Journal of Communication and Computer 11(2014) 469-477
doi: 10.17265/1548-7709/2014.05 007

Implementation of Kalman Filter on PSoC-5 Microcontroller for Mobile Robot Localization

Garth Herman, Aleksander Milshteyn, Airs Lin, Manuel Garcia, Charles Liu, Darrell Guillaume, Khosrow Rad and Helen Boussalis

Department of Electrical and Computer Engineering Structures, Pointing, and Control Engineering (SPACE) University Research Center

Abstract: Robots facilitate exploration of hazardous environments during response to catastrophe. Autonomous robotic platforms involved in search and rescue operations require accurate position and orientation (localization) information for self-navigation from its current position to its subsequent destination. A Hybrid Routing Algorithm Model has been proposed by the SPACE (structures, pointing and control engineering) URC (university research center) at California State University of Los Angeles. This model envisions three-layered terrain mapping with obstacle representations from various information sources such as satellites, UAVs and onboard range sensors. A* path-finding algorithm is applied to the outer two layers of the model (Layer 1 and Layer 2), while dynamic A* algorithm is responsible for innermost layer (Layer 3) navigation. The mobile robot localization information is computed using data obtained from a 9 Degrees of Freedom Inertial Measurement Unit. While gyroscope sensors provide the system the instantaneous radial velocity of a turning platform, these sensors are also susceptible to drift. Accelerometers are extremely sensitive to vibrations, and along with fluctuating magnetic fields, both accelerometers and magnetometers exhibit noisy behaviors when localizing the robot. Since the IMU contains all three sensors, a Kalman Filter is implemented on a PSoC-5 microcontroller to fuse data from the IMU sensors. This reduces standard deviation between measurements and improves reported heading accuracy, hence provides reliable information on the robot’s localization and improves mapping.

Key words: System-on-chip, mobile robot, Kalman Filter, IMU.

1. Introduction

Advances in robotics technology increase the utility of diversely applied platforms. Robots mitigate risk to emergency responders, and facilitate exploration of dangerous environments. Emergency responders may use the device to identify the location of victims of catastrophe. Exploration of Mars depends upon versatile robotic platforms. Navigation algorithms are necessary to accurately guide these vehicles through the surrounding environment avoiding structural obstacles. Mars Exploration Rovers utilize multiple localization methods, including the deployment of an accelerometer, gyroscope, magnetometer and motor encoders, to navigate an extraterrestrial environment [1]. Rovers have several advantages over stationary nodes, as they examine more territory, can be directed to points of interest, and position themselves to sunny locations during winter months.

This work introduces a semi-autonomous control, where a HCS (host computing station) assigns global tasks to the mobile robot and is operated either manually or autonomously. The mobile robot initially informs the HCS of its facing direction and receives destination coordinates information from the HCS. An onboard computer determines the optimal path and commands the robot to move between intermediate nodes which lie along the calculated path towards the intended destination. The mobile robot’s onboard sensors include motor encoders, accelerometer, gyroscope and magnetometer, which inform the HCS of the current robot location within a margin of error

Corresponding author: Aleksander Milshteyn, M.Sc., research fields: computer engineering, embedded systems. E-mail: tashkent04@hotmail.com.

relative to the sensors. Unfortunately, inaccuracies in the output collected from each individual sensor introduce unreliable data to the system.

Several sensors provide information pertinent to the localization of a robot. An accelerometer indicates the direction of gravity when at rest. A gyroscope indicates radial velocity. A magnetometer measures the strength of surrounding magnetic fields. Each of these sensors introduces noise to the system and the gyroscope introduces drift. A magnetometer indicates direction, but results are also noisy and do not provide high precision measurements.

Data fusion techniques have been investigated to fuse each of these sensors and to allow the robot to accurately report localization information to the HCS.

The scope of this paper includes the system architecture of the semi-autonomous robotic platform, the utilization of the PSoC-5(Programmable System on Chip-5) microcontroller, communication between the PSoC-5 and embedded computer and the implementation of Kalman Filter [2]. The rest of this paper is organized as follows: Section 2 describes multiple codependent computers that comprise the system architecture of the proposed mobile robot; Section 3 explains communication protocols developed to facilitate communication between the HCS, onboard laptop, and a Programmable System on Chip-5LP; Section 4 describes the Kalman Filter applied to the Inertial Measurement Unit control algorithm and its software implementation; Section 5 compares the result of the Kalman Filter to the unfiltered kinematic data; Section 6 concludes the paper and presents future work.

2. System Architecture

The robotic platform is operated from a HCS that receives aerial imagery. The operator uses this imagery to indicate the presence of large obstacles on a low-resolution map, and provide a destination to the robot. The robot utilizes a PSoC-5 to acquire sensor data and perform control algorithms. An embedded computer provides real-time path-finding and creates a high-resolution local map.

The robotic system architecture consists of two primary components, the HCS and the onboard embedded hardware. The HCS is responsible for calculating the optimal paths on environment maps of different resolutions. Environment mapping is divided into three maps, defined at resolutions of 1 km by 1 km (Level 1), 100 m by 100 m (Level 2) and 10 m by 10 m (Level 3). A pilot specifies a beginning and ending destination on the Level 1 map. The HCS is responsible for calculating static optimal paths for Level 1 and Level 2 scaled maps. The HCS receives incoming map data from an aerial or space vehicle and communicates the path information to the mobile robot. The embedded computer performs a dynamic path finding algorithm to traverse Level 3 maps [3]. The HCS creates Level 1 and Level 2 maps from incoming aerial map data. The HCS also processes robotic sensor data, maps visualization data in an OpenGL environment on three scaled levels, and tags multimedia data to the appropriate coordinate location. Finally, the HCS provides remote control capabilities in missions requiring human intervention.

The embedded hardware consists of three primary layers: the Algorithm Layer, Platform Layer, and Driver Layer. The Algorithm Layer is responsible for computing navigational search algorithms, obstacle detection, and obstacle avoidance on Level 3 maps. The Platform Layer is responsible for the steering, sensor fusion, image processing, and kinematics of the mobile robot.

The Driver Layer contains the sensor interfaces used for Robotic unit. Sonar and Infrared sensors interface with the PSoC-5. The Inter-Integrated Circuit (I2C), PWM (pulse-width modulation) control, ADC (analog-to-digital converter) and UART (universal asynchronous receiver/transmitter) standards are used for the above sensor interfaces. The robot system architecture is depicted in Fig. 1.

The PSoC-5 serves as the microcontroller to support sensors, motor controllers and communications for the mobile robot. The PSoC provides the current localization of the mobile robot to the controlling computer through a serial port. The PSoC-5 performs data acquisition, applies control algorithms and standardizes output, while the controlling computer handles all path-finding logic.

The PSoC-5 acquires kinematic sensor information. Navigation data determined in the Algorithm Layer and passed through the Motor Control Algorithm block of the Platform Layer on the PSoC-5 drives an actuator interface to control motors. The motor kinematic sensors obtain data for proper mapping and unit localization. Sonar and IR sensors, along with Microsoft Kinect image acquisition, provide real time obstacle monitoring to allow the mobile robot to sustain autonomous and semi-autonomous missions. Fig. 2 illustrates the direction of information flow the embedded system layers, excluding communications. The complete system interaction diagram is demonstrated in Fig. 3.

3. Programmable System on a Chip

Unique advantages of the Cypress PSoC-5 microcontroller include programmable voltage, instrumentation and amplifiers. PSoC-5 provides configurable timers, counters and PWM units and each component is accessible through a proprietary application-programming interface. The PSoC-5 Digital Subsystem features configurable UDB (universal digital block) units. Each UDB contains programmable array logic and programmable logic device functionality.

A wide variety of peripherals are supported. In addition to the UDB array's flexibility, PSoC-5 provides configurable dedicated digital blocks that target specific functions. UDB units utilized by the mobile robot include up to 32-bit PWM, counter and timer UDB units. A Dedicated I2C communication block is also used. Multiple clocks can be configured to drive the digital components. The Analog Subsystem of PSoC-5 includes multiplexors, comparators, voltage references, op-amps, mixers and Trans Impedance Amplifier. Digital and Analog signals can be routed by any GPIO pin [4].

A lack of parallel processing, small RAM capacity, and a maximum clock rate of 48 MHz provide certain limitations for PSoC-5. A controlling embedded

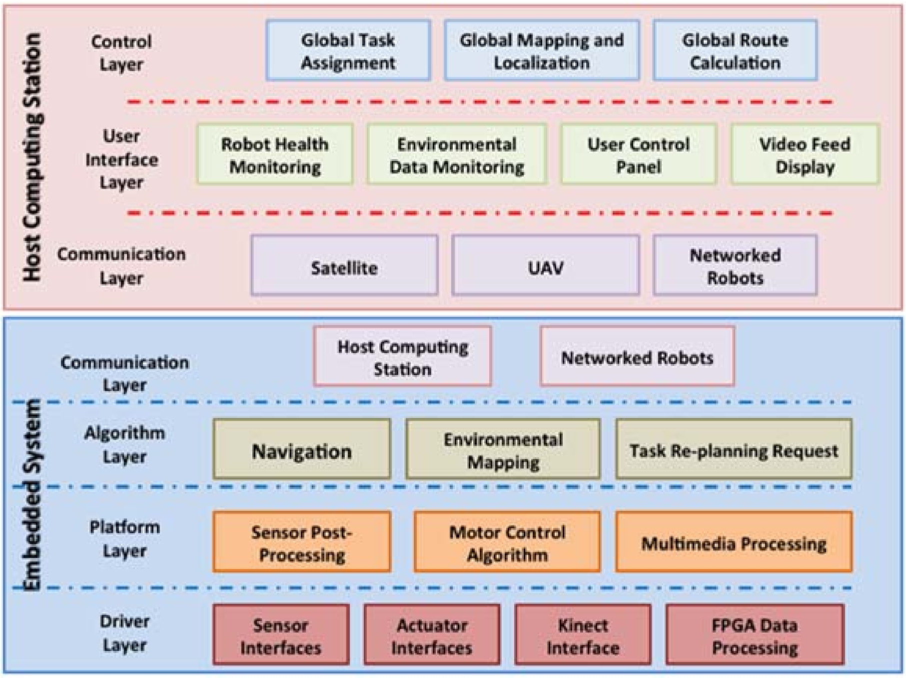

Fig. 1 System architecture.

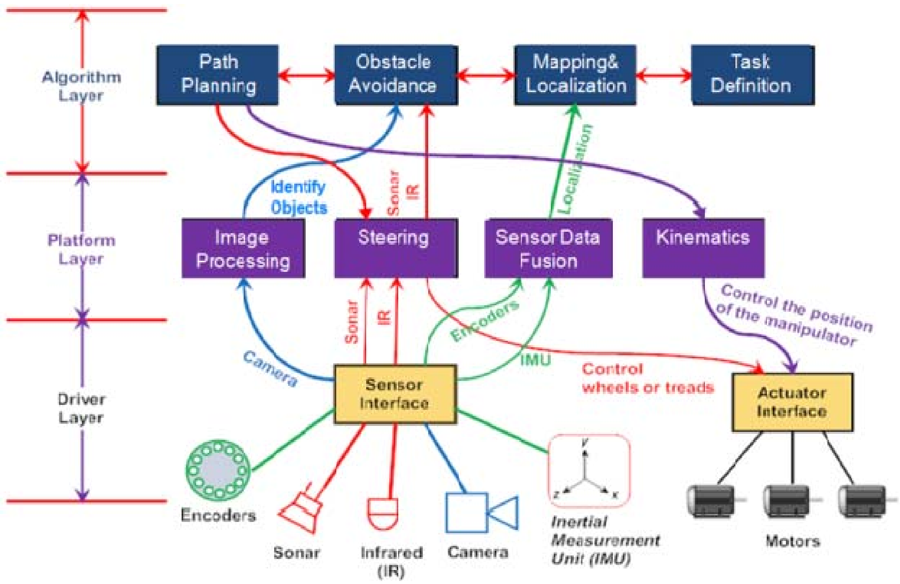

Fig. 2 Embedded hardware layers.

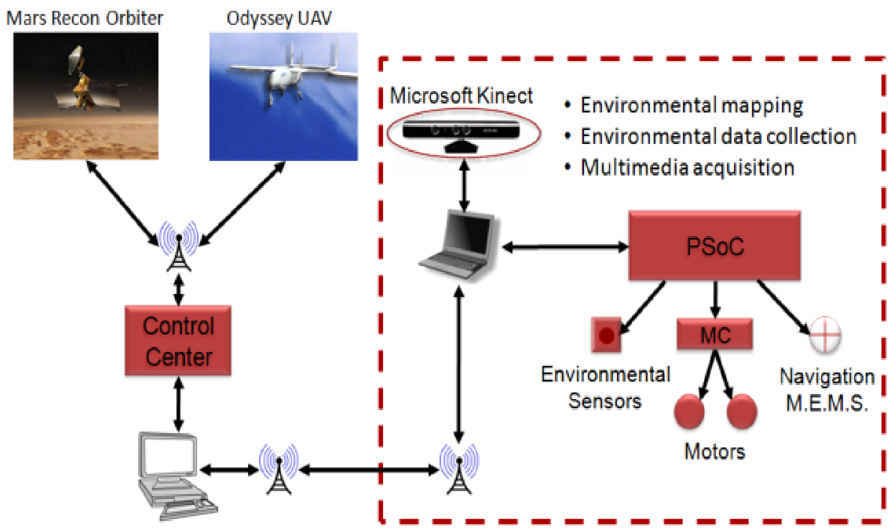

Fig. 3 System control and communication.

computer is responsible for path-finding computations.

The manufacturer of the PSoC-5 microcontroller publishes a free software suite called PSoC Creator to facilitate design implementation [5]. This software allows drag-and-drop component placement as shown in Fig. 4. Basic components include wires and sheet connectors, while more complex components include I2C Master controllers and PWM blocks. These components can be fixed function blocks comprised of dedicated hardware, or UDB components built from programmable logic arrays.

Mapping and navigating an environment requires the knowledge of robot's position. The PSoC-5 acquires data from multiple sensors to inform control algorithms providing position information to navigation software. Current design of the mobile robot incorporates range sensors, kinematic sensors, and a wireless communication device. Infrared and ultrasonic range sensors provide distance measurements between the robot and nearby obstacles. An accelerometer, compass, gyroscope, and two motor encoders convey data informing the kinetics of the robot. The accelerometer, compass, and gyroscope are

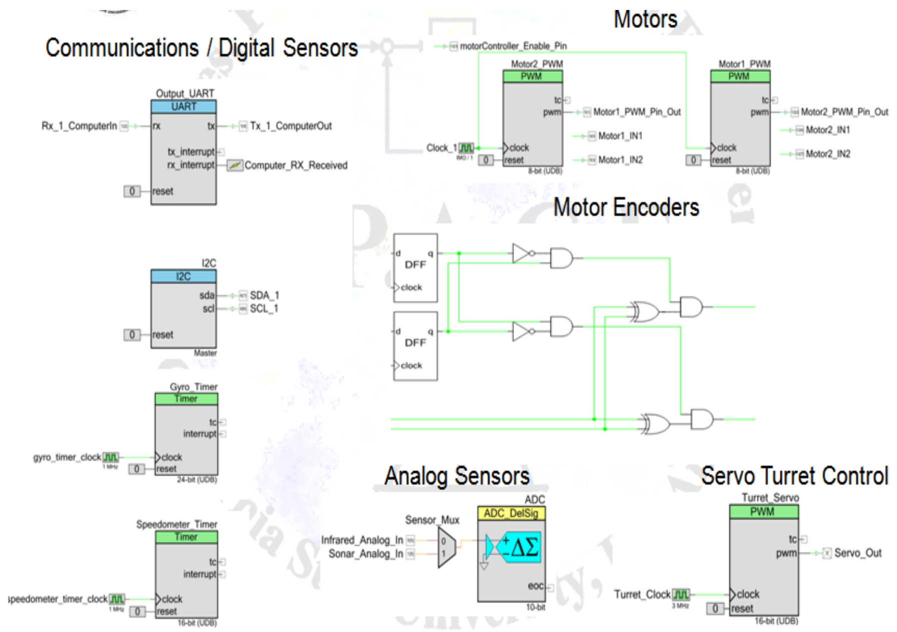

Fig. 4 Mobile robot design components in PSoC creator.

contained on an integrated circuit, forming an IMU (integrated measurement unit). The motor encoders allow motor rotations to be counted. A global positioning service device provides positional data. Communication is handled by a wireless UART IC (integrated circuit) called "Wixel". All sensors are digital except for the infrared sensor. The sonar sensor is capable of digital or analog output.

The range sensors are mounted on a servo on the top front of the robot. The IMU sensor and GPS are mounted on the top back. The left and right sides of the robot are each driven by one motor. The motor is connected via a series of gears to two wheels, which are connected by a tread. Motor encoders are attached to each motor.

4. Kalman Filter

Optimal estimators can be used to anticipate the expected sensor result from the inaccurate and uncertain observations. A Kalman Filter is a recursive optimal estimator that is applied to the robot's IMU [6].

Once the data of sensor measurements is obtained, it can be used to produce an anticipated result that should concur with the other sensor results. If the anticipated values yielded by the accelerometer differ from those measured by the gyroscope, a scalar multiple called the Kalman Gain is applied to the anticipated result.

A Kalman Filter is a recursive algorithm, which reduces error in a system by successively comparing an estimated state of a system to the actual state. Four vectors constitute the filter: (1) the estimated state; (2) the estimated covariance; (3) the innovation vector; and (4) the innovation covariance. A final element of the filter, the Kalman Gain, is determined by utilizing these vectors.

The current priori state vector is given by:

$$\bar{x}_t(-) = x_{t-1} + u \tag{1}$$

where, x_{t-1} is the previous state estimate and u is the control vector, thus each priori state at time t is the sum of the previous state and a control vector. In this application, the control vector is given in degrees per second by the gyroscope. The state vector is composed of the current roll, pitch and yaw inclinations of the IMU.

The predicted covariance vector between the accelerometer and gyroscope noise is given by:

$$\overline{C_t}(-) = C_{t-1} + E_x \tag{2}$$

where, C_{t-1} is the previous covariance vector and E_x is related to the noise generated by the gyroscope, called the process noise. Process noise is the variance of the event being checked by a measurement. In this system model, the gyroscope is assumed to be correct, but noisy. The accelerometer and magnetometer are used to measure axial rotations. The accelerometer and magnetometer results are checked against the report of the gyroscope.

The innovation vector is the difference between the measured and predicted states of the system, given by:

$$y_t = z_t - x_t(-) \tag{3}$$

where, z_t is the new measurement vector.

The innovation covariance is given by the sum of the predicted covariance and the variance in the measurement:

$$S = \overline{C_t}(-) + E_z \tag{4}$$

where, E_z is the noise generated by the accelerometer, called the measurement noise.

The Kalman Gain can now be computed, and is given by:

$$K_t = E_x \times S^{-1} \tag{5}$$

The state and covariance estimates can be updated now that this scalar is known. The covariance estimate update is given by:

$$\overline{C_t}(+) = I - K_t + \overline{C_t}(-) \tag{6}$$

Successive iterations of this algorithm will provide increasing accuracy of the filter. The state updates are given by:

$$\overline{x_t}(+) = x_{t-1}(-) + K_t y_t \tag{7}$$

This algorithm is then iterated. The covariance is also updated, and this new state prediction is used for the state prediction provided in Eq. (1).

The PSoC-5 implementation of the Kalman Filter is based on existing Arduino software [7]. The source software was written in C++, which the PSoC-5 does not support. The source software also implemented a different sensor suite than the one utilized by the robot.

The Host Computing Station depends upon information provided by the gyroscope, accelerometer, and compass to determine the location of the mobile robot at all times. Since each sensor introduces error to the system, a method of combining and filtering output is desired. A Kalman Filter provides a solution by performing data fusion and minimizes the error propagated by the sensors over time. Fig. 5 represents the algorithm.

The Kalman Filter requires two vectors and two matrices to be defined before it begins operation: the

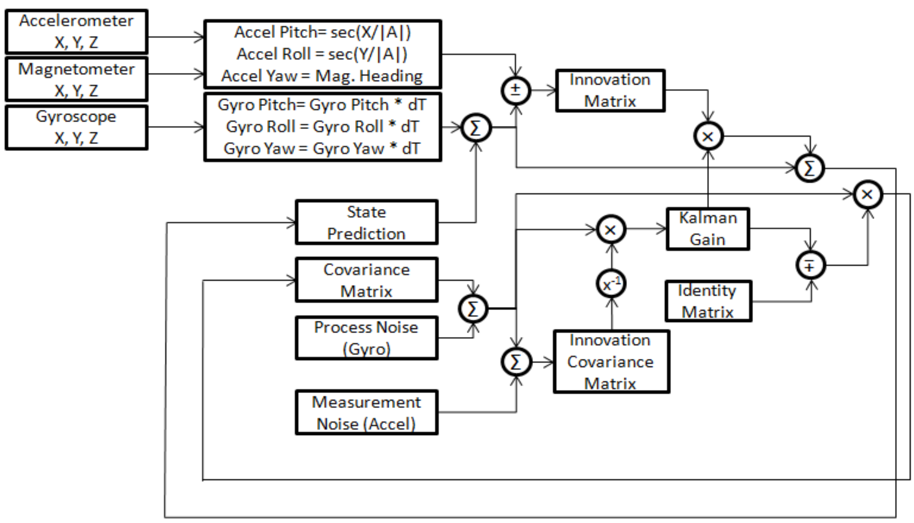

Fig. 5 Kalman filter implementation on PSoC-5.

measurement and process vectors, and the measurement and process variance matrices. The measurement vector is obtained from the accelerometer and magnetometer. The process vector is obtained from the gyroscope. The matrices are defined by the measured variance of functions of these vectors. Each vector is obtained by a series of functions that process data from the IMU. A function defined in the IMU retrieves accelerometer and magnetometer (measurement) data to obtain the current *g*-force vector. The magnitude of the result is computed, and roll and pitch are calculated trigonometrically. Yaw is computed from the magnetometer based on its tangential components of indicated directions:

$$rollAccel = \cos\frac{rollAxis}{|a|}$$

$$pitchAccel = \cos\frac{pitchAxis}{|a|} \tag{8}$$

$$YawMag = \arctan\frac{East.x}{North.x}$$

where, *East.x* and *North.x* are the components of east and north vectors provided by the magnetometer, relative to the direction of gravity and the x-axis of the IMU.

The magnetometer compass heading is passed to the yaw value for the measurement vector. Yaw is obtained using components of the magnetometer and accelerometer by calculating relative east and relative north vectors. The gyroscope yields angular velocity, and a timer is used to integrate the angular velocity over time to track the angular displacement of the IMU. The timer is reset after each computation. The product of the angular velocity, in degrees per second, and the value read by the timer since it was last reset, is the angular displacement of the IMU.

The roll, pitch, and yaw are stored into a three-by-three array floating-point array, representing a square diagonal measurement matrix. The roll, pitch, and yaw are respectively stored along the diagonal. The angular displacement is stored in another 3 × 3 array, along the diagonal. Several three by three arrays of floating point numbers are established, and initialized to zero, for each matrix needed to implement the Kalman Filter: the state estimate, covariance, innovation, innovation covariance, and Kalman Gain matrices.

The result of the measurement matrix is monitored over several thousand samples to determine the variance of each axis. The same observation is made to find the variance of the control matrix. Once the stochastic information is known, the data may be passed to the Kalman Filter.

The first operation of the filter is a check to bind the filtered result of the roll, pitch, and yaw between -180° and 180°. This is followed by the matrix operations to perform Kalman Filtering.

Once the framework is established, the Kalman Filtering algorithm is applied indefinitely. An interrupt timer is used to call the Kalman Filter, and the timer is disabled between iterations. The algorithm requires more time to execute than output data rate of the IMU, so the time between interrupts is limited only by the computational time required by other tasks.

5. Results

An apparatus built to allow three axis rotations was used to test the navigation capability of the robot and to verify localization algorithms utilizing the IMU. This testing apparatus allows independent motion along each of three axes: roll, pitch, and yaw. Three tests for each implementation of the IMU were conducted: (1) raw, integrated, gyroscope data; (2) Kalman filtered data utilizing the full IMU and (3) Direction Cosine Matrix data utilizing a PI controller and the full IMU. Each test case examines yaw information under fixed pitch positions. The IMU is affixed to a plate of wood on top of the apparatus. Wood was chosen because it will not influence magnetometer in the IMU. Fig. 6 depicts the testing configuration.

To test the varied angular data acquisition methods, the servo that controls yaw is calibrated. The duty cycle to a PWM driving the pitch servo is experimentally investigated until the values that bring the servo nearest to 90° from an initial position are found. Then the values that bring the pitch servo nearest to 20° pitch are determined. Once the servo is determined to be optimally configured, a program is run to observe the compass heading determined Kalman Filter and DCM algorithms. Values provided by the unfiltered gyroscope and magnetometer, which are also observed during the experiment.

The program measures the initial angle reported by the gyroscope. For raw gyroscope values, this is 0°. The angular rate is determined after servo rotates to 90° and is motionless. A short delay between executing servo motion allows the two filters to stabilize before initializing the next rotation. This delay is retained when measuring the raw values to maintain identical test conditions between each experiment. Results are analyzed in terms of mean angle traversed in the clock-wise and counter-clockwise direction, and the mean heading in each direction. The mean angles and standard deviations are given for each experiment.

The servo calibration found the angle of rotation nearest 90° to be 90°± 2°. The pitch angles of the platform were determined to be zero, pitched down 20°, and pitched up 20.25°. At each inclination, the yaw servo was rotated clockwise 100 times and counter-clockwise 100 times. The resulting measure at the end of each rotation was tabulated. When using the Kalman Filter, DCM, and only the magnetometer, the measure reported is the compass heading. When using gyroscope values, the measure reported is relative to the initial yaw of the sensor. The result data are presented in Table 1 and include the counter-clockwise measurement, clockwise measurement, and the difference between each measurement. Fig. 7 shows

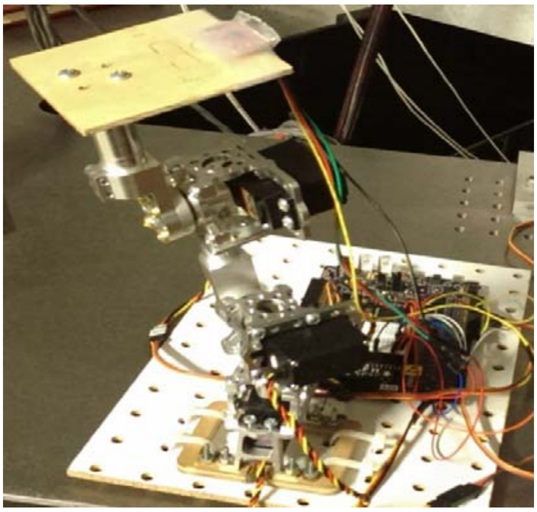

Fig. 6 Test apparatus with IMU attached.

Table 1 Heading and arc measurement results.

Pitch	Kalman	°CW	°CCW	dθ
20°	μ	-40.81	39.30	80.10
	σ	0.54	0.67	0.43
0°	μ	-41.48	46.82	88.28
	σ	0.74	0.79	0.38
-20°	μ	-40.17	42.95	83.10
	σ	0.67	0.68	0.45
	Gyro:	°CW	°CCW	dθ
20°	μ	-0.59	81.02	81.60
	σ	0.80	0.34	0.84
0°	μ	0.26	89.20	88.95
	σ	0.54	0.45	0.56
-20°	μ	0.26	83.80	83.54
	σ	0.66	0.51	0.68
	Mag:	°CW	°CCW	dθ
20°	μ	-51.57	56.43	107.99
	σ	3.43	8.41	9.17
0°	μ	-49.86	51.45	101.33
	σ	2.84	3.44	4.26
-20°	μ	-45.09	47.35	92.45
	σ	2.73	3.84	4.92

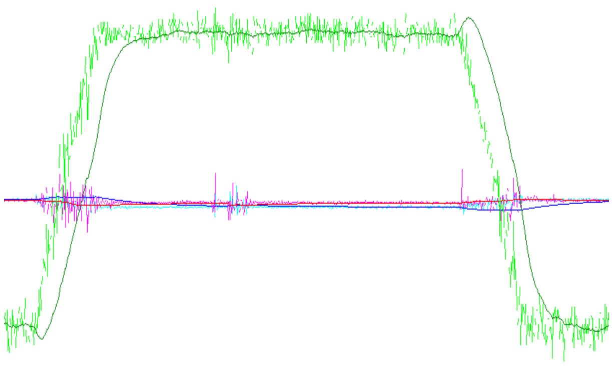

Fig. 7 Magnetometer-only (light green) versus Kalman filtered (dark green) data for yaw rotation.

the magnetometer noise removal on the filtered yaw axis.

The Kalman Filter exceeds the precision of the arc measured by the gyroscope at all inclinations of pitch. Heading information is gained by incorporating the magnetometer, but accuracy is lost without tilt compensation. The arc measured by the filter follows that measured by the gyroscope. The control vector given to the filter is derived from the gyroscope, while the measurement vector is derived from the magnetometer. Since the variance of the magnetometer is greater than that of the gyroscope, the filter favors the results provided by the control vector once the measured state is near the estimated state. The theoretical Kalman Filter anticipates this result.

6. Conclusion

This implementation of a Kalman Filter provides precise measurements to be obtained by a PSoC-5 from an IMU. The Kalman Filter improves the reliability of the gyroscope when deployed on level surfaces and heading information is gained by its implementation. The accuracy of the filter would be improved if a method of tilt compensation was applied.

Future work includes incorporating this Kalman Filter into a DCM. A DCM tracks the roll, pitch, and yaw of an IMU with respect to an external and consistent inertial reference frame. A DCM provides accurate heading information when used with an IMU and would benefit from the precision delivered by the Kalman Filter.

References

[1] Powell, M. W., Crockett, T., Fox, J. M., Joswig, J. C., and Norris, J. S. 2006. "Targeting and Localization for Mars Rover Operations." Presented in the Information Reuse and Integration 23: 16-18. doi: 10.1109/IRI.2006.252382.

[2] Kriechbaum, K. L. PhD Dissertation: "Tools and Algorithms for Mobile Robot Navigation with Uncertain Localization" URL: < http://thesis.library.caltech.edu/2363/1/kriechbaum-thesis .pdf>.

[3] Stentz, A. 1996. "Map-Based Strategies for Robot Navigation in Unknown Environments." In *proceedings of the AAAI Spring Symposium on Planning with Incomplete Information for Robot Problems*.

[4] PSoC ® 5: CY8C55 Family Datasheet URL: http://ecologylab.net/courses/sensoryInterfaces/resources/CY8C55DataSheet.pdf .

[5] PSoC ® Creator ™ Quick Start Guide URL: http://www.cypress.com/index.cfm?docID=45972.

[6] Kalman, R. E. "A New Approach to Linear Filtering and Prediction Problems Transactions of the ASME." *Journal of Basic Engineering* 82 (Series D): 35-45.

[7] Kammerlen, L. 2012. Kalman Filter Tutorial, 9DOF Accelerometer, Gyro and Compass. Jaycon Systems Projects Forum. Retrieved February 27, 2014, URL: http://www.jayconsystems.com/forum/viewtopic.php?f=22&t=40.