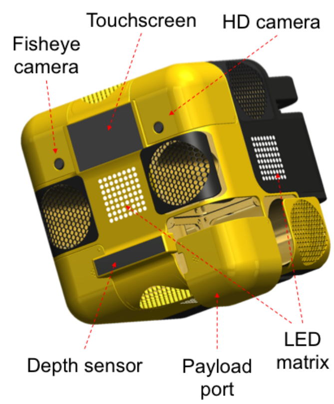
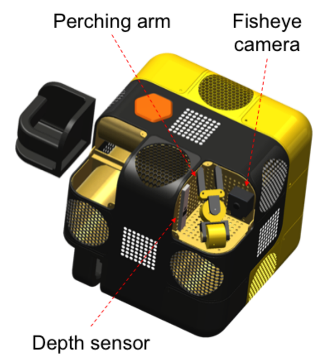
Avionics and Perching Systems of Free-Flying Robots for the International Space Station

Jongwoon Yoo, In-Won Park, Vinh To, Jason Q. H. Lum, and Trey Smith
Intelligent Robotics Group
NASA Ames Research Center
Mountain View, CA 94035
Email: {jongwoon.yoo | in.w.park | vinh.to | jason.q.lum | trey.smith}@nasa.gov

***Abstract*—This paper introduces NASA's new free-flying robot, Astrobee, especially focusing on its avionics and perching subsystems. Astrobee is a cube-shaped autonomous robot designed for various missions on the International Space Station (ISS). Its major goal is to offload routine and repetitive work from the ISS crews and assist their science activities. Astrobee is also designed for scientists to use it as a micro-gravity robotics research platform. It can host various science equipment and software, allowing scientists to conduct their experiments using Astrobees on the ISS. The robot has a small compliant, detachable arm with a griper so that it can perch on the ISS wall to support long duration tasks. This arm will grasp ISS handrails to hold its position without using propulsion or navigation subsystems to minimize power consumption. Due to its special missions and operating environment, Astrobee has a set of unique design requirements. This paper gives an overview of Astrobee and the details of its avionics and perching subsystems with distinctive design challenges. We also present the trade studies that we have conducted to decide the critical hardware and software components for the avionics.**



(a) Front



(b) Back

Fig. 1. Astrobee free-flying robot

I. Introduction

Since 2014, the National Aeronautics and Space Administration (NASA) has been developing a new free-flying robot called *Astrobee*. Astrobee is a cube-shaped autonomous robot (Fig. 1), which will serve as a robotic assistant on the International Space Station (ISS) starting from 2017. Its major goal is to offload routine, repetitive, or simple but long-duration tasks, such as conducting environment surveys, taking sensor readings, or performing routine maintenance, from the ISS crews and assist their science activities.

Astrobee is designed on top of technologies and lessons learned from Smart Synchronized Position Hold, Engage, Reorient, Experimental Satellite (SPHERES) [1][2] free-flying robot, which has operated on the ISS since 2003. Like SPHERES, Astrobee can host science hardware and software, allowing scientists to conduct their experiments in a micro-gravity environment. It has multiple expansion ports where research payloads can be attached for demonstrating, testing or use on the ISS. Unlike SPHERES, of which operating space is limited to $2 \times 2 \times 2\mathrm{m}^3$ due to its beacon-based localization, Astrobee extends the operating space to the entire US Orbital Segment (USOS) of the ISS by using vision-based localization. It also enhances computing environment for guest scientists by having a dedicated quad-core general purpose ARM processor for micro-gravity robotic research, while SPHERES provides limited computing capacity based on a digital signal processor.

The current concept design of Astrobee is depicted in Fig. 1. It can autonomously navigate and avoid obstacles in the ISS USOS based on a fisheye camera and depth sensor on its front side (Fig. 1(a)). It also has an HD camera on board, which allows it to serve as a remotely controllable mobile camera platform. A touchscreen, multiple LED matrices, microphones, and speakers are equipped for Human-Robot Interaction (HRI) research.

On the back side, Astrobee has a small, compliant, and detachable arm with a griper as shown in Fig. 1(b). This arm will grasp the handrails in the ISS to hold its position without using propulsion or navigation to minimize power consumption. Detection of handrails is done by using another fisheye camera and depth sensor on the back side. The arm will also support Astrobee robots grasping each other to enable future research related to satellite servicing.

Due to its special missions and operating environment, Astrobee has a set of unique design requirements. This paper gives an overview of Astrobee project and the details of its avionics and perching subsystems with distinctive design challenges. We also present the trade studies that we have conducted to decide the critical hardware and software components for the avionics. A prototype implementation with a micro-gravity test environment is presented, demonstrating the performance and capabilities of Astrobee.

978-1-4799-1920-8/15/$31.00 ©2015 IEEE

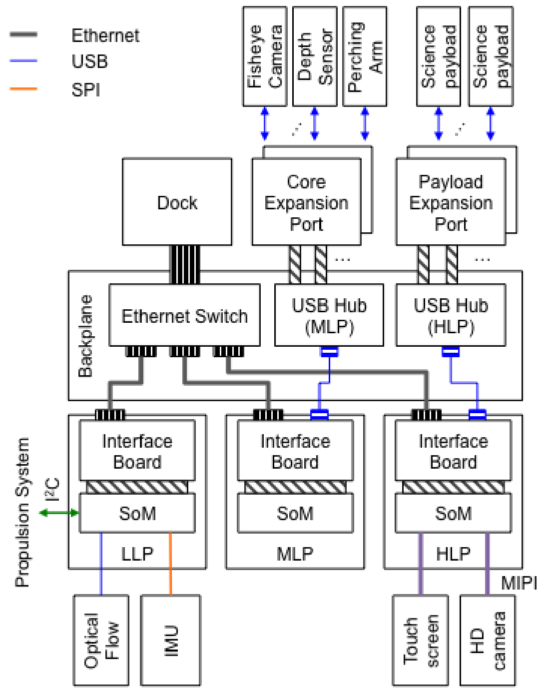


Fig. 2. The block diagram of the Astrobee avionics.

II. AVIONICS

The avionics provides computation and communication resources to Astrobee. This section give an overview of the avionics and then describes the trade study on processor selection.

A. *Overview*

The block diagram of avionics is shown in Fig. 2. Astrobee has three different processor boards for computing: Low-Level Processor (LLP), Middle-Level Processor (MLP), and High-Level Processor (HLP). Each processor board has different hardware and software features depending on its role as explained later in this section. These processor boards are connected to the backplane, which has an Ethernet switch and two multi-port USB hubs. Ethernet is chosen as the main communication technologies for the processors to communicate with each other. The dock interface is also connected to the Ethernet switch for data transfer between Astrobee and the ISS infrastructure while the robot is docked. External devices, such as cameras, can be connected to either MLP or HLP through the expansion ports. Each port has a standardized pinout compatible with USB 2.0 and mating structure, allowing any USB 2.0 device to be connected to the robot. For example, two fisheye cameras, two depth sensors, and the perch arm are connected to the MLP through the expansion port. Guest scientists can attach their experiment devices to the HLP for testing.

The LLP runs the most timing- and safety-critical tasks. An example of this type of tasks is the precise control of propulsion system. In our prototype, for example, four variable pitch propellers (VPPs) used for motion generation are controlled by software running on the LLP, which has four closed-loop threads that control the speed and pitch of each

	Overall	Power Cons	Computing Power	Comms	SW Dev Cost	HW Dev Cost	Modularity
Priority		2	1	1	2	3	1
Odroid-XU3	2.74	2	5	4	5	4	1
CM-FX6	2.90	3	2	3	4	2	4
phyCORE-OMAP5430	3.12	3	3	5	1	3	5
OMAP5430 Pico ITX	2.61	3	3	5	3	4	1
DragonBoard 8084	3.04	1	5	5	1	4	5
IFC6410	2.31	2	4	2	5	4	1
IFC6540	2.59	1	5	5	4	4	1
IFC6400	3.03	2	4	2	5	3	4
IFC6501	3.14	1	5	3	4	2	5

Fig. 3. Trade study result for the middle level processor candidates.

VPP at 100 Hz. The LLP's another important task is braking the robot in case of any unexpected failure in navigation system running on the MLP. To avoid hitting the ISS wall or equipment there while navigation is not supported, the LLP controls the propulsion system to stop the robot and hold its position, waiting for the crews to come and help it. To this end, the LLP has an optical flow sensor and inertial measurement unit (IMU) to estimate the velocity of the robot on board [8][9]. The LLP publishes the optical flow and IMU measurements so that software running on the MLP or HLP can access the data. The main role of MLP is to combine all sensor data and run vision-based mapping and navigation algorithms. It can also localizes the robot uing WiFi signals [10]. The HLP is a dedicated processor module for guest scientists.

B. *Trade Study*

We decided to use commercial System-on-Module (SoM) products to reduce the time and cost of processor module development. Choosing the appropriate processor modules that meet the given design requirements is one of the most important issues in designing the avionics. We designed a trade study which scores processor modules from one to five points in six different attributes and gives overall ratings. The attributes should be chosen carefully so that given design requirements are well reflected. The importance of each attribute could vary depending on the processor module types. For example, the computing power is the most critical attribute for the MLP, which runs the core flight software. The importance is then used as a weight when calculating the overall score. The attributes for the MLP are explained below:

- *Computing power*: The MLP runs vision-based mapping and navigation techniques which demand significant computing power. Our flight software algorithms are expected to require computing power similar to Project Tango [4]. Higher scores are given to the processor modules that have similar specifications to the Tango device.
- *Software development cost*: The software running on the MLP is based on Linux and the Robot Operating System (ROS). Therefore, the processor modules which officially support ROS-friendly Linux distributions, device drivers, and active open source communities get higher scores.
- *Hardware development cost*: Processor modules which can be easily integrated to the backplane get higher scores. For example, tiny Single-Board-Computers (SBCs) can be easily connected to the backplane with some standard

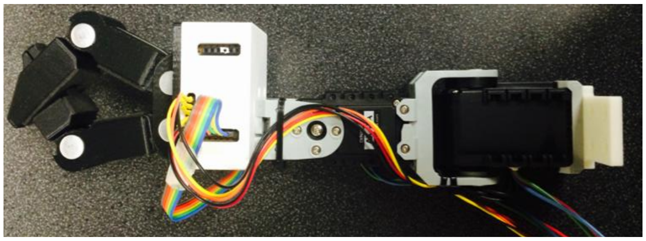


Fig. 4. Snapshot of the first prototype of the Astrobee perching arm.

cables, while SoMs require custom interface boards that route high speed signals.

- *Modularity*: The MLP should be able to be replaced by the crews on the ISS in case of failure or upgrade. The more bolting, nutting, or cabling is required, the more time and effort are needed for the crews to replace the processor module. Therefore, SoMs with edge connectors get higher scores as they can be swapped like memory cards.
- *Communications*: To communicate with the LLP, HLP, peripherals, and the payload on the expansion ports, the MLP should have Ethernet, I^2C, USB 2.0, and USB 3.0 interfaces. WiFi is also required to communicate with the outside of the robot.
- *Power consumption*: Astrobee should run off batteries for about 3 to 9.5 hours depending on the use cases. Of course, the lower power consumption the higher scores.

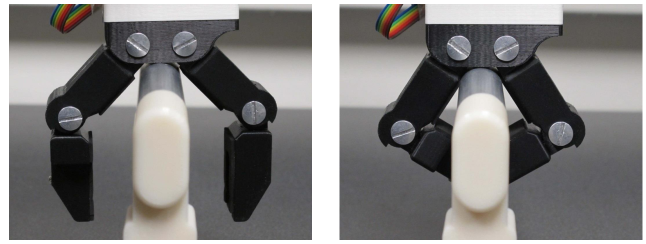
Based on the trade study result shown in Fig. 3, Infoce Computing's IFC6501 SoM [6] is chosen as the MLP of Astrobee. We conducted the similar trade studies for the LLP and HLP. As results, the Wandboard Dual [7] and IFC6501 are selected as the LLP and HLP, respectively.

III. Perching Arm

As a part of the Astrobee robotic system, a compliant, detachable perching arm is being developed to support long duration tasks. This arm will grasp ISS handrails to hold its position without using propulsion or navigation to minimize power consumption.

The perching arm of Astrobee robot is required to be small, lightweight and compliant, where the expected total mass should be less than 200 g. In order to meet the allocated mass, power, and size requirements, a compliant claw gripper with a two degree-of-freedom (DOF) arm is being developed as the first prototype of an Astrobee perching arm as shown in Fig. 4. The 2-DOF arm consists of 2 Dynamixel AX-12A motors and the tendons in the gripper are connected to a Pololu metal gearmotor. The length and mass of the Astrobee perching arm are 24.0 cm and 315.0 g, respectively. The 2-DOF arm is used to stow the gripper inside of the outer structure during flight so that it is not exposed to collision hazard while stowed. When the arm is successfully perched, it can also operate as a pan-tilt module for a camera attached on the opposite side of the robot to support remote monitoring operations.

BeagleBone Black (BBB) board is the main controller, which communicates to middle level processor (MLP) using ROS. Dynamixel motors are directly controlled from the BBB via USB-serial converter and the Pololu motor is controlled using a Baby Orangutan (B-328) board, where the desired commands are sent from the BBB via serial. Level shifter is added to deliver 3.3V signals to the BBB and 5V signals to the Baby Orangutan board. Voltage converter is used to convert a battery voltage (7.2V) into 11.0V to Dynamixel motors and 5.0V to the BBB.



(a) Opened (b) Closed

Fig. 5. Snapshot of the Astrobee perching arm grasping an ISS handrail

Fig. 5 shows the open and closed configurations of the perching arm while grasping an ISS handrail. The gripper uses torsional springs for joint flexion and an actuated tendon for extension. This allows gripping force to be maintained even with the motor turned off. It also allows external forces to open the gripper by overcoming spring torques, rather than having to back-drive the motor. Furthermore, independent flexion torques at the proximal and distal joints provides passive compliance to the shape of the grasped object; the perching procedure is thus robust to positioning errors with respect to the handrail.

A total of 3 torsional springs (2 at the proximal joint and 1 at the distal joint) are used at each joint to produce a gripping force. When the gripper is fully open as shown in Fig. 5(a), the proximal joint makes a 45.0° wide opening with respect to the palm and the distal link makes a 45.0° wide opening with respect to the proximal link, which translates to a torque of 105.9 Nmm and 44.3 Nmm at the proximal joint and the distal joint, respectively. When the ISS handrail is grasped, as shown in Fig. 5(b), the gripping forces at the proximal joint and the distal joint are 3.47 N and 2.87 N, respectively.

IV. Conclusion

This paper introduced NASA's new free-flying robot, Astrobee, especially focusing on its avionics and perching subsystems. Astrobee will offload routine and repetitive work from the ISS crews and be used as a micro-gravity robotics research platform. The robot has a small compliant, detachable arm with a griper so that it can perch on the ISS wall to support long duration tasks. This arm will grasp ISS handrails to hold its position without using propulsion or navigation subsystems to minimize power consumption. This paper also presented a set of unique requirements of Astrobee and distinctive design challenges. We also presented the trade studies that we have conducted to decide the critical hardware and software components for the avionics.

Acknowledgment

This research was supported by an appointment to the NASA Postdoctoral Program at NASA Ames Research Center, administered by Oak Ridge Associated Universities through a contract with NASA.

References

[1] D. Miller, *et al*., "SPHERES: A Testbed for Long Duration Satellite Formation Flying in Micro-gravity Conditions," *AAS/AIAA Space Flight Mechanics Meeting*, Paper 00–110, 2002.

[2] J. Enright, *et al*., "The SPHERES Guest Scientist Program: Collaborative Science on the ISS," *IEEE Aerospace Conference*, vol. 1, pp. 35–46, Mar. 2004.

[3] G. A. Dorais and Y. Gawdiak, "The Personal Satellite Assistant: An Internal Spacecraft Mobile Monitor," *IEEE Aerospace Conference*, vol. 1, pp. 333–348, 2003.

[4] Google, "Project Tango," https://www.google.com/atap/project-tango/

[5] The Open Source Robotics Foundation, "The Robot Operating System (ROS)," http://www.ros.org/

[6] Inforce Computing, Inc., "IFC6501," https://www.inforcecomputing.com/.

[7] Wandboard, "Wandboard Dual", http://wandboard.org/

[8] D. Honegger, *et al*., "An Open Source and Open Hardware Embedded Metric Optical Flow CMOS Camera for Indoor and Outdoor Applications," *IEEE International Conference on Robotics and Automation*, (*ICRA'13*), pp. 1736–1741, 2013.

[9] V. Grabe, *et al*., "A Comparison of Scale Estimation Schemes for a Quadrotor UAV based on Optical Flow and IMU Measurements," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, (*IROS'13*), pp. 5193–5200, 2013.

[10] J. Yoo *et al*., "WiFi Localization on the International Space Station," *IEEE Symposium on Intelligent Embedded Systems*, (*IES'14*), pp. 21–26, 2014.