

Mighty Eagle: The Development and Flight Testing of an Autonomous Robotic Lander Test Bed

Timothy G. McGee, David A. Artis, Timothy J. Cole, Douglas A. Eng, Cheryl L. B. Reed, Michael R. Hannan, D. Greg Chavers, Logan D. Kennedy, Joshua M. Moore, and Cynthia D. Stemple

APL and the Marshall Space Flight Center have been working together since 2005 to develop technologies and mission concepts for a new generation of small, versatile robotic landers to land on airless bodies, including the moon and asteroids, in our solar system. As part of this larger effort, APL and the Marshall Space Flight Center worked with the Von Braun Center for Science and Innovation to construct a prototype monopropellant-fueled robotic lander that has been given the name *Mighty Eagle*. This article provides an overview of the lander's architecture; describes the guidance, navigation, and control system that was developed at APL; and summarizes the flight test program of this autonomous vehicle.

INTRODUCTION/PROJECT BACKGROUND

APL and the Marshall Space Flight Center (MSFC) have been working together since 2005 to develop technologies and mission concepts for a new generation of small, autonomous robotic landers to land on airless bodies, including the moon and asteroids, in our solar system.¹⁻⁹ This risk-reduction effort is part of the Robotic Lunar Lander Development Project (RLDDP) that is directed by NASA's Planetary Science Division, Headquarters Science Mission Directorate. During this ongoing collaboration, APL has led development of several subsystems, including guidance, navigation, and control (GNC); flight software; and mechanical design. MSFC has led development of several subsystems, including propulsion and power systems. A variety of

technology risk-reduction efforts, illustrated in Fig. 1, have been performed to explore technologies to enable low-cost missions.

As part of this larger effort, MSFC and APL also worked with the Von Braun Center for Science and Innovation (VCSI) and several subcontractors to construct the *Mighty Eagle*, a prototype monopropellant-fueled robotic lander. The primary objective for the lander was to test and mature the GNC system for the final 30–50 m (98–164 ft) of the descent stage before landing. The prototype development effort also created an integrated team of engineers at MSFC and APL who could quickly transition to potential space missions. The lander has proven itself as a test bed to provide hands-on

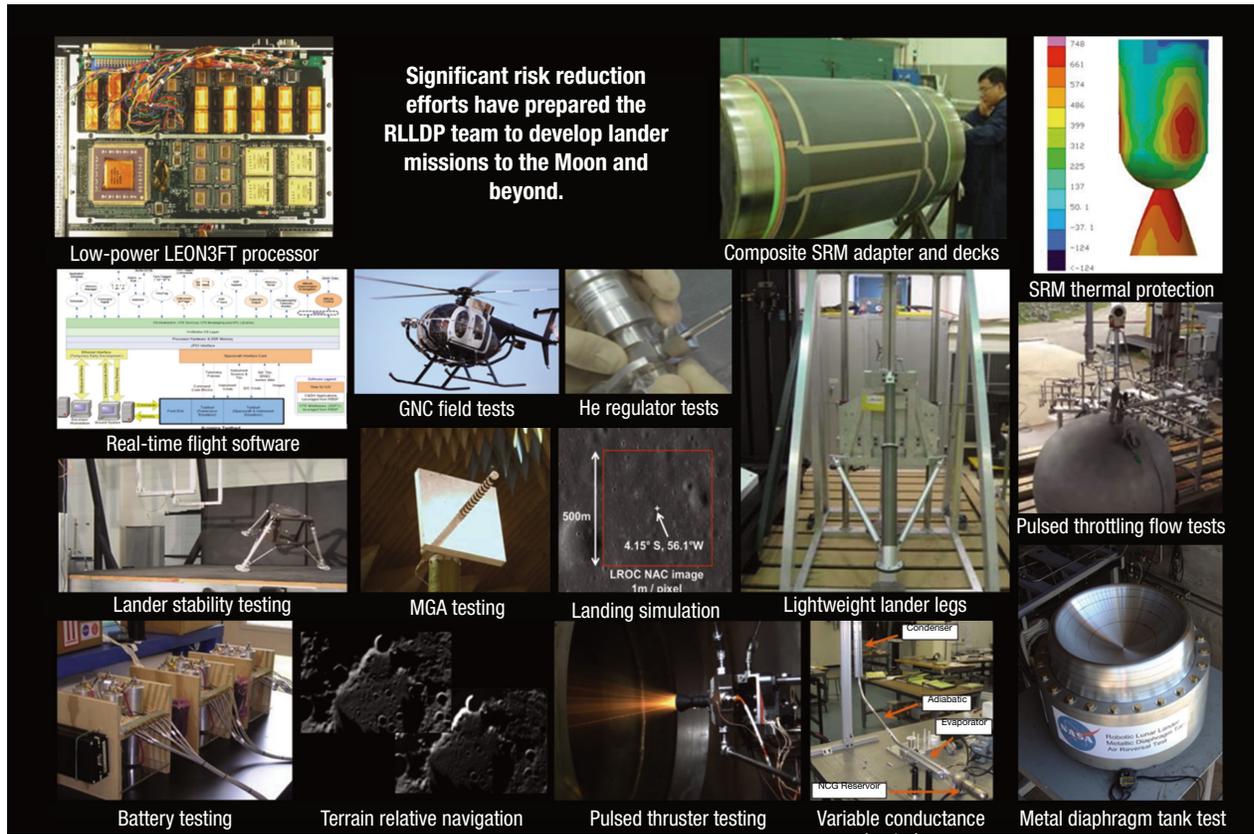


Figure 1. Risk reduction activities on RLLDP. MGA, multiple gain antennae; NCG, noncondensable gas; SRM, solid rocket motor.

experience to young engineers and demonstrate other technologies, including optical navigation.

This article provides an overview of the prototype lander, with an emphasis on its GNC system, and outlines the testing performed on the lander both leading up to and including multiple flight test campaigns. The article is divided into three main sections. The first section provides an overview of the vehicle hardware, including actuators and sensors. The second section describes the software architecture and algorithms used by the GNC subsystem, including several optical navigation systems. The final section summarizes the various simulation, processor-in-the-loop, and flight tests that were performed on the prototype lander.

VEHICLE HARDWARE DESCRIPTION

The prototype lander has a three-legged configuration, as shown in Fig. 2. The footpads of the three legs form a triangle with sides 2.13 m (7.0 ft) long. The distance between the two decks is 0.61 m (2.0 ft). The dry mass of the vehicle is approximately 206 kg (454 lbm), and it can fly with 116 kg (256 lbm) of propellant and 7 kg (15 lbm) of pressurant. This allows a maximum flight time of approximately 45–50 s with 5% reserves at touchdown.

Propulsion System

A blowdown 90% pure hydrogen peroxide monopropellant propulsion system that is pressurized using regulated high-purity nitrogen provides actuation for both the attitude control system (ACS) and the descent control systems. Hydrogen peroxide was chosen for the prototype system because its decomposition byproducts, steam and oxygen, are both nontoxic, and it provides sufficient energy density to achieve the target flight times. The propulsion system, built by Dynetics in collaboration with MSFC, APL, and VCSI, feeds 16 monopropellant thrusters: twelve 44.5 N (10 lbf) attitude control thrusters, three 267 N (60 lbf) descent engines, and a throttleable engine with a maximum thrust of approximately 3114 N (700 lbf). The thruster configuration is illustrated in Fig. 3. The 12 attitude thrusters are grouped into six coupled pairs to allow torque to be applied independently to each of the three rotation axes of the vehicle. The three fixed descent engines provide the vertical thrust to control the vehicle’s altitude and descent rate. The large throttleable engine provides Earth gravity cancellation (EGC). The EGC engine nominally produces a thrust of five-sixths the weight of the lander throughout the flight to approximately simulate lunar gravity for the rest of the system by nulling the difference between Earth and lunar

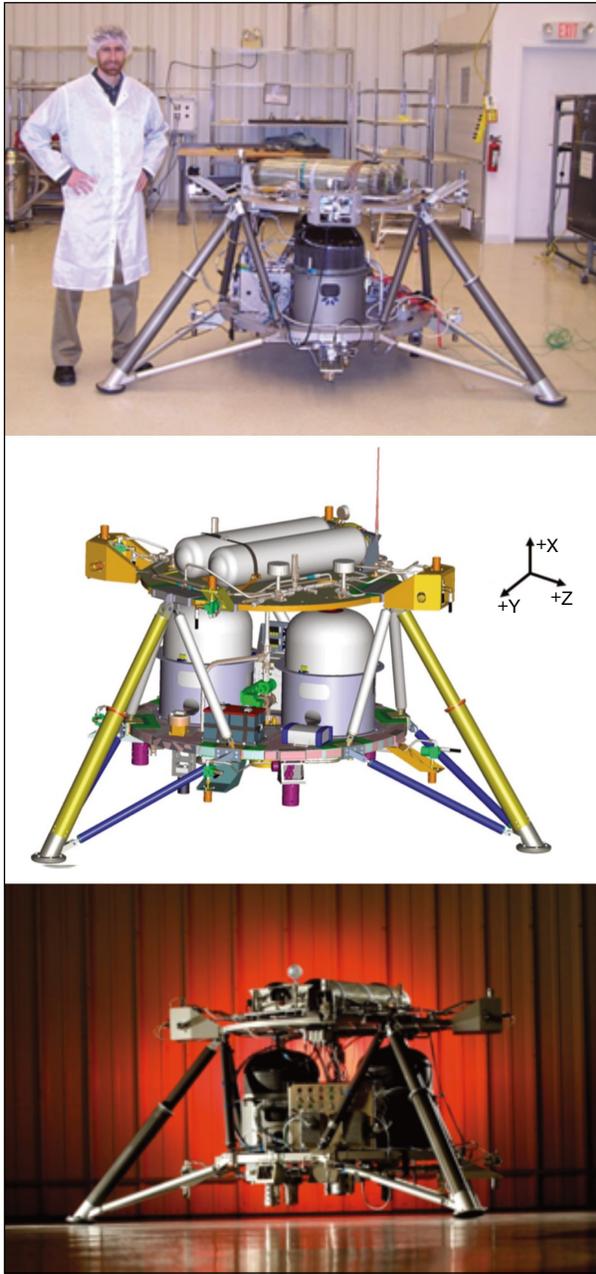


Figure 2. Mighty Eagle prototype lander. (NASA)

gravity. A fixed EGC engine was chosen over a gimbaled design to minimize system complexity, cost, and schedule constraints.

Sensors

The GNC sensors, shown in Fig. 4, were selected to provide flight-like feedback to ensure applicability to a lunar mission and include an inertial measurement unit (IMU), radar altimeter, optical camera, and ground contact sensors.

The LN200 IMU¹⁰ provides angular rates and three-axis accelerations at 400 Hz that are used to propagate

the vehicle attitude, velocity, and position. The Northrop Grumman-produced LN200 was chosen because the non-space-qualified version used on the Mighty Eagle is an affordable solution that meets the performance requirements, while the space-rated version offers a potential solution for an actual lunar mission.

The Type 2 Miniature Radar Altimeter (MRA) provides vertical position from 0.2 to 100 m (0.66–328 ft), covering the full altitude range of the Mighty Eagle.¹¹ The Roke-produced MRA provides a low-cost and low-power option for the prototype. For actual lunar missions, several sensors, including a Honeywell HG8500 series altimeter¹² and the APL-developed miniature lidar altimeter, are under consideration. The Honeywell HG8500 has heritage with Mars landing missions, and the lidar altimeter is being developed as a low-power, low-mass, and low-cost alternative for a variety of planetary missions. Parallel testing of these sensors was conducted during a helicopter field test as part of the larger RLLDP.

A commercially available digital camera, the illunis RMV-4201 with an Active Silicon Phoenix D48CL Frame Grabber, allows testing of optical navigation algorithms on the lander. The images from this nadir-facing camera can be used to derive lateral velocity and terrain relative position. This camera provides representative images for developing the algorithms but would be replaced with a space-qualified sensor for the lunar missions.

Contact switches mounted on the main pivot point of the legs provide a positive indication that the vehicle is on the ground and can, along with other inputs, safely terminate the propulsion system.

A NovAtel ProPak global positioning system (GPS) receiver is included on the vehicle, although its data are not made available to the primary GNC system because no equivalent sensor is available on actual lunar flights. The GPS information is logged for postprocessing and also used by onboard autonomy software that can declare autonomous abort sequences if the vehicle exceeds predefined flight envelopes.

Processor

The lander avionics unit (LAU), developed by Southwestern Research Institute (SwRI), performs the onboard processing, including processing GNC sensor data, running GNC algorithms, performing data logging, and running optional optical navigation algorithms. The LAU includes a flight-like BAE Systems RAD750, a common processor on space missions, with 256 MB of memory running at 133 MHz. The avionics could support alternate single-board computers when faster models are developed and qualified. Although this processor has a low rate compared with modern non-space-qualified processors, it was chosen to represent

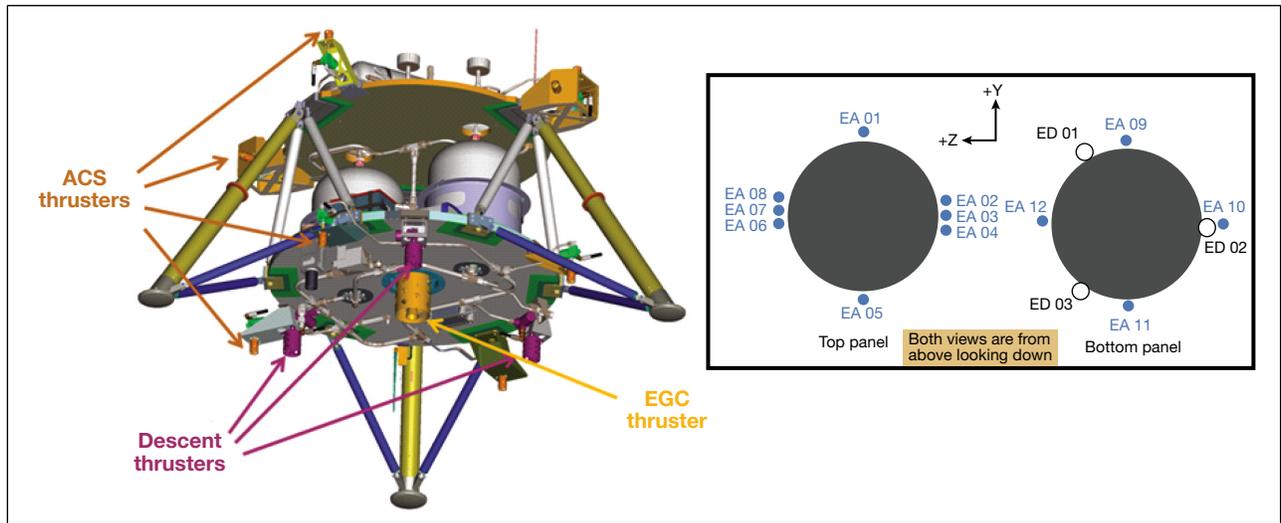


Figure 3. Thruster configuration. EA, attitude engine; ED, descent engine. (NASA)

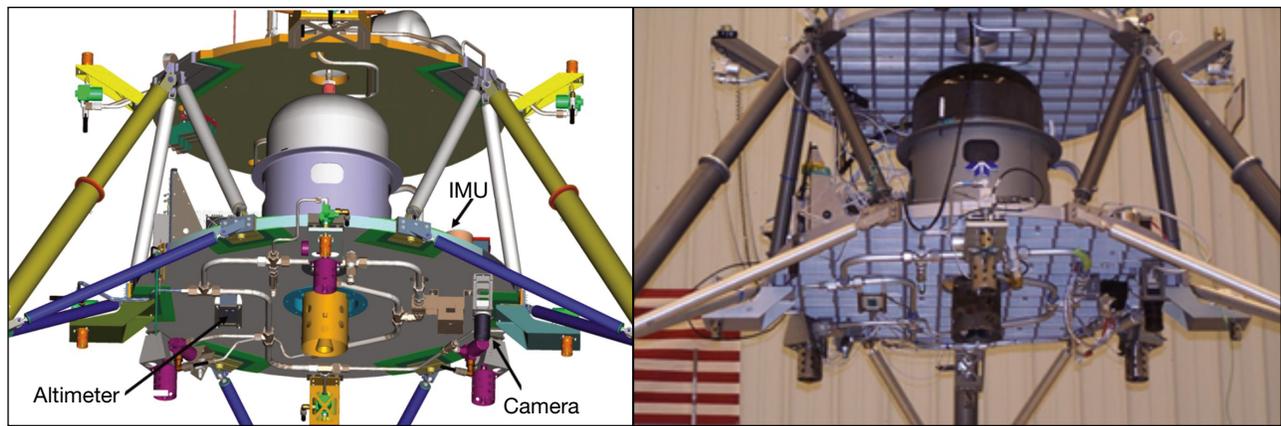


Figure 4. Sensor configuration. (NASA)

currently available flight processors for actual space missions. A large multi-megabyte memory card was also added to the LAU to perform data archiving of GNC flight data, various temperature and pressure measurements, and acquired images.

Structures and Legs

The majority of the vehicle structure, including the two circular deck plates, the legs, and the propellant tanks, is constructed of aluminum and composite materials. All lander materials were chosen to meet compatibility requirements with hydrogen peroxide.

The circular decks provide both a large part of the vehicle structure and protection for the high-price components of the vehicle, including the LAU, the IMU, and the propellant tanks, which are located between the decks. Two independent sets of decks were designed and built for the lander. The first set was designed by using a composite/aluminum honeycomb sandwich to provide

the team with experience working with these materials, which could be used to reduce structure mass on an actual space-flight vehicle. A second pair of aluminum decks were also designed and manufactured in-house at MSFC. Mechanical and thermal modeling and analysis were performed on both sets of decks, although the aluminum decks were installed on the prototype because they were available at an earlier date.

Mechanical engineers at APL designed and tested shock-absorbing legs for the prototype. The damping mechanisms include both a hydraulic telescoping damper for the nominal loading conditions and a single-use, crushable honeycomb for larger loads. Through experience gained through the leg testing process, the APL team was able to refine the leg design, removing approximately 13 kg (29 lbm) of mass per leg, corresponding to a reduction of leg mass by one third. The design was highly successful, accommodating the harsh testing environments and varying landing conditions.^{13,14}



Figure 5. Tether system for initial shakeout flights. (NASA)

To perform initial shakeout flights, APL engineers also designed a tether system, shown in Fig. 5, to constrain vehicle motion. The tether geometry allows adequate translation and attitude motion of the vehicle to perform short vertical flights approximately a foot off of the ground, while preventing the vehicle from tipping over in the event of a malfunction. Each of the three tethers connects a lander foot to the ground and consists of steel cable inline with a Miller SofStop Shock Absorber. These off-the-shelf shock absorbers, nominally used as part of equipment to protect workers from falls, provide damping to absorb any kinetic energy of the vehicle if the tethers are pulled taught. Additional steel cables, in parallel with the shock absorbers, limit the absorbers' maximum travel. This innovative, low-cost system allows lower-risk testing of the vehicle after any design modifications or maintenance.

AUTONOMOUS GNC SYSTEM

The onboard flight software, including the GNC algorithms, was designed to match that of an actual lunar lander vehicle as closely as possible within the time and schedule constraints of the program. The flight software and GNC subsystems are responsible for processing sensor data, estimating the vehicle state, and commanding thrusters control the vehicle along the desired trajectory.

Software Architecture

The software architecture is built around the core flight executive (cFE) modular software environment developed by NASA Goddard Space Flight Center.¹⁵ The cFE provides abstraction layers and multiple key services, shown in Fig. 6,

including board initialization, event logging, and a software bus, to promote reusability across processors and missions. Additional modular applications are added as needed to handle various vehicle tasks including GNC, optical navigation algorithms, sensor input preprocessing, actuator control, and data logging. All communications between applications are messages handled in a “publish and subscribe” manner. Applications subscribe to individual messages but are unaware of the publishers of these messages. Applications publishing messages are unaware of the recipients, if any, of these messages. Isolating module interaction to message passing allows applications to be swapped out with no effect as long as the published messages remain constant. This modular architecture allowed incorporation of various software components from lander team members at different locations. For example, engineers at APL designed the primary GNC module, GNCA, and an optical navigation module to estimate lateral velocity from images. Engineers working at VCSI and MSFC designed the backup GNC module, GNCB, a control module to drive the throttleable EGC engine, and the vehicle state man-

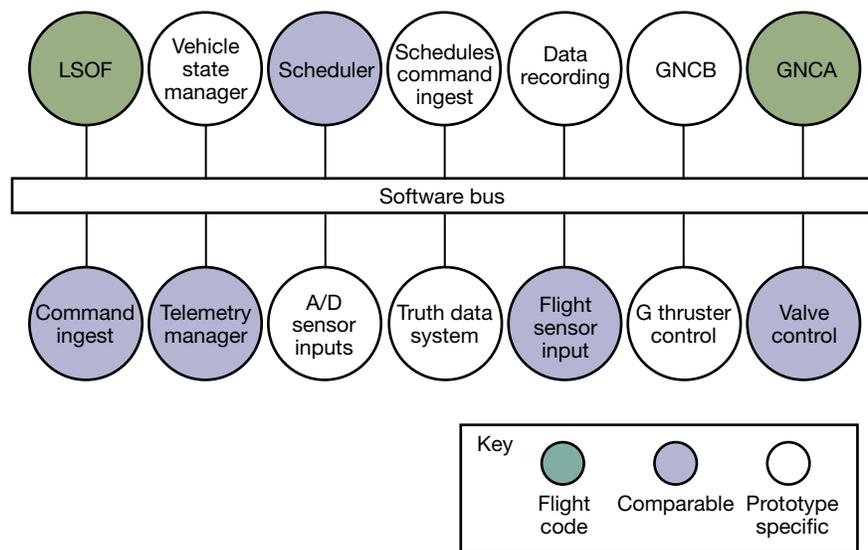


Figure 6. Modular software architecture.

ager that performs all sequencing operations, ensures a unique commanding source between the primary and backup GNC systems, and initiates autonomous soft-abort sequences if the GNC navigation states or commands exceed predefined thresholds. These components were seamlessly integrated into the final system by using the cFE architecture.

Ground System and Vehicle Commanding

The lander team chose the APL-developed Mission Independent Ground System (MIGS) software for command and control of the vehicle. MIGS, developed for the Van Allen Probes mission, incorporates the commercial L-3 Telemetry West InControl core command and telemetry software. Selecting MIGS for vehicle operations allowed the lander team to develop expertise and experience with the same ground system that could be used for an actual robotic lunar lander mission. The ground system was designed around two console positions, Command (CMD) and Data (DAT), which would be manned by mission operations personnel during flights. CMD uplinked the GNC sequences to the flight computer and issued commands to initiate flight software transitions, open or close propulsion system valves, and power on avionics equipment. DAT monitored and reported vehicle telemetry and was responsible for making telemetry-based manual flight abort calls. The team also developed flight rules and predefined responses for off-nominal events that might occur during the autonomous phases of the flight.

The vehicle trajectory for each flight was constructed by using a sequence of position and velocity commands given to the lander in a topocentric coordinate system with its origin at the takeoff location. Once the flight sequence was initiated, the lander autonomously performed a thruster warm-up sequence and navigation system initialization before autonomously executing the commanded maneuver sequence. Maneuvers were nom-

inally performed by commanding an open-loop constant velocity command in the desired direction of motion followed by a position command to hold the desired final position. The vertical and lateral control modes were decoupled and could be set independently in either position or velocity mode. This setup allowed four discrete translation control modes to be commanded, as illustrated in Table 1.

Guidance, Navigation, and Control

The algorithms and software for the GNC systems,¹⁶ with exception of the image processing algorithms, were designed using MathWorks' Simulink and Embedded MATLAB software packages. The algorithms were first created and tested entirely within the Simulink simulation environment. The GNC engineers then used the Real-Time Workshop package by MathWorks to auto-generate C code to execute the GNC system.¹⁷ The autogenerated algorithm code was functionally wrapped with hand-generated code to handle all application and task creation, initialization, and interfaces with the cFE communications. The wrapper code takes all input data packets and converts them into the format and units expected by the GNC blocks, executes the GNC blocks at the appropriate times, and converts the output of the GNC blocks into telemetry packets that are published at the desired rates. The majority of the GNC code on the flight system, including state propagation, guidance, and thruster commanding, runs at 50 Hz. A second 1-Hz navigation filter incorporates lower-rate sensor data including altimeter measurements and optical measurements.

The 50-Hz input data consist of sensor data, desired state commands, and messages from the image processing system. The IMU data contain buffered 400-Hz data from the previous 20-ms window and are first passed through a preprocessing module that performs data checking, removes known biases, and outputs average angular rates and translational accelerations rotated into the vehicle body frame. These data are then sent to the attitude determination system (ADS) and the navigation filter, which estimate the vehicle state. The commanded lateral vehicle state is used by the lateral guidance module to determine the commanded vehicle orientation. This desired attitude command, along with vertical motion commands, is sent to the control module that determines the appropriate thrusters to fire. The architecture of the GNC software is illustrated in Fig. 7.

ADS and Navigation Filter

Before takeoff, the IMU is used to determine the initial vehicle attitude relative to the topocentric frame. Accelerometer data are used to estimate the ground normal force and, thus, the vehicle attitude relative to vertical. The magnitude of the sensed normal

Table 1. Translation control mode

		Vertical Guidance Mode	
		Position	Velocity
Lateral Guidance Mode	Position	Hover 	Vertical ascent and descent 
	Velocity	Lateral translation 	Descent and ascent with lateral translation 

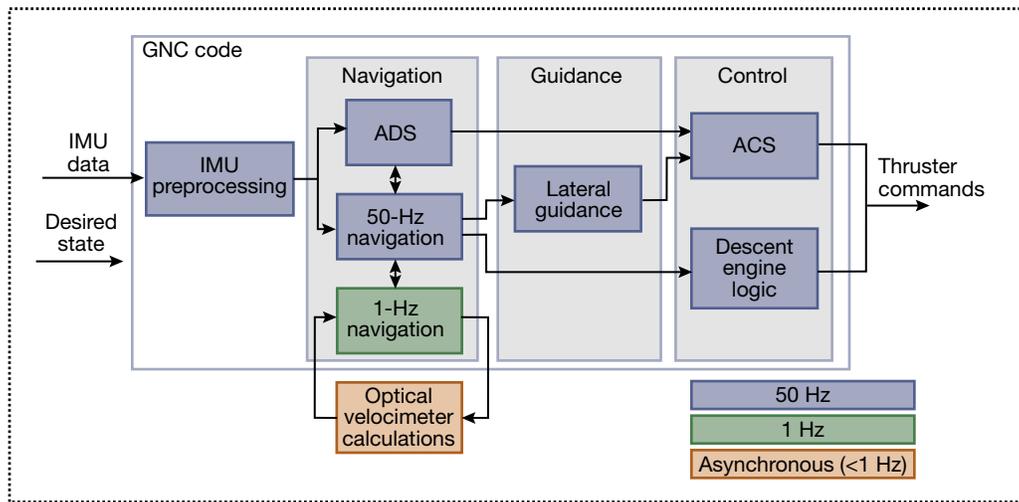


Figure 7. GNC layout.

force is used to estimate accelerometer bias along the normal force direction. During this initialization, the gyroscope measurements are also used to measure the direction of Earth's rotation. These data are provided to the navigation system to compensate for the rotation of Earth relative to the inertial frame. Once the flight sequence begins, the ADS can propagate only the initial vehicle orientation by using measured angular rates from the IMU and the estimated Earth rotation rate. On a real mission, the attitude determination would also incorporate attitude measurements from an onboard star tracker.

The translation navigation filter is decoupled from the attitude filter and consists of two parts: a fast 50-Hz propagator and a 1-Hz filter. The fast component of the filter propagates the position and velocity by using an onboard gravity model and measured accelerations from the IMU that are rotated using the attitude estimate from the ADS. Other measurements from the altimeter and optical navigation system are buffered along with the IMU data and sent to the 1-Hz navigation filter that incorporates them using Kalman filtering techniques. The 1-Hz filter maintains a covariance estimate for the vehicle state and provides state corrections to the 50-Hz propagator. This approach, using filters with multiple rates, is based on heritage from previous APL spacecraft that used a similar approach for incorporating star tracker measurements in the ADS and allows for measurements that are either out of sequence or delayed within the 1-s window to be time-ordered and processed. This approach also allows the computationally expensive steps of the navigation filter to be run at lower priority and spread over a larger time frame to even out computational loading.

Because optical navigation measurements can take multiple seconds to process, they are incorporated into

the navigation filter by using state augmentation and fixed-point smoothing.¹⁶ Although the actual image processing algorithms can take several seconds to run, the fact that images are taken is known immediately. At the time each image is taken, the navigation filter state is augmented to include the position of the vehicle when the image was taken in addition to the vehicle position at the current time. When an optical measurement arrives after computation processing is completed, it contains information about the difference in the vehicle location at two times in the past. By using augmented states, this measurement can be treated by using a standard Kalman filter where the measurement is not dependent on the current vehicle state but on the augmented states.

Lateral Guidance

The lateral position and velocity of the vehicle are controlled by tilting the vehicle to apply a portion of the vertical descent engine thrust in the lateral direction. The inputs to the lateral guidance law are the commanded position and velocity in the topocentric frame and the estimated vehicle position, velocity, and acceleration. A weighted sum of the components of the position error, velocity error, and acceleration that are perpendicular to the local vertical are used to define a total lateral error. A correction to the vehicle attitude is calculated to tilt the vehicle relative to vertical to provide a lateral force proportional to this lateral error up to a saturation limit. This attitude correction is also fed through a low-pass filter to prevent high-frequency changes in the commanded attitude. This attitude correction relative to vertical is combined with the nominal commanded attitude that defines the roll of the vehicle around the local vertical to determine the total commanded attitude.

Thruster Control

The thruster control logic is divided into three discrete parts: the ACS, the descent engine control, and EGC control. The ACS determines how to fire the small attitude control thrusters to achieve the desired attitude. The descent engine logic determines when to fire the three descent engines to track the vertical position and velocity commands. The EGC control system throttles the EGC to target a thrust equal to five-sixths of the Earth weight of the vehicle.

The lander ACS is based on phase plane logic similar to that used on previous APL missions, including STEREO (Solar TERrestrial RELations Observatory). The direct inputs to the ACS are the commanded and estimated vehicle quaternion, the commanded and estimated angular velocities, and the angular acceleration that is estimated by filtering the measured velocity. The commanded quaternion is calculated by the lateral guidance law, and the commanded angular velocity is always set to zero on the prototype. The angular error is also integrated to allow integral control in the ACS to mitigate the effects of one-sided dead banding resulting from constant disturbance torques on the vehicle. A weighted sum of the angular position, velocity, acceleration, and integrated angular error is calculated and projected onto the thrust axis of each of the ACS thrusters. For each thruster, if the projected error exceeds a threshold, that thruster is commanded to fire. Additional logic in this phase plane control limits the maximum angular rates, either by always commanding a thruster to fire if the angular rate is below a negative threshold in its thrust direction or by preventing firing if its angular velocity exceeds a second threshold.

The three descent engines are controlled in unison by a single vertical fire command. The inputs to this logic block are the commanded vertical position and velocity, the estimated vehicle position and velocity from the navigation system, and the commanded input mode for either vertical position or velocity mode. When the vehicle is in vertical position control mode, the vertical velocity command is forced to zero. When the vehicle is in vertical velocity mode, the vertical position error is nulled. The vertical position and velocity errors are sent into phase plane logic that compares a weighted sum of these errors with a threshold. Hysteresis is added to the system by changing the threshold on the basis of whether the thrusters are currently firing. This hysteresis prevents rapid pulsing of the descent engines. Similar to the ACS control, additional logic limits the magnitude of the vertical velocity by always commanding a fire if the vehicle is descending too quickly or preventing firing if the vehicle is ascending too quickly.

Because an actual lunar lander would not have a thruster to cancel the difference between lunar and Earth gravity, the EGC control is performed outside of the primary GNCA control by the GNCA software

block. This allows the GNCA algorithms to control a lander experiencing simulated lunar gravity. The EGC control module uses a lookup table to determine the throttle valve position on the basis of the estimated mass of the vehicle that is calculated from the initial mass and the estimated mass of the propellant used.

Optical Navigation

Several optical navigation strategies to estimate ground relative velocity, demonstrate autonomous rendezvous and capture, and identify landing hazards have been explored using the Mighty Eagle lander. The optical navigation algorithms tested on the lander are designed to run at the lowest priority level to prevent interference with the flight control algorithms and safety algorithms that determine abort criteria.

As part of the larger RLLDP effort, APL has continued to expand its expertise with optical navigation algorithms to estimate both terrain relative position and velocity by using passive optical cameras. These efforts have built on existing APL experience and algorithms from past studies for small body and lunar landings.^{18–23} To demonstrate the feasibility of these algorithms on flight computers as part of a complete navigation package, during 2010–2011, the team integrated a software module that performs one of its algorithms, Least Squares Optical Flow (LSOF), to estimate lateral velocity with a downward-looking camera on the Mighty Eagle. This type of capability offers as a low-mass and low-cost alternative to Doppler velocimeters. The LSOF algorithm¹⁶ uses the common image processing technique of gradient-based optical flow to calculate warping matrices between two successive images. The algorithm first uses the estimated position and orientation of the camera at each image to determine an initial warping estimate. The optical flow algorithm then uses the image gradients and changes in intensity between the two images to calculate a correction to this warping. Several versions of the algorithm were explored that use different numbers of degrees of freedom for the correction warping. One version calculates an eight-degree of freedom correction to the warping. Assuming a planar surface, this warping allows estimation of the change in relative position and orientation of the camera between the two images and also the normal of the ground plane. A modified three-degree of freedom version estimates only the change in position, assuming the attitude is adequately known from independent sensors. Offline testing of the LSOF algorithm required approximately 3 s on a MCP750 running at 233 MHz to calculate a solution. While also running the GNC software on the same processor, the LSOF algorithm ran at a lower priority and required only 1 s longer. During tests on the Mighty Eagle vehicle, with a lower clock speed of 133 MHz and additional software modules to perform

extensive data logging that would not be present on an actual mission, the LSO algorithm required up to 11 s to calculate a solution.

In 2012, an Autonomous Rendezvous & Capture (AR&C) algorithm was added to the flight software by engineers at MSFC. AR&C is a cFE application that was developed using Embedded MATLAB and blocks from the Computer Vision System Toolbox from MathWorks. MathWorks' Real-Time Workshop was used to auto-generate C code for integration into the flight software. The AR&C algorithm identifies a known target of four white circles in the landing area. After image acquisition of a 768×768 image, the image is thresholded into a binary image. Blob analysis is used to determine the locations and properties of the white blobs against the black background. The target is located by matching the blobs to the expected target shape. A Newton–Raphson iterative algorithm is then used to determine the target location by comparing the shape of the target in the image with the known image pattern. Finally, the determined target position is used to determine a guidance position command to provide to the GNCA algorithm to land on the target. After the AR&C algorithm was tuned for efficiency, it was able to generate one solution approximately every 1.7 s. This was more than adequate to demonstrate the concept.

Currently, during 2013, the Mighty Eagle team is working to demonstrate hazard avoidance. For this effort

the illunis camera will be replaced by a stereo camera. Hazards such as boulders, slopes, and craters will be recognized from the stereo image, and safe landing sites will be identified. Hazard avoidance is much more computationally intensive than AR&C, so a dedicated image processor is needed. A commercially available laptop, the Panasonic Toughbook 31, will be installed on the lander as the image processor. The laptop will run cFE under Linux and will acquire the stereo images, generate the disparity map, find hazard-free landing sites, output the calculated landing site coordinates to GNCA, and log data. The laptop will communicate with the cFE applications on the SwRI avionics unit via the cFE software bus network. To the best of our knowledge, this will be the first time that the cFE software bus network will be used in flight.

PREFLIGHT TESTING AND PREPARATIONS

Before the actual flight tests of the Mighty Eagle lander, a variety of tests of the hardware and software systems of the vehicle, shown in Fig. 8, were performed to ensure proper functioning of the system. Before integration with the vehicle structure, acceptance tests of the propulsion system were performed to evaluate both individual thrusters and the integrated propulsion system. Additional hot-fire tests were performed after the integration of the propulsion system with the vehicle. Finite

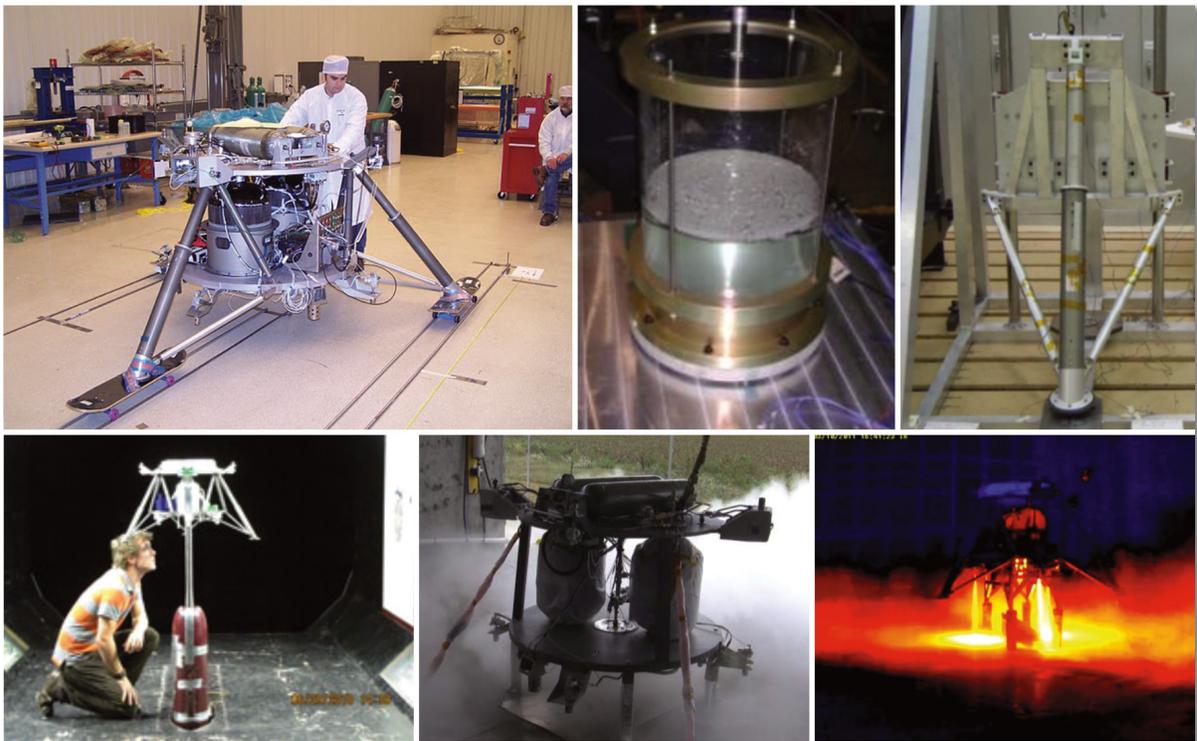


Figure 8. Testing of Mighty Eagle subsystem components. Clockwise from upper left: GNC polarity test, propellant slosh testing, leg drop test, infrared of integrated hot-fire test, propulsion acceptance test, and wind tunnel test. (NASA)

element models of the different vehicle structure configurations were developed to analyze vehicle stiffness and strength. The performance of the lander legs was demonstrated through drop tests to demonstrate the legs' strength and shock absorption capability. A wind tunnel test was performed on a scale model of the lander to characterize the disturbance torques and forces on the vehicle from relative wind on the vehicle. Propellant slosh tests were performed to characterize the motion of the liquid propellant during flight. Several vibration table tests were also done to characterize the performance of the IMU vibration isolation mounts. Once the vehicle was assembled, a polarity test of the navigation system was also performed to ensure proper mounting and communications between the various system components.

The GNC engineers developed a high-fidelity simulation of the lander system that includes estimated sensor and actuator performance, including misalignments and known disturbances such as wind and propellant motion. By using the system knowledge and uncertainty estimates from the various component tests and system requirements, the team performed a wide variety of Monte Carlo simulations to demonstrate the robustness of the system to the full range of expected variability in flight conditions and lander configuration. Additional hardware-in-the-loop simulations were also used by the flight software, ground system, and mission operations teams to aid in the development and testing of the ground and flight software interfaces, the integration and testing of the ground and flight software, and development of the flight test operations procedures. During software integration and testing, the team embraced the concept of "test like you fly, fly like you test." The entire life cycle for ground system and mission operations was exercised, including software installation and configuration, command and telemetry database builds, command and telemetry display development, command script development, operations product validation, real-time command and telemetry processing, telemetry playback, telemetry data archiving, and postflight data retrievals, including telemetry plots and history reports. Before the first actual Mighty Eagle test flight, the flight operations team was able to practice and refine the flight procedures used for the actual flight tests.

FLIGHT TESTING

After the completion of the final preflight tests in March

Table 2. Summary of completed indoor flight tests

Flight no.	Flight time (s)	Target altitude (m)	Purpose
1	9	0.55	Tethered low hover, demonstrate stability
2	18	0.5	Tethered low hover, demonstrate "land now" abort
3	26	1.0	Low hover, demonstrate untethered stability
4	33	2.0	Longer hover flight over range of fill levels
5	19	0.9	4 m lateral translation at 0.5 m/s
6	14	5.0	High ascent with varying descent rates

and April of 2011, the Mighty Eagle team transitioned from the development stage to the flight testing stage. To date, there have been three successful flight test campaigns, with a fourth campaign currently under way in 2013.

Indoor Flight Testing

The first set of six successful flight tests was performed during June and July of 2011 at an indoor test facility at the Redstone Test Center in Huntsville, Alabama. These initial tests, summarized in Table 2, were designed to demonstrate stable flight of the vehicle and the ability to perform a variety of vertical and lat-



Figure 9. Tethered indoor flight test. (NASA)

eral maneuvers. The first two flights were performed with the vehicle tethered to the ground, allowing low altitude flight but limiting vehicle travel. These tethered test flights were adopted as part of the nominal checkout procedure to demonstrate stable flight after any software or hardware configuration changes to the vehicle. In the first flight, illustrated in Fig. 9, the primary GNC system, GNCA, autonomously controlled the vehicle through the entire flight. In the second test, a manual “Land Now” soft-abort command was issued to test the functionality that commanded the backup GNCA system to land the vehicle.



Figure 10. Infrared view of flight test 4. (NASA)

The third flight was the first untethered flight of the vehicle. During all of these flights, the vehicle demonstrated good attitude control, with the angle from vertical around each axis staying below 1.2° while the primary GNC was controlling the vehicle. During the first two flights, it was observed that within approximately 0.1 m (0.33 ft) of the ground, the effectiveness of the vertical thrusters was reduced from ground effects. To mitigate against this effect, the lander was placed on small aluminium riser blocks starting with the third flight to increase thrust during takeoff. During the first three flights, the vehicle also experienced a vertical drift, indicating a bias in the navigation system. This drift was traced to several contributing factors, including a small, uncompensated bias in the accelerometer resulting from IMU vibrations and a large number of erroneous altimeter measurements caused by the altimeter sensing debris disrupted by the thrust. Additional tuning to GNC parameters on subsequent flights mitigated these effects.

During the second set of three indoor flight tests, shown in Figs. 10–12, the flight envelope of the vehicle was expanded to demonstrate the flight capabilities of the vehicle within the constraints of the indoor facility. The fourth flight extended the flight time to 33 s to demonstrate stable control over a larger range of propellant fill levels. In the fifth flight, shown in Fig. 11, the vehicle was commanded to perform its first lateral translation maneuver of 4 m (13 ft) at a rate of 0.5 m/s (1.6 fps). In the sixth and final indoor test flight, shown in Fig. 12, the vehicle was commanded to ascend to 5 m (16 ft) and then descend with an initial rate of -1.7 m/s (-5.6 fps) and then slow to -1.0 m/s (-3.3 fps) before touching down.

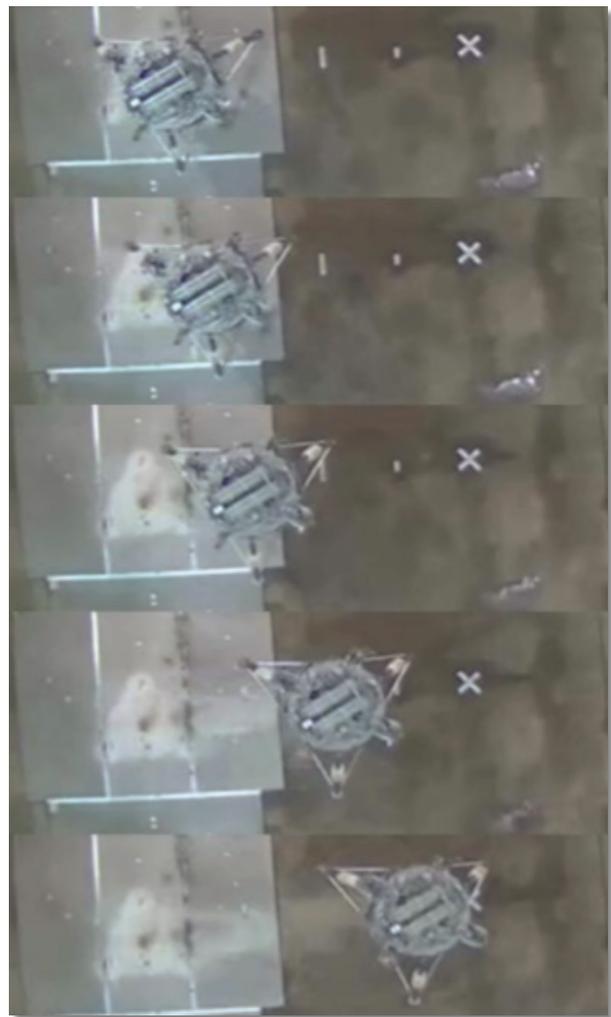


Figure 11. Overhead view of first lateral maneuver flight 5. (NASA)

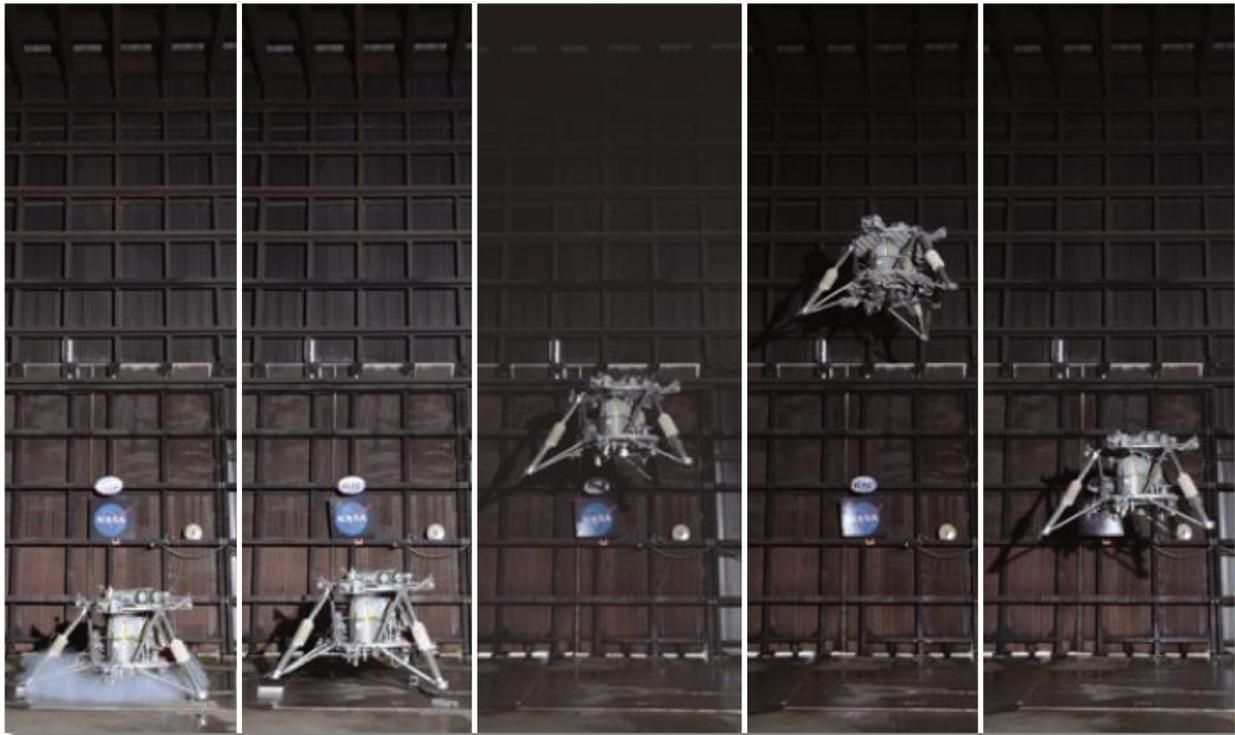


Figure 12. Sixth and final indoor test flight to 5 m altitude. (NASA)

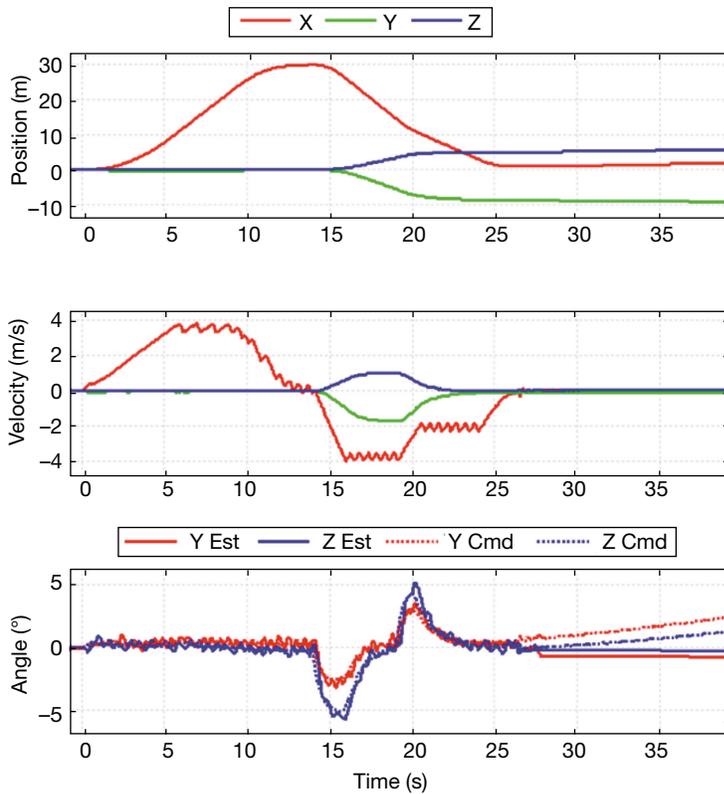


Figure 13. Flight profile of outdoor flight to 30 m with lateral translation.

Outdoor Flight Test Campaign 2011

After the completion of indoor testing, the Mighty Eagle team transitioned operations to an outdoor test range also at the Redstone Test Center. The technical objective of the outdoor tests, flown during September through November of 2011, was to increase the flight envelope of the vehicle to include higher-rate translations at up to 2 m/s (6.6 fps), descents from 30 m (98 ft), and a 90° slew of the vehicle around the vertical axis before touchdown. A secondary objective of the outdoor test flights was to demonstrate optical velocity estimation on the vehicle. Similar to the indoor test sequence, initial outdoor tests were performed using tethers to verify vehicle performance and operation of the flight termination sequence. After these initial checkout tests, the flight envelope was gradually increased, leading to a lateral 10 m (33 ft) translation while descending from 30 m (98 ft) to approximate a terminal lunar descent, as shown by the flight profile in Fig. 13.

The largest obstacle during the outdoor test was a gradual decrease in thruster performance during later flights, resulting from gradual catalyst degradation that is



Figure 14. Outdoor flight tests in 2011. (NASA)

not uncommon with hydrogen peroxide systems. This degradation was visible during multiple flights when the exhaust plumes of the thrusters became visible from incomplete decomposition of the propellant. Figure 14 illustrates two outdoor flight tests, one before visible exhaust plumes and one with visible plumes. Although the reduced thrust did not compromise the safety of the vehicle, it did prevent several flights from fully achieving their objectives. During one flight with a larger propellant load, the vehicle had insufficient thrust to lift off, and during two flights, manual aborts were declared as a precaution when large amounts of visible exhaust limited visibility of the vehicle. The demonstration of the optical velocity algorithms was also limited by degraded propellant decomposition. The image processing code successfully ran on the flight processor and communicated with the navigation filter, although the velocity measurements were autonomously rejected because of poor image quality resulting from visible thruster exhaust in the field of view. In mid-November, a checkout and refurbishment of the catalyst beds was performed, and the final two test flights demonstrated the restoration of the system to its nominal performance. Overall, the outdoor test sequence proved to be very successful in demonstrating the flight capabilities of the vehicle. The sequence of performed outdoor tests during 2011 is summarized in Table 3.

Outdoor Flight Test Campaign 2012

With completion of GNC system validation from the 2011 flight test series, internal NASA independent research and development funding was obtained

to demonstrate additional optical guidance in 2012. This work built on a long history of using AR&C for docking, capture, and berthing operations at MSFC. Applications for AR&C on a free-flying lander include satellite servicing, debris mitigation, sample return, and near-Earth object proximity operations. By using the nadir camera on the lander, an AR&C algorithm was added to the flight software. The AR&C target consists of four circles, as shown in Fig. 15. With knowledge of the diameters and the relative positions of the target circles, the image processing algorithm solves for the target in camera coordinates. The AR&C algorithm then uses the position estimate from GNCA and transforms the target position from the camera to topocentric coordinates. This target position is then passed to GNCA as its commanded position. This architecture allows full reuse of the validated GNCA flight code, conserving limited project resources, while demonstrating the new capability of in-flight adjustments to vehicle trajectory. A summary of the AR&C flight testing is shown in Table 4. During initial flights, the AR&C algorithm was run in an open-loop mode where the AR&C optical target detection algorithm ran onboard the vehicle but did not feed any commands to the GNCA module that followed a predefined trajectory. During later flights, the algorithm was run closed loop, and the target position was sent to the GNCA module as its guidance command.

In addition to demonstrating the AR&C capability on the lander, during the 2012 flight test campaign, the Mighty Eagle lander flight operations began to transi-

Table 3. Mighty Eagle flight profiles performed during outdoor testing in 2011

Objectives	Maximum altitude (m)	Flight time (s)	Status
Tethered checkout of infrastructure	0.8	12.0	Successful
Tethered checkout of flight termination system	0.8	12.0	Successful
1 m height, 10 m translate at 1 m/s	1.0	30.0	Unsuccessful (auto-aborted take-off after missed avionics packet)
0.6 m height, 10 m translate at 1 m/s	0.6	30.0	Successful
1 m height, 10 m translate at 2 m/s, slew 90°	1.0	42.0	Unsuccessful (insufficient thrust)
1 m height, 10 m translate at 2 m/s, slew 90°	1.0	30.0	Successful
Ascend to 10 m, hover, translate 10 m at 2 m/s, descend at (2, 1) m/s	10.0	30.0	Successful
Ascend to 10 m, hover, translate 10 m at 2 m/s, descend at (2, 1) m/s	10.0	17.0	Partially successful (manual abort commanded at 10 m)
Ascend at 0.5 m/s, hover 6 s, descend at 0.5 m/s	1.0	10.0	Successful
Ascend to 10 m, hover, translate 10 m at 2 m/s, descend at (2, 1) m/s	10.0	30.0	Successful
Ascend to 30 m at 3.7 m/s, translate 10 m while descending at (3.7, 2, 1) m/s, brief hover at 1 m	30.0	27.0	Successful
Ascend to 10 m, translate 10 m while descending to 2 m, ascend to 10 m, descend back to starting point with brief hover before touchdown	10.0	17.0	Partially successful (soft touchdown and manual abort after half maneuver)
Tethered checkout of catalyst bed refurbishment	0.8	13.0	Successful
Ascend to 10 m, hover, translate 10 m at 2 m/s, descend at (2, 1) m/s	10.0	30.0	Successful

tion to a new team of engineers. Several team members with existing experience took leadership roles, and several new, young engineers were brought onto the team. This transition demonstrated the value of using the lander both to train and engage young engineers with hands-on experience and also to demonstrate sensors and capabilities that could extend beyond the primary mission of lunar landing.

Ongoing Flight Test Campaign

In 2013, the Mighty Eagle team was awarded additional NASA independent research and development funding at the MSFC to extend the AR&C effort to demonstrate hazard avoidance. The plan is to use a commercial stereo camera to identify hazards, including boulders, slopes, and craters, in a simulated lunar terrain field. Hazard avoidance flight testing is under way in 2013.

CONCLUSIONS

APL and the MSFC collaborated to develop and flight test the Mighty Eagle robotic lander test bed. The team has flown multiple flight test campaigns starting



Figure 15. Mighty Eagle descending to the AR&C target. (NASA)

Table 4. Summary of 2012 AR&C flight series

Flight	Summary
1	Tethered 14 s flight time, 0.8 m max altitude, functional checkout, EGC throttle motor errors
2	Retest of flight, 1. 14 s flight time, 0.8 m max altitude, expand understanding of temperature-dependent EGC throttle motor errors
3	AR&C open loop, 32 s flight time, 10 m max altitude, 10-in.-diameter circular targets, no AR&C solutions were generated, noted dirty lens
4	AR&C closed loop, 32 s flight, 10 m max altitude, nine AR&C solutions generated, image processing parameters updated before this flight
5	AR&C open loop, 36 s flight, 30 m max altitude, new target (24-in.-diameter circles), 10 AR&C solutions generated
6	AR&C closed loop, 36 s flight, 30 m max altitude, 11 AR&C solutions generated, culmination of AR&C testing
7	Tethered, 10 s flight, 0.8 m altitude, test of “land now” in GNCSB with descent rate a function of altitude
8	Envelope expansion flight (higher, faster, longer), 43 s flight, 51 m max altitude, ascend at 6 m/s
9	Tethered, 14 s flight, 0.8 m altitude, change to lightweight legs
10	Student Launch Initiative demonstration, 34 s flight, 30 m max altitude, first ascent with downrange translation component

in the summer of 2011 that have gradually demonstrated increased capabilities of the lander and descents from as high as 50 m (164 ft) to simulate the terminal descent to the lunar surface. In addition to serving as a system demonstration for technologies needed for an actual lunar mission, the Mighty Eagle development effort has also served as a catalyst for developing an integrated team of APL and MSFC engineers, providing young engineers with valuable hands-on experience and also providing an extensible platform that has been used to demonstrate additional mission concepts and sensor systems.

ACKNOWLEDGMENTS: A large number of talented team members across multiple disciplines and organizations contributed to the prototype lander and larger RLLDP efforts during the several years of development and flight testing. Although not all of the team members can be listed here, the authors would like to acknowledge some of the team members with whom they regularly interacted during the development and testing of the Mighty Eagle. During the initial development and flight tests, Julie Bassler was the MSFC program manager, and Larry Hill was the MSFC deputy program manager. Brian Mulac was the lead engineer of the prototype development. Brian Morse and Dewey Adams were the APL program leads through various stages of the program. At APL, several GNC engineers, including David Carrelli, Adam Fosbury, Jim Kaidy, Murty Challa, Stella Shapiro, and Wayne Dellinger, supported GNC algorithm development, verification, and testing. Tom Criss developed the initial prototype version of the LSOF algorithm, and

Wen-Jong Shyong kept our development and autocode generation environment running smoothly. Gail Oxtan, Scott Martucci, and Bob Davis supported development of the GNC wrapper code and integration and testing of the autogenerated GNC code on embedded processors. Doug Reid implemented the LSOF algorithm in C code, and Justin Thomas developed the camera control and emulation software. Paul Lafferty managed installation of the MIGS ground software and provided support. Sanae Kubota and Ben Ballard provided systems integration support. The mechanical team included Terry Betenbaugh (lead analyst), Deva Ponnusamy (leg design and testing), Gordon Maahs (leg analysis and testing), Marc Briere (nonlinear analysis), and a design team of Bill French (ATK) and Darrius Pergosky. A Huntsville-based team of MSFC and VCSI engineers, including Patrick O’Leary, Mike Turbe, John Wetherbee, Josh Eliser, and Dan Gunter, developed the lander operations procedures and screen displays, developed the backup GNC system, and integrated and tested the full GNC software build in the vehicle flight software. Scott Gilley was the avionics lead during the lander development, Kyle Daniel was the safety lead, Todd Freestone was the communications lead, and Charlie Adams led the alignment and weight and balance for the lander. Robert Harris led the original red crew that was responsible for vehicle flight preparations. Mark Wells, of Dynetics, was the lead propulsion designer. Other red team members included T. J. Talty, Michael Thomas, and Wayne Neumaier, who also provided propulsion engineering support. Jason Adams is the current lead engineer for ongoing Mighty Eagle testing.



Figure 16. The Mighty Eagle team.

REFERENCES

- ¹Morse, B. J., Reed, C. L. B., Kirby, K. W., Cohen, B. A., and Bassler, J. A., "NASA's International Lunar Network (ILN) Anchor Nodes Mission," in *Proc. 59th International Astronautical Congress 2008*, Glasgow, Scotland, paper IAC-08-A.3.2.B1 (2008).
- ²Morse, B. J., Reed, C. L. B., Eng, D. A., Kirby, K. W., Cohen, B. A., et al., "NASA's International Lunar Network (ILN) Anchor Nodes Mission Update," in *Proc. 60th International Astronautical Congress 2009*, Daejeon, Republic of Korea, paper IAC-09.A.3.2.B6 (2009).
- ³Morse, B. J., Reed, C. L. B., Kirby, K. W., Cohen, B. A., Harris, D. W., et al., "NASA's International Lunar Network (ILN) Anchor Nodes and Robotic Lunar Lander Project Update," in *Proc. 2010 Global Lunar Conf.*, Beijing, Peoples Republic of China, paper GLUC-2010.1.5.B3 (2010).
- ⁴Morse, B. J., Reed, C.L., Ballard, B.W., Cohen, B.A., Bassler, J.A., et al., "NASA's Robotic Lunar Lander Development Project Update," in *Proc. 61st International Astronautical Congress 2010*, Prague, Czech Republic, paper IAC-10.A.3.2.B3 (2010).
- ⁵McGee, T. G., Cole, T., Artis, D., Morse, B. J., Reed, C., et al., "NASA's Robotic Lunar Lander Development Project: Initial Flight Testing Results of a Robotic Lunar Lander Test-Bed," in *Proc. 62nd International Astronautical Congress 2011*, Cape Town, South Africa, paper IAC-11.A.3.2.B.5 (2011).
- ⁶Adams, D., Ballard, B., Cole, T. J., McGee, T. G., Reed, C., et al., "NASA's Robotic Lunar Lander Development Project," in *Proc. 63rd International Astronautical Congress 2012*, Naples, Italy, paper IAC-12.A.3.2.C.1 (2012).
- ⁷Pollard, A., and McGee, T., "Design Considerations for a Guidance, Navigation, and Control Sensor System for a Robotic Lunar Lander," in *Proc. 2011 IEEE Aerospace Conf.*, Big Sky, MT, doi: 10.1109/AERO.2011.5747339 (2011).
- ⁸Chavers, G., Ballard, B., Cohen, B. A., McGee, T., Moore, J., et al., "Robotic Lunar Lander Development Status," in *Proc. 2012 Global Space Exploration Conf.*, Washington, DC, paper GLEX-2012.03.1.8x12731 (2012).
- ⁹Cohen, B. A., Chavers, D. G., and Ballard, B. W., "NASA's Robotic Lunar Lander Development Project," *Acta Astronautica* **79** (2012), 221–440.
- ¹⁰Northrop Grumman, Capabilities, "LN-200 FOG Family Advance Airborne IMU/AHRS," <http://www.northropgrumman.com/Capabilities/LN200FOG/Pages/default.aspx> (accessed 4 Dec 2010).
- ¹¹Roke Manor Research Ltd., "Miniature Radar Altimeter: MRA Type 2," Issue 05-000, <http://www.roke.co.uk/resources/datasheets/MRA-Type-2.pdf> (February 2013).
- ¹²Honeywell, "Honeywell HG8500 Series Radar Altimeter," DFOISR 03-S-1426 http://www51.honeywell.com/aero/common/documents/myaerospacatalog-documents/MilitaryAC/HG8500_Radar_Altimeter.pdf (accessed 13 Jun 2003).
- ¹³Cole, T. J., Bassler, J., Cooper, S., Stephens, V., Ponnusamy, D., et al., "The Challenges of Designing a Lightweight Spacecraft Structure for Landing on the Lunar Surface," in *Proc. 61st International Astronautical Congress 2010*, Prague, Czech Republic, paper IAC-10.A.3.2.C.7 (2010).
- ¹⁴Cole, T., Chavers, D. G., Ponnusamy, D., and Betenbaugh, T., "The Mechanical Design of an Earth-based Demonstrator for the Robotic Lunar Lander Development Project," in *Proc. 2012 Global Space Exploration Conf.*, Washington, DC, paper GLEX-2012.03.P.4x12733 (2012).
- ¹⁵Wilmot, J., "Implications of Responsive Space on the Flight Software Architecture," in *Proc. 4th Responsive Space Conf.*, Los Angeles, CA, paper RS4-2006-6003 (2006).
- ¹⁶McGee, T., Kaidy, J., Criss, T., Reid, D. Oxtton, G., and Hannan, M., "Guidance, Navigation, and Control Development for a Robotic Lunar Lander Testbed," in *Proc. 2011 34th Annual American Astronautical Society Rocky Mountain Section Guidance and Control Conf.*, Breckenridge, CO, paper 11-032 (2011).
- ¹⁷Dellinger, W. F., Salada, M. A., and Shapiro, H. S., "Application of MATLAB®/Simulink® to Guidance and Control Flight-Code Design," *Proc. 22nd Annual American Astronautical Society Rocky Mountain Section Guidance and Control Conf.*, Breckenridge, CO, Vol. 101, pp. 301–316 (1999).

- ¹⁸Adams, D., Criss, T. B., and Shankar, U. J., "Passive Optical Terrain Relative Navigation Using APLNav," in *Proc. IEEE Aerospace Conf.*, Big Sky, MT, doi: 10.1109/AERO.2008.4526303 (2008).
- ¹⁹McGee, T. G., Shankar, U., Shapiro, S., Shyong, W. J., Krupiarz, C. J., et al., "Performance Analysis of the APLNav System for Passive Optical Lunar Navigation," in *Proc. AIAA Guidance, Navigation, and Control Conf.*, Chicago, IL, paper AIAA 2009-5808 (2009).
- ²⁰White, M., Criss, T., and Adams, D., "APLNav Terrain Relative Navigation Helicopter Field Testing," in *Proc. AIAA Guidance, Navigation, and Control Conf.*, Chicago, IL, paper AIAA 2009-5807 (2009).
- ²¹Shankar, U. J., Shyong, W. J., Criss, T. B., and Adams, D., "Lunar Terrain Surface Modeling for the ALHAT Program," in *Proc. 2008 IEEE Aerospace Conf.*, Big Sky, MT, doi: 10.1109/AERO.2008.4526300 (2008).
- ²²Lopez, N., and McGovern, A., "Increasing Confidence in Exploration Technologies Utilizing Synthetic Data Products," in *Proc. 2009 AIAA SPACE Conf. and Exposition*, Pasadena, CA, paper AIAA 2009-6704 (2009).
- ²³Kaidy, J., McGee, T., Criss, T., Heyler, G., Shyong, W. J., and Guzman, J., "Spacecraft Precision Landing System for Low Gravity Bodies," in *Proc. Annual American Astronautical Society Rocky Mountain Section Guidance and Control Conf.*, Breckenridge, CO, Vol. 137, pp. 221–238 (2010).

The Authors

Timothy G. McGee led the GNC development for the Mighty Eagle lander. He is a member of the Senior Professional Staff at APL. Dr. McGee's technical areas of interest include guidance and control of autonomous systems, image processing, optical navigation, and robotic systems. **David A. Artis** and **Timothy J. Cole** were the software and mechanical leads, respectively, for the Mighty Eagle. **Douglas A. Eng** is currently the Group Supervisor of the Space Systems Applications Group within the Space Department at APL and provided systems engineering support for the Mighty Eagle. **Cheryl L. B. Reed** is the current APL Program Manager for the RLLDP. **Michael R. Hannan** provided additional GNC support for the project at MSFC. **D. Greg Chavers** and **Cynthia D. Stemple** currently lead the RLLDP efforts at MSFC. **Logan D. Kennedy** and **Joshua M. Moore** are engineers at MSFC who have provided systems engineering support throughout RLLDP. For further information on the work reported here, contact Timothy McGee. His e-mail address is timothy.mcgee@jhuapl.edu.

The Johns Hopkins APL Technical Digest can be accessed electronically at www.jhuapl.edu/techdigest.