positive y-axis is towards the right, and the positive z-axis down.

There are an infinite number of possible navigation frames and an infinite number of ways to measure orientations, however for a given application only a few will make sense. Our application is for a three dimensional attitude at or near the earth's surface so we will used the conventional Euler angles of pitch roll and yaw in a North-East-Down coordinate system. See [5], [6], and [8] for more information on reference frames and representations.

Solving for attitude is nothing more than solving for the appropriate orientation changes between the navigation fixed reference frame and the body fixed reference frame. With this understanding, we now must determine attitude with measurement of gravity and the earth's magnetic field. Both gravity and the earth's magnetic field are assumed for all points in our NED coordinate system. Gravity is a constant 1g throughout, and the earth's magnetic field has been mapped and modeled using [3]. The problem of determining attitude is to find the specific orientation of our body frame that will match the two known vectors in the navigation frame to the two measured vectors in the body frame, also known as Wahba's Problem.

2.2 Wahba's Problem

Named for Grace Wahba after her submittal of the problem to SIAM Review, Wahba's Problem is to determine the best fit rotation to match vectors that are measured in one reference frame and known in a second [12]. To avoid ambiguities, both vectors must be non-zero and must not be collinear. This problem was originally presented for attitude estimation of satellites where measurements made in a satellite fixed reference frame are compared to known values in the navigation frame.

One can visualize how the vector matching of the earth's magnetic field and gravity between two reference frames can be used to determine orientation in the following example. Imaging an object that suspended in air around San Francisco, California and near the surface of the earth. If the object is aligned with our NED coordinate system, that is the object is level and pointed north, it will be experiencing the earth's gravitational field pulling straight down at 1g. Looking up the value for earth's magnetic field at this point shows the object would also experience a 0.4193 gauss magnetic field at an angle $54.8990°$ below the horizon and $21.9425°$ east of north.

Now, imagine this object was to undergo a series of unknown rotations resulting in an unknown attitude. The object would still be experiencing the same fields, that is the vectors would have the same lengths, however they will no longer be in the same directions. The final attitude of the
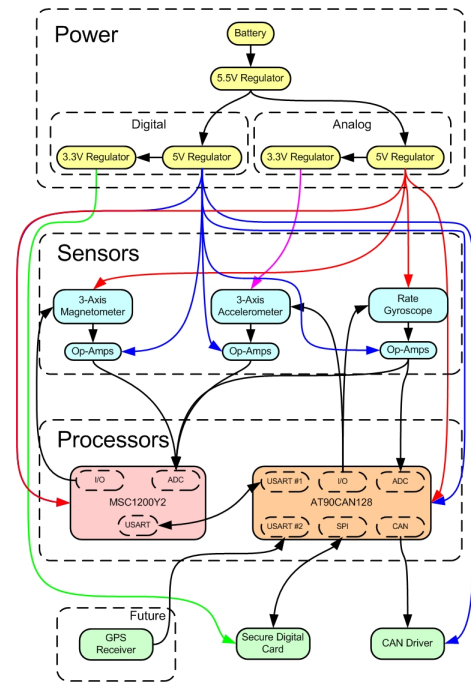

Figure 2. Functional block diagram of the hardware presented in this paper.

object could be found if, hypothetically, one could move the earth around the observation point, recording the rotation angles, until the observed field vectors match that of the object before any rotations.

This is exactly what Wahba's Problem is asking, what rotation of the navigation frame would result in a match between the navigation frame measured vectors and body frame measured vectors and, once solved, this rotation is the attitude of the body frame in reference to the navigation frame. There are several known solutions to this problem including the two presented in [12] along with a quaternion solution presented in [8]

3 HARDWARE

The original hardware specification for this project was to design and implement the hardware required to monitor and record, at the highest resolution possible and at a rate of 100 Samples Per Second (SPS), both the earth's magnetic field and gravity on the x, y, and z axes. These requirements have been met, then exceed, by including circuitry that allows for additional data points to be collected for future implementations. All of the hardware in this design will be discussed here, including items not part of the original specification and are not required for the attitude solution. Figure 2 shows a functional block diagram of the hardware components.

A total of ten data points are generated in this implementation and are classified into two categories. Eight of these data points are classified as primary sensors, including the six originally specified parameters plus values from

a rate gyroscope monitoring rotations about the *z*-axis and a temperature sensor embedded in one of the processors. The final two sensors, the two secondary sensors, are the rate gyroscope's internal temperature and internal 2.5 volt reference.

The primary sensors, except for the processor temperature, represent the most important data points and the ones most likely to change rapidly. Primary sensors are sampled at 100 SPS with 24-bits of resolution. The processor temperature is included as a primary sensor only because it is internal to the processor that is gathering the other primary data points, and it can not be monitored externally. Secondary sensors are less likely to change rapidly and are of less significance, so are monitored at only 10-bits of resolution by a separate ADC.

3.1 Power

For mobility, the core power source for this project is batteries. A single switching regulator with an input range of 12.5 volts to 8.0 volts outputs 5.5 volts to two linear 5 volt regulators. Two regulators are used to isolate the analog components from the digital components. There are two devices, one digital, one analog, that require 3.3 volts DC, so these components are powered from two separate linear regulators that are connected to the corresponding analog and digital 5 volt regulators.

3.2 STMircoelectronics LIS3L06AL Accelerometer

The LIS3L06AL from STMicroelectronics is a three axis MEMS accelerometer in a single 8-terminal Ceramic Land Grid Array (LGA) package for less than $15.00. This chip outputs three analog voltages, each between zero and the 3.3 volt supply, corresponding to accelerations in the *x*, *y*, and *z* directions. This sensor has two sensitivity modes, the first with a measurement range of ±2g's on each axis and the second with a range of ±6g's on each and is sensitive to both static accelerations such as gravity and dynamic acceleration such as vibrations. The LIS3L06AL detects accelerations by monitoring the position of a mass suspended within the chip and includes a self test feature.

Before leaving the chip, each signal is passed through an onboard resistor allowing for an external capacitor to be added between each output and ground to form a passive low-pass RC filter. For the purpose of this project it was decided that a minimum bandwidth of 20 Hz would be sufficient for acceleration in the *x*, *y*, and *z* directions so a 0.047 μF capacitor is used. Due to component tolerances, the resulting bandwidth is between 24.43 Hz and 40.51 Hz. Finally, the signals pass through separate Operational Amplifiers (OpAmps) with a nominal gain of 1.499. This amplifies a full scale output from the accelerometer of 3.3 volts to between 4.91 volts and 4.98 volts for the analog to digital converter.

3.3 Analog Devices ADXRS150 Yaw Rate Gyroscope

The ADXRS150 from Analog Devices is MEMS gyroscope in a 32 terminal Ball Grid Array (BGA) package for under $50.00. This chip outputs three analog voltages between zero and the supply voltage that represent the observed rotation rate in the plane of the chip, the temperature of the chip, and the internal 2.5 volt reference. The rate output of the sensor has a range of ±150 degrees per second of rotation and is determined by monitoring the Coriolis acceleration.

In addition to outputting the rate of rotation, the internal temperature, and the chip's internal 2.5 Volt reference, the ADXRS150 also provides a self test feature. The rotational rate output is amplified on the chip through an active low-pass filter, the capacitor of which is external to the chip and can be set by the user. In this case, a 0.015 nF capacitor is used for a nominal 58.9 kHz bandwidth, but due to component tolerances, has a range of between 55.5 Hz and 62.7 Hz. Finally, after leaving the chip, the signals go through their own unity gain OpAmps to boost the available current to the ADC.

3.4 Honeywell HMC1043 Magnetometer

Honeywell's HMC1043 is a three axis magnetometer in a 16-pin Leadless Plastic Chip Carrier (LPCC) package for less than $41.00. Each axes of the magnetometer consists of a four element Wheatstone bridge that is sensitive to magnetic fields within the range of ±6 gauss. As the magnetic fields change, the magneto-resistive elements will buildup a memory. To wipe the elements clean, the magnetometer has a pair Set/Reset straps built into the chip.

External circuitry including capacitors and dual channel MOSFET switches are used to generate both set and reset pulses sufficient to clear all three axes of the HMC1043 magnetometer. As the current drawn out of the two outputs must be negligible for the Wheatstone bridges to work correctly, the output pins from each magnetometer are connected directly to OpAmps. The high impedance inputs of an OpAmp result in insignificant current flow out of the magnetometers. Furthermore, running each OpAmp in a differential configuration allows for the two outputs for each axis to be subtracted at the same stage while also boosting the available current for the ADC conversions.

3.5 Processors

There are two 8-bit processors in use for this project, both running on the same circuit board and connected to each other through a serial buss running at 115,200 baud. This section will discuss the rolls of these two processors only to the level required to explain the hardware used. Further details of the programming will be discussed in Section 4.

3.5.1 Texas Instruments MSC1200Y2

The MSC1200Y2 from Texas instruments is an 8051-based microprocessor with an 9-input Delta-Sigma ADC, 4kB of

flash memory, 256 Bytes of SRAM, various I/O pins, and a Universal Synchronous/Asynchronous Receiver/Transmitter (USART) in a 48 pin Thin Quad Flat Pack (TQFP) package for less than $10.00. Although an internal 2.5 volt reference is available, a high precision external reference is used instead to increase overall performance. Programming of the MSC1200Y2 is performed through the non-standard use of serial port requiring special on-board hardware and custom software.

The hardware features that are used here include the serial programming capability, USART for communications, two 8-pin I/O ports, and the Delta-Sigma ADC. This processor includes a single Delta-Sigma ADC that can sample nine inputs (eight external inputs, one internal) through a multiplexor. The task for the MSC1200Y2 is to read the primary sensor analog inputs, do the Analog to Digital Conversions at 24-bits, send alternating set and reset pluses to the magnetometers though a pair of I/O ports, and finally output the ADC results across the processor's USART.

Do to the required sample rate, this task load results in an approximate 45% utilization of the processor. Synchronization of the various steps performed by this processor is critical so it was decided that it was not feasible to perform all the data manipulation and data storage on this single device. For this reason, a second processor is attached to the serial output of the MSC1200Y2 to collect, process, and store the readings.

3.5.2 Atmel AT90CAN128

Atmel's AT90CAN128 is an AVR series microcontroller with 128 kB of flash memory, 4 kB of SRAM, 53 I/O pins, an 8-port 10-bit ADC, dual USARTs, and a CAN interface in a 64 pin TQFP package for under $17.00. The hardware features that are used here are both USARTs, one for programming and the other for serial communication with the MSC1200Y2, the SPI buss to connect to the SD card, one I/O port for buttons and Light Emitting Diodes (LEDs) to indicate the SD card's status, and one I/O port to connect to the accelerometer scale pin and the self test pins on both the accelerometer and rate gyroscope. The CAN interface pins are connected to an external CAN driver and DB-9 port for future implementations.

The tasks dedicated to this processor are to collect data over the serial port from the MSC1200Y2, perform any data manipulation that may be needed, and finally save the data to an external Secure Digital (SD) card. Additionally, the required hardware is in place for connection to a GPS receiver, communication over a CAN interface, and performing ADC conversions of the secondary sensors in the future. Three external buttons and LEDs are provided for user interaction. In the current implementation, one button is used for both the mounting and un-mounting of a SD card's file system and the LEDs are used to initiate the SD card status and activity.
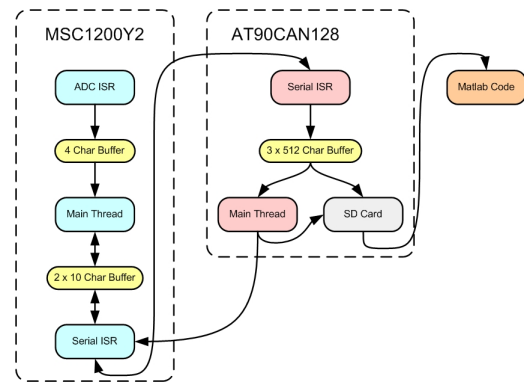

Figure 3. Functional block diagram of the software presented in this paper.

3.6 Data Storage

For persistent data storage, a Secure Digital card is attached to the AT90CAN128. Communication between the processor and SD card are conducted via an SPI interface. Because SD cards are 3.3V devices and are interfaced with a 5V device, 1kΩ resistors are placed between the SD card and AT90CAN128 on any signal going to the SD card. This results in a maximum, worst case current to through the SD card of 1.7mA, low enough not to damage the card. The maximum data rate available between the AtMega128 and the SD card is limited by the maximum speed of the processor's SPI buss, in this case a clock speed of 3.6 MHz.

4 SOFTWARE

The original software specification for this project mirrored that of the hardware specification, to design and implement the firmware required to monitor and record, at the highest resolution possible and at a rate of 100 Samples Per Second (SPS), both the earth's magnetic field and gravity on the x, y, and z axes. Just as with the hardware, these specifications have been met, then exceed, by provisioning for the future use of the additional hardware components.

There are three software components to this project, the code running on the MSC1200Y2, the code running on the AT90CAN128, and finally the Matlab code used to read the SD card and perform post analysis. Just as with the hardware, the software is designed to support reading all ten data points, even through only six are being used here. Figure 3 shows a functional block diagram of the software components presented here.

4.1 MSC1200Y2 Code

As mentioned earlier, the task for the MSC1200Y2 is to sample the primary data points at 24-bits and transmit those values across the USART while sending alternating set and reset pulses to the magnetometer. To perform these tasks, three separate threads run on the processor with context switches handled by Interrupt Service Routines (ISRs). The lowest priority thread is the main loop, the medium pri-
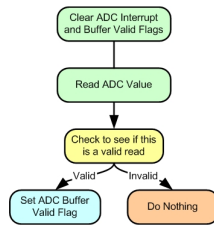
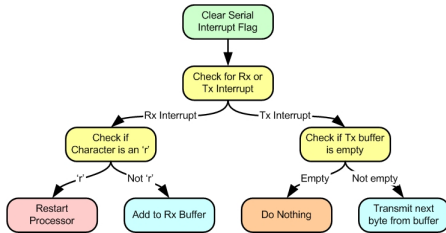**Figure 4.** MSC1200Y2 ADC ISR flow diagram.

Figure 5. MAC1200Y2 serial ISR flow diagram.

Figure 6. MAC1200Y2 main thread flow diagram.

ority thread is the USART ISR, and the highest priority thread is the ADC ISR. To reach the level of performance required, the most critical aspect to the code running on the MSC1200Y2 is the timing.

Every time an ADC conversion is complete, an interrupt is triggered. It is the ADC ISR's task to service that interrupt by clearing the interrupt flags, reading the ADC value, deciding if it is a valid conversion, and if so, setting the ADC buffer valid flag appropriately. Figure 4 shows the flow diagram for the ADC ISR.

Once the data is collected, it must be transmitted. 100 times per second a line of data is generated consisting of a 32-bit counter, the 9 ADC values each at 24-bits, and an 8-bit spacer between each of the 10 values, for a total of 328 bits. Serial data is transmitted in 8-bit increments with a start and stop bit added by the USART for a 20% protocol overhead. The serial communications between the MSC1200Y2 and AT90CAN128 is run at 115,200 bits per second (bps) even though less than 40,000 bps of data is transmitted. By transmitting data much faster than is actually required, we have both reduced the time needed to transmit each line and left space for more data in the future.

Every time the USART receives a byte or is done transmitting a byte, a serial interrupt is triggered. It is the roll of the Serial ISR to move new bytes from the transmit buffer to the USART buffer whenever a byte transmit is complete, and move received bytes into the receive buffer when bytes are received. Additionally, there is a special character, a lower case "r", that when received by the serial ISR will cause a reboot command to be generated, restarting the processor. Figure 5 shows the flow diagram for the Serial ISR.

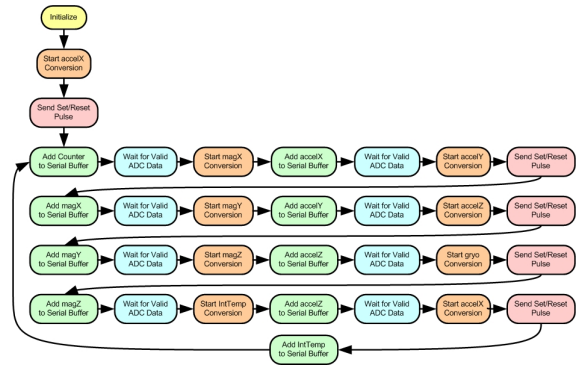The main thread acts as the glue between the Serial ISR and ADC ISR. The task assigned to the main thread is to first initialize the processor and all of the functions and then to loop over the conversions, that is switching the ADC input and starting the conversions, sending set/reset pulses, and once a valid conversion is complete, moving values from the ADC buffer to the serial transmit buffer. During the initialization processes, the first character received is used to calibrate the USART with the correct data rate and parameters. After calibrated, configuration settings such as ADC filter mode and raw sampling rate are sent over the serial line from to the MSC1200Y2.

As much effort as possible has been put into equally spacing all the sensor readings, placing everything that is not time critical in the waiting time between conversions. As soon as a conversion is complete the next conversion is started, and only then is the value from the previous conversion moved from the ADC buffer to the serial buffer. The order of magnetometer and non-magnetometer sensors have been staggered such that there is enough time to fire a set or reset pulse and wait for sensor's output to stabilize before each magnetometer reading is started. Figure 6 shows the flow diagram of the main thread.

All of the code written for the MSC1200Y2 is original for this project except for compiler include files, register definition files, and an ADC read utility written in assembly and provided by Texas Instruments. The Integrated Development Environment (IDE) and compiler used is a custom build from Raisonance specifically for the MSC series. The software used to interface with the custom hardware on the board and program the processor is a tool also provided by Texas Instruments.

4.2 AT90CAN128 Code

The task assigned to the AT90CAN128 software is to mount and initialize an SD card, read initialization parameters from the SD card and send them to the MSC1200Y2, collect data from the MSC1200Y2 over the serial port, perform computations on the data, and finally save the data to the SD card. When the AT90CAN128 is powered up it initializes itself and then waits for a button to be pressed indicating that the SD card has been inserted and is ready to be mounted. Once the button is pressed the SD card is mounted and the
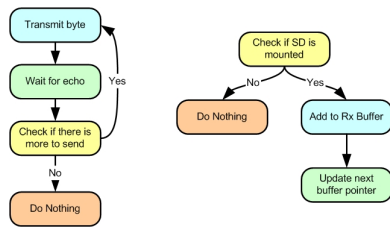
Figure 7. AT90CAN128 flow diagrams for the serial transmit loop (left) and serial receive ISR (right).

Figure 8. AT90CAN128 main loop flow diagram.

AT90CAN128 confirms that the SD card is formatted in FAT16. FAT16 is used because it is the easiest file system to implement while maintaining support for volumes up to 2 GB in size and allowing the card to be mounted on any compatible PC.

For a correct implementation of a FAT16 file system, every time the file grows to fill the current cluster, the file allocation table must be searched for the next available cluster, and that cluster added to the existing chain. Because the file allocation table is more than one sector long, multiple sectors may need to be read and written to find the next available cluster and add it to the existing chain. This procedure results in inconsistent sector write times and is unacceptable for our time sensitive approach. To avoid these sporadic sector write wait times, this implementation makes use of a single continuous scratch file and data is written in 512 byte sectors starting at the beginning of the scratch file and continuing until the end.

The AVR library provided by ProcyonEngineering allows for USART transmits and receives to be handled independently. As transmissions from the AtMega128 to the MSC1200Y2 are done only at the very beginning and very end, and are not time critical, a polled-waiting loop is used. Receiving data on the other hand is very time critical as data can show up in the receive buffer at any time and must be removed before another byte arrives and overwrites the buffer. A serial receive ISR is used to move new bytes from the receive buffer into the data buffer as soon as they arrive. Figure 7 shows flow diagrams for both the serial transmit polled-waiting loop and the serial receive ISR.

A set of three 512 byte buffers are maintained for the serial data received by the AT90CAN128. The receive ISR writes directly into these buffers and the main loop monitors their status. As soon as one of the buffers is full, the SD write function is called to write the buffer contents to the next sector on the SD card. The SD write function is provided by ProcyonEngineering's AVR library.

When the un-mount button is pressed, the serial receive function stops filling the buffers, the last partial buffer is sent to the SD card, both the directory table and file allocation tables are updated to reflect the correct file size, the remaining unused linked clusters become the new scratch file, and a restart character is sent to the MSC1200Y2 so that it will be ready to start over collecting data. Figure 8
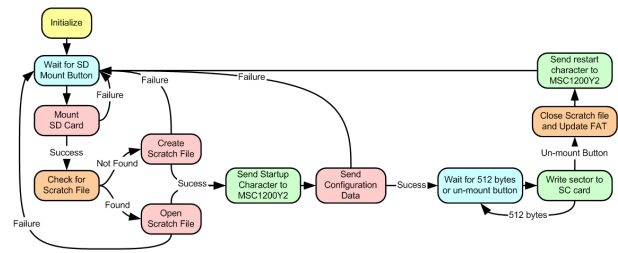
shows the flow diagram of the AT90CAN128's main loop.

In the final implementation, 1,641 bytes of SRAM are used for data leaving the remaining 2,455 bytes for stack space and future implementations. The majority of the code for the AT90CAN128 is original except for configuration and helper functions provided by ProcyonEngineering and files included in the WinAVR compiler. Files included in a Circuit Cellar article on FAT16 implementation were used as a guide [11]. Programming of the AT90CAN128 is performed through special hardware on the board and AVR-Dude software.

4.3 Matlab Data Import Code

Because the data is stored on the SD card's FAT16 file system, the SD card can be removed, placed into a computer, mounted, and data files copied. Data import code can then read the data files into Matlab for post processing.

5 CALIBRATION

To extract accurate readings from corrupt sensor data, mathematical models must be built to take into account the various sources of error. There are many potential error sources for a sensor, some that apply to all types of sensors, some that apply only to a particular class of sensors, and some that are caused by what the sensors are trying to observe. For the purpose of this paper, only three forms of error will be considered, null shift errors, scale errors, and alignment errors. Null shift errors cause a constant shift of a sensor's output, scale errors cause in an amplification if a sensor's output, and sensor misalignment cause correlations between the sensors of a multiple axis sensor set. For our purposes, we do not care what the sources of these errors are, but rather care only about the total error of each class.

To simplify our error model, we will make the assumption that all error sources are constant over other variables such as position, time, and temperature. Although for a particular sensor this is likely incorrect, these assumptions result in a sufficiently accurate result. This approach can be readdressed at a later time to further increase the accuracy of this model and the overall result.

A forth type of error that will be considered only to be dismissed is sensor noise. There is a white noise present on all sensors, as well as quantization noise on the ADCs. Here, this noise will be addressed by over-sampling and

averaging. If enough samples of the same point were to be taken and averaged, white gaussian noise would not appear in the result. Over-sampling of an input and averaging the results can be done in software, but can be done much more cleanly and accurately in hardware, such as in a Delta-Sigma ADC. The very definition of a Delta-Sigma ADC is an analog to digital converter that samples an input over a period of time and outputs a value corresponding to the average. This is precisely why the quality of the output of an Delta-Sigma ADC increases the longer it sits on a single input.

5.0.1 Single Sensor Model

The single sensor model is a mathematical model that address the error sources for a stand alone sensor. Our model for the output of a non-ideal single sensor takes into account scale errors and offset errors and is:

$$x_m = ax_t + x_o \quad (1)$$

where x_m is the measured value, a is the combination of all scale errors, x_t is the true value, and x_o is the the combination of all constant errors. Calibration would be required for an individual sensor to determine the two unknown variables a and x_o. Although this is a valid error model for an individual sensor, it is not sufficient for sensor sets designed to monitor multiple axes.

5.0.2 Three Axis Sensor Model

To monitor a real world value in three dimensions, three individual sensors can be mounted orthogonally and combined into a single, three axis sensor set. In this configuration, misalignment error parameters must be added to our error model. Various misalignment errors can be the result of several factors not only a physical misalignment.

First, we assume that the x-sensor is perfectly aligned with the x-axis, and therefore Equation 1 holds for the x-sensor and we can solve for x_t as:

$$x_t = \frac{x_m - x_o}{a} \quad (2)$$

Next, we define ϕ as the y-sensor misalignment, or the angle between the y-sensor and the orthogonal y-axis. Because of this misalignment, the y-sensor is not only sensitive to fields y-axis, but is also sensitive to the x-axis. The equation for the y-axis becomes:

$$y_m = b\left(y_t \cos(\phi) + x_t \sin(\phi)\right) + y_o \quad (3)$$

where b is the scale, y_t is the true value, and y_o is the offset. Taking a moment to examine this equation, one can see that if there were no misalignment, that is $\phi = 0$, Equation 3 would reduce to $y_m = by_t + y_o$ as we would expect. Likewise, if the misalignment were complete, that is $\phi = \pm 90°$,

the equation would be $y_m = bx_{true} + y_o$, also like we would expect.

Plugging Equation 2 into Equation 3 and solving for y_t gives:

$$y_t = \frac{\frac{y_m - y_o}{b} - \left(\frac{x_m - x_o}{a}\right)\sin(\phi)}{\cos(\phi)} \quad (4)$$

Now that we have equations for the x and y-sensors, we turn our attention to the z-sensor. Two more misalignment angles must now be defined, ρ, the angle between the z-sensor and the true, orthogonal x-z plane, and λ, the angle between the z-sensor and the y-z plane. Using the same approach as for the y-axis sensor, the equation for the z-sensor is:

$$z_m = c(z_t \cos(\rho)\cos(\lambda) + x_t \sin(\rho)\cos(\lambda)$$
$$+ y_{true}\sin(\lambda)) + z_o \quad (5)$$

where c is the scale, z_t is the true value, and z_o is the offset. Plugging the previously solved x_t and y_t into equation 5 and solving for z_t gives:

$$z_t = \frac{\frac{z_m - z_o}{c} - \left(\frac{x_m - xo}{a}\right)\sin(\rho)\cos(\lambda) - \frac{\frac{y_m - y_o}{b} - \left(\frac{x_m - x_o}{a}\right)\sin(\phi)}{\cos(\phi)}\sin(\lambda)}{\cos(\rho)\cos(\lambda)} \quad (6)$$

We have now solved for the true values on the x, y, and z axes based the measured values and as a function of the scale errors, offset errors, and misalignment errors. Now calibrations must be performed to determine the nine unknown error parameters a, b, c, x_o, y_o, z_o, ϕ, ρ, and λ. Once these values are found, Equations 2, 4, and 6 can be used to calculate real world values from live sensor data.

5.1 Calibration Procedure

A valid calibration procedure is a set of routines and algorithms that will produce the required error parameters so that the real world values can be generated from raw data. The calibration procedure used here is an expansion and generalization of that discussed in [9].

The rotation of a perfect three axis sensor set within a constant field will generate x, y, and z data points that, when graphed on a three axis plot, will all lay on the surface of a perfect sphere with a radius equal to the strength of the detected field and centered at the origin. If the three axis sensor set is not ideal, the various forms of error discussed previously will serve to alter the resulting sphere. Scale errors will stretch the sphere into an ellipsoid, offset errors will shift the sphere away from the origin, and misalignment errors will distort the sphere to what looks like a rotated ellipsoid. Due to these observations we know that there is a correlation between the nine values a, b, c, x_o, y_o, z_o, ϕ, ρ, and λ and the parameters that would distort a sphere to this ellipsoid.

5.1.1 The Calibration Ellipsoid

The equation for a sphere of radius R is $R^2 = X^2 + Y^2 + Z^2$. If an ideal x-y-z sensor set was rotated in a constant field of magnitude R, the resulting plot on a three dimensional axis would be $R^2 = x_t{}^2 + y_t{}^2 + z_t^2$. Instead of ideal sensors, the sensor models that we are using are expressed in Equations 2, 4, and 6. Substituting in these equations gives the equation for our shifted, stretched, and tilted sphere as:

$$R^2 = \left(\frac{x_m - x_o}{a}\right)^2 + \left(\frac{\frac{y_m - y_o}{b} - x_m \sin(\phi)}{\cos(\phi)}\right)^2 +$$

$$\left(\frac{\frac{z_m - z_o}{c} + \left(\frac{x_m + x_o}{a}\right)\sin(\rho)\cos(\lambda) + \frac{\left(\frac{y_m + y_o}{b}\right) + x_m \sin(\phi)}{\cos(\phi)}\sin(\lambda)}{\cos(\rho)\cos(\lambda)}\right)^2 \tag{7}$$

Equation 7 is not linear in terms of the error parameters, however a two-step estimation approach can be performed [7]. If one was to expand Equation 7 they would find that it can be expressed in the following form:

$$z_m{}^2 = Ax_m{}^2 + By_m{}^2 + Cx_my_m + Dx_mz_m + Ey_mz_m$$
$$+ Fx_m + Gy_m + Hz_m + I \tag{8}$$

Equation 8 is linear in terms of A through I which are in turn algebraic functions functions of the error parameters and R. This relationship can be written in matrix form and n data points combined into:

$$\mathbb{X} \times \mathbb{P} = \mathbb{W} \tag{9}$$

where $\mathbb{X}$ is an $n \times 9$ matrix with each row being one measurement, $\mathbb{P}$ is a 9×1 matrix consisting of parameters A through I, and finally $\mathbb{X}$ is a $n \times 1$ matrix of ones. Finally, $\hat{\mathbb{P}}$, the best-fit estimation of $\mathbb{P}$ in the least squares sense, can be found as:

$$\hat{\mathbb{P}} = (\mathbb{X}^T\mathbb{X})^{-1}\mathbb{X} \tag{10}$$

and the error parameters can then be solved algebraically.

To review, we now know that we can rotate a non-ideal three axis sensor set in a constant field and graph the raw data and it will approximate a shifted, distorted ellipsoid. We also now have equations representing all three sensors that takes into account scale errors, null offset errors, and orthogonality errors. Combining these two items, using a two-step, non-linear estimator, and assuming we know R, we can find estimations for the error parameters a, b, c, x_o, y_o, z_o, ϕ, ρ, and λ.

5.2 Calibration Results

The circuit board used for this project was rotated through various angles and more than 78,000 data points from the three axis magnetometer sensor were collected. After recording data, the calibration algorithm discussed in this paper was used to generate calibration parameters and correct the
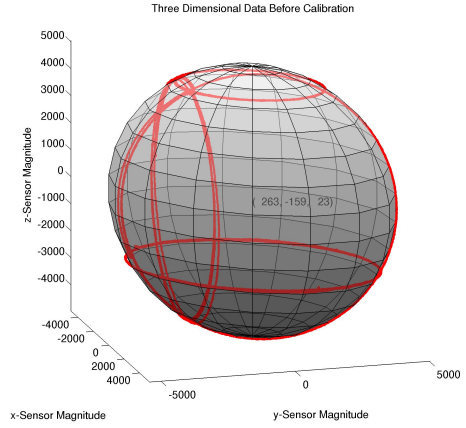

Figure 9. Raw data from three axis magnetometer plotted on estimation ellipsoid based on calibration parameters with center point labeled.
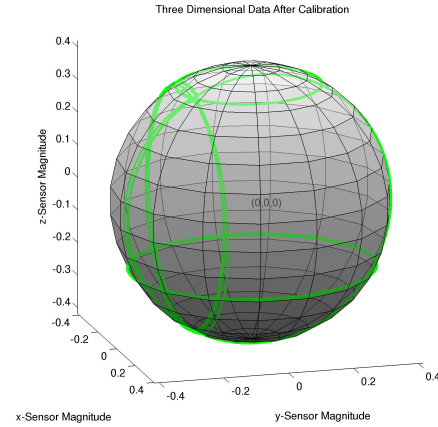

Figure 10. Corrected data from three axis magnetometer plotted on sphere with radius equal to the earth's magnetic field and center pointed labeled.

raw data. The total earth magnetic field at the point where this data was collected was found using [3] as 0.4193 gauss. Figure 9 shows the raw data on the best fit calibration ellipsoid, and Figure 10 shows the corrected results on a sphere with radius equal to the magnitude of the earth's magnetic field. Table 1 shows the calibration results for both the magnetometers and accelerometers.

6 EXPERIMENTAL RESULTS

There are three criteria that we will use to analyze the quality of the sensor outputs, specifically, sensor noise, sensor drift, and attitude repeatability. Sensor noise is the short term corruption of data that gets through the Delta-Sigma ADC to contaminate the sensor readings. Sensor drift is a long term instability of sensor output resulting in a drift of output values over time. Attitude repeatability is the ability of the sensors to output the same values every time the

	Magnetometer Set	Accelerometer Set
x_o	-27.30	1,366.15
y_o	-379.69	-530.69
z_o	659.29	2,523.66
a	12,657.39	13,519.11
b	13,045.14	14,165.62
c	12,6693.89	14,680.03
ϕ	-6.13°	0.51°
ρ	-5.97°	0.02°
λ	2.69°	-0.09°

Table 1. Results from calibration of magnetometer and accelerometer sensor sets.

hardware is placed in the same orientation.

A rotation rig was constructed to allow the circuit board to freely rotate in any direction. Constructed from three concentric circles allowed to rotate about pivots, this tool allows a stable rotation about a specific center point and has been used to evaluate the output quality. Additionally, two laser pointers attached to the circuit board and mounted at 90° to each other allow for repeatability of specific attitudes by pointing at specific marked points on surrounding walls.

In order to test repeatability, the rig was repeatedly placed in several specific orientations with pauses for data collection. These specific orientations were determined by the two laser pointers and marks on surrounding walls. Next the rig was left in a random orientation for several hours for the collection of long term data.

Using tactical grade as our target, our goal was to achieve a drift of no more than 1.0° in heading over a one hour period and an attitude repeatability with no more than 1.0° of error.

6.1 Sensor Noise

After calibrating the three axis magnetometer and three axis accelerometer, short segments of stable data are examined for sensor noise. To address noise on the individual sensors we look at both the standard deviation as well as minimum and maximum reading of each sensor output over a one minute period. If there was no noise able to get through the Delta-Sigma ADCs, the standard deviations would be zero and the minimum and maximum be the same value for each sensor.

Examining the outputs from all six sensors over the same one minute period showed incredibly clean results. Figures 11 and 12 shows one minute of data from all three sensors of the accelerometer and magnetometer sensor sets respectively. The magnetometer readings have standard deviations of 0.15% of the full scale range and a minimum to maximum variation under 1.25% while the accelerometers readings have standard deviations of 0.07% of full scale range and a minimum to maximum variation below 0.60% for the one minute of data.
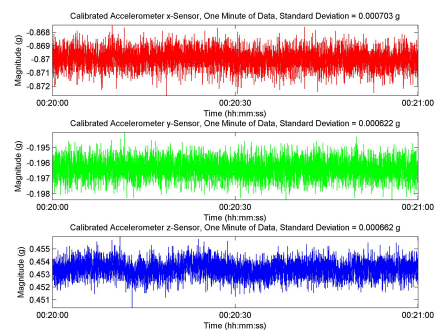

Figure 11. One minute of data from the three calibrated accelerometers.
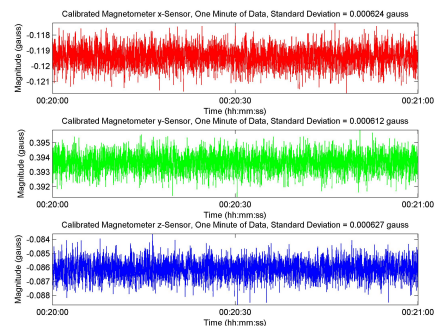

Figure 12. One minute of data from the three calibrated magnetometers.

Sensor noise must also be addressed in how it effects the entire sensor set. The total magnitude of the fields observed by each sensor set is calculated as the square root of the sum of the squares of the three axes of sensitivity. Figure 13 shows the total magnitudes sensed for the same one minute period on both the magnetometer and accelerometer sensor sets. The magnetometer magnitude has a standard deviation of 0.14% of the full scale range and a minimum to maximum variation of 1.08% while the accelerometers readings has a standard deviation of 0.07% of full scale range and a minimum to maximum variation of 0.51% for the one minute of data.

Here we note here that the noise parameters of the individual sensors and the sensor sets as a whole are nearly identical. Secondly, we note that the mean magnitudes for the two sensor sets are very close to the known magnitude of the sensed fields, specifically a magnetic field of 0.4193 gauss and an acceleration of 1 g. The preliminary attitude calculations from these readings are shown in Figure 14 and have standard deviations of less than 0.125° and total drifts of under 0.6° for each of the three Euler angles.

6.2 Sensor Drift

The second performance metric of interest is long term sensor drift. Figure 15 shows the total magnitude of one hour
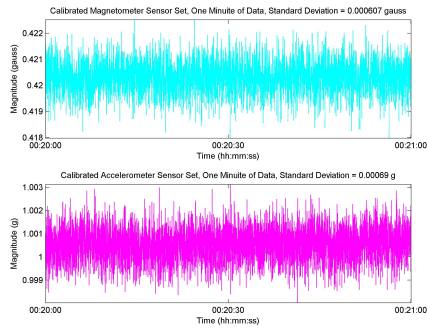
Figure 13. Total magnitude of one minute of data from the calibrated magnetometer and accelerometer sets.

Figure 14. Graph of pitch, roll, and yaw angles calculated for one minute of data.
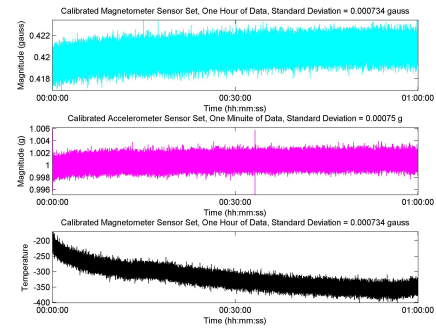

Figure 15. Total magnitude of one hour of data from the calibrated magnetometer and accelerometer sets plus raw temperature readings from rate gyroscope.
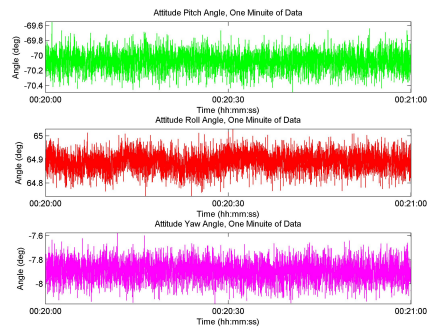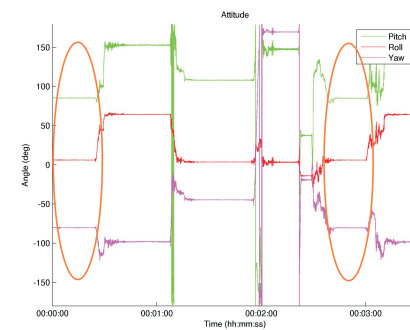

Figure 16. Pitch, roll, and yaw calculated for attitude repeatability data, areas of concern circled. Note: Vertical lines are jumps between negative and positive 180 degrees.

of data from the magnetometers and accelerometers in addition to a raw, un-calibrated temperature readings. This sensor drift falls slightly outside our $1.0°$ target with standard deviations on the order of $0.15°$ and minimum to maximum variations of up to $1.3°$, however improvements can still be made. One of the assumptions of our calibration algorithm is that our sensor readings are not a function of temperature, however this is not the case. As can be seen in Figure 15, there is an obvious correlation between temperature and sensor readings of both the accelerometer set, and more notably magnetometer sensor set. Modification of our calibration algorithm to take this into account will significantly reduce our sensor drift and has been left for future work.

6.3 Sensor Repeatability

The final test is for repeatability of sensor output when the hardware is put in the same orientation. As described previously, using two laser pointers attached to the rotation rig the hardware was able to be placed in a specific orientation, rotated through various angles, then put back in the same orientation. The results of this test were within our performance requirements and an example of the values from one of the orientations is shown in Figure 16 with the repeated attitude circled. The attitude calculations show that this hardware platform generates repeatable readings to the

order of $0.25°$ with it quite likely that some, if not most, of that error is due to the inability to perfectly align the laser pointers with their previous positions.

7 CONCLUSIONS

A new Attitude Heading Reference System has been presented that is of both high quality and low cost. This hardware and firmware system currently shows sensor noise resulting in errors of less than one degree, sensor drift of slightly over one degree per hour, and repeatability under a quarter of a degree. Further calibration that compensates for temperature and software filters that reduce stray readings are expected to further improve these results.

The hardware precedented is compete for the AHRS solution and is built such that it will also be sufficient for a complete positioning solution. Like the hardware, the software is flexible enough such that this one platform is sufficient for future developments. The final INS/GPS solution that will result from this development process will be a drop in replacement for any single GPS application and provide more accurate results at higher bandwidths.

8 FUTURE WORK

This project has generated the required hardware for a high quality and low cost attitude solution and has been the first step in producing a Inertial Navigation System, also of high quality and low cost. There are several aspects that are left for future work. To finish the AHRS solution, the following items must be completed:

- Modify the calibration algorithm to include temperature. The results of the existing calibration algorithm is very accurate when our assumptions are correct, but when the temperature is not stable there is an effect on the sensor outputs, most notably the magnetometers.

- Modify the calibration algorithm to align the x, y, and z axes of the magnetometer to the x, y, and z axes of the accelerometer. The current calibration algorithm does compensate for misalignments within a sensor set but does not compensate for misalignments between sensor sets.

- Generate, simulate, and test an algorithm to compute attitude solutions from the three axis magnetometer and three axis accelerometer values. This algorithm will likely be similar to that presented in [8] and used here, but the final implementation will be specific for this application and should run on the AT90CAN128 and the results outputted on the CAN interface.

And the following items are left as future work in addition to the items above to finish the complete GPS/INS solution:

- Generate a calibration algorithm for the rate gyroscope. A gyroscope is provided in this hardware suite for the positioning solution, however it needs to be calibrated for accurate results.

- Attach a GPS receiver to the provided serial port. The second USART on the AT90CAN128 is provided for a GPS receiver and code to interface GPS to the processor is provided by ProcyonEngineering.

- Generate, simulate, and test an INS algorithm that will take as input three axes of magnetic field, three axes of acceleration, one plane of angular rate, and GPS to generate a high quality output consisting of, at a minimum, longitude, latitude, altitude, attitude, and velocity.

REFERENCES

[1] DARPA Grand Challenge Home. http://www.darpa.mil/grandchallenge/, September 2006.

[2] Brian T. Baeder and Jeff L. Rhea. Gps attitude determination analysis for uav. In Scott A. Speigle, editor, *Navigation and Control Technologies for Unmanned Systems, Orlando, FL*, volume 2738, pages 232–43. SPIE, May 1996.

[3] C. E. Barton. Revision of international geomagnetic reference field release. *EOS Transactions*, April 1996.

[4] M. Elizabeth Cannon and Gérard Lachapelle. Submeter real-time differential gps and attitude determination for unmanned navigation. In Scott A. Speigle, editor, *Navigation and Control Technologies for Unmanned Systems, Orlando, FL*, volume 2738, pages 244–55. SPIE, May 1996.

[5] G. Creamer. Spacecraft Attitude Determination Using Gyros and Quaternion Measurements. *The Journal of Astronautical Sciences*, 44(3):357 – 371, July - September 1996.

[6] G. H. Elkaim. *System Identification for Precision Control of a WingSailed GPS-Guided Catamaran*. PhD thesis, Stanford University, Stanford, CA, 2001.

[7] C. C. Foster and G. H. Elkaim. Development of the Metasensor: A Low-Cost Attitude Heading Reference System for use in Autonomous Vehicles. In *Institute of Navigation ION-GNSS Conference, Fort Worth, TX*. ION, 2006.

[8] D. Gebre-Egziabher, G. H. Elkaim, J. D. Powell, and B. W. Parkinson. A Gyro-Free, Quaternion Based Attitude Determination System Suitable for Implementation Using Low-Cost Sensors. In *Proceedings of the IEEE Position Location and Navigation Symposium, PLANS 2000*, pages 185 – 192. IEEE, 2000.

[9] D. Gebre-Egziabher, G. H. Elkaim, J. D. Powell, and B. W. Parkinson. A non-linear, two-step estimation algorithm for calibrating solid-state strapdown magnetometers. In *8th International St. Petersburg Conference on Navigation Systems, St. Petersburg, Russia*. IEEE/AIAA, 2001.

[10] Demoz Gebre-Egziabher. *Design and Performance Analysis of a Low-Cost Aided-Dead Reckoning Navigation System*. PhD thesis, Department of Aeronautics and Astronautics, Stanford University, Stanford, California 94305, December 2001.

[11] W Hue I Sham and P Rizun. Portable fat library for mcu applications. *Circuit Cellar*, pages 18–26, March 2005.

[12] Grace Wahba. Problem 65-1 (Solution). *SIAM Review*, 8:384 – 386, 1966.