AIAA 2015-0801
AIAA SciTech
5-9 January 2015, Kissimmee, Florida
56th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference

Leveraging Open Standards and Credit-Card-Sized Linux Computers in Embedded Control & Robotics Education

Thomas Bewley, James Strawson
Coordinated Robotics Lab, UC San Diego, La Jolla, CA, 92093-0411

Clark Briggs
ATA Engineering, Inc., San Diego, CA, 92128

We present a new interdisciplinary educational program in multifunctional robotics built around three versatile vehicles – a mobile inverted pendulum (BeagleMiP), a four-wheel-steerable rover (BeagleRover), and a quadrotor (BeagleQuad) – all built around the TI BeagleBone Black (BBB), together with a new printed circuit connector board, or "cape", designed specifically for multifunctional robotics applications. The program is coupled with a C library of key robotics functions, together with working example codes and educational materials, making it exceptionally easy to get started, and to extend. Volume 1 of the program targets embedded control & robotics education at the "maker" level, and is easily accessible in high schools and beyond. Building directly on Volume 1, Volume 2 of the program targets the "professional" level, building on knowledge of ODEs, dynamics, control theory, and op amps, and is useful in senior-level college capstone courses & industry. Prototype versions of BeagleMiP were used by the authors during the last three offerings of UCSD's fall-quarter senior-level Embedded Control & Robotics course. The electronics used have evolved from an Arduino Nano and various small development boards, assembled on an 830-point solderless breadboard, to the powerful $50 BBB paired with our highly extensible robotics cape, which incorporates the commonly-needed additional electronics for many robotics applications, including an IMU, barometer, battery management, H-bridges, real-time clock with battery backup, and over a dozen convenient connectors, into a very small footprint. The best-practices programming techniques fostered by this program implement an advanced software architecture written in C, with different threads running at different loop rates and priorities. The resulting educational program will be commercially available soon, marketed in partnership with WowWee Robotics.

I. Introduction & Background

Modern disruptive technologies such as 3D CAD software, 3D printing & laser cutting, PCB design software, mail-order PCB fabrication & low-volume PCB assembly, smartphone and tablet SDKs, high-capacity lithium batteries, a huge variety of COTS sensors and motors, and sub-$50 credit-card-sized Linux computers like the Raspberry Pi and BeagleBone Black, together with the ongoing spin-off of cutting-edge low-power consumer cellphone technology into the robotics space, as well as the broadening availability of low-cost manufacturing capabilities, necessitate a major interdisciplinary revamping in the teaching of embedded control & robotics. The authors have developed a unique modular system of vehicles, electronics, software, and educational materials, built upon open standards and Linux, to support education in embedded control & robotics, both at the maker level, in high schools and beyond, and at the "professional" level, in college and industry. The result is an unprecedented, interdisciplinary, highly motivating learning experience based on agile high-function robotic vehicle prototypes that students continue to hack, extend, reference, and learn from long after the completion of formal classwork.

Universities typically disjoin the study of challenging technical fields into fragmented departments, like mechanical & aerospace engineering, electrical engineering, computer science, control & cybernetics, and human factors. Modern applications in robotics require the rejoining of these traditionally isolated disciplines of study. The effective design and control of robotic vehicles requires a fundamentally interdisciplinary perspective that is ill served by keeping the teaching of its constituent technical components disjoint. There is an emerging need for educational institutions to distill and relate these constituent disciplines, and the remarkable recent advancements therein, to a new generation of roboticists. Towards this end we have developed this program, which aims to provide a systematic, integrated introduction to the fundamental technologies and techniques available, focusing on control & coordination algorithms, open standards & tools, and software architectures that may be broadly used.

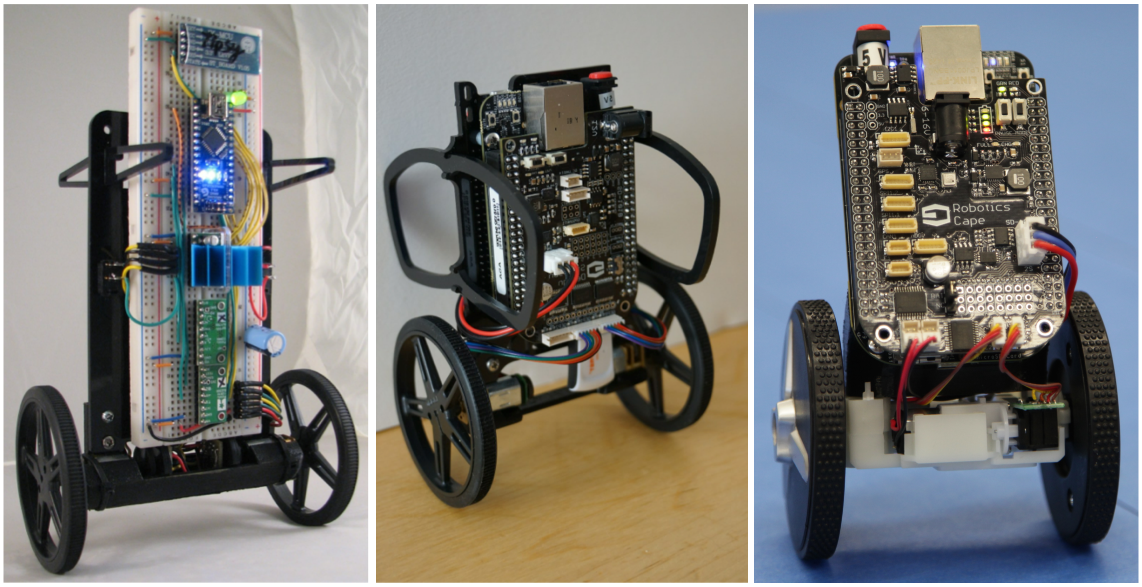

Figure 1. Mobile Inverted Pendulums individually built and programmed by students in the (left) 2012, (center) 2013, and (right) 2014 offerings of MAE143c at UCSD; the last two are prototype BeagleMiPs.

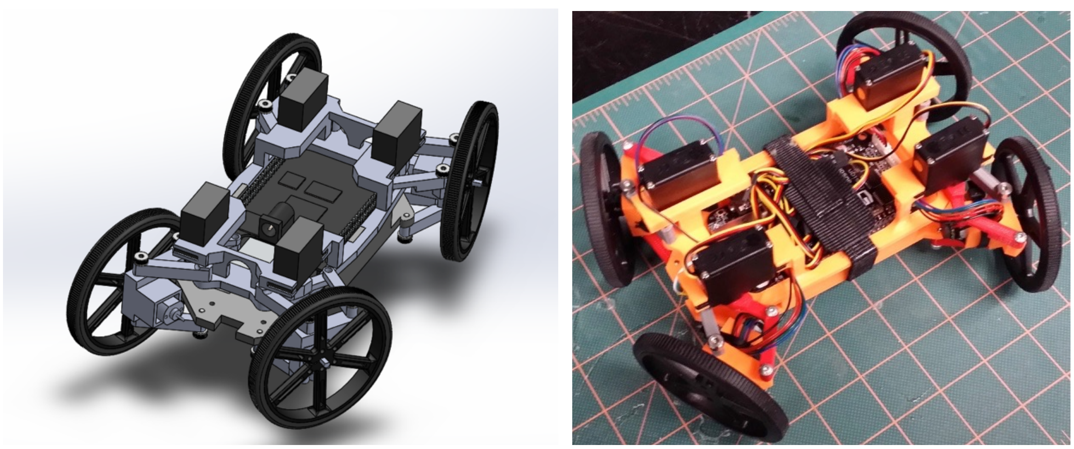

Figure 2. BeagleRover, a four-wheel rover characterized by four-wheel drive and “extreme” four-wheel steering, which facilitates unique maneuvers such as turning in place and sideways translation for perfect parallel parking.

Figure 3. BeagleQuad, a reference aerial platform for studying 3D dynamics, autopilots, attitude hold, and multivehicle coordination. BeagleMiP, BeagleRover, and BeagleQuad share the same electronics and codebase.

II. BeagleMiP, BeagleQuad, and BeagleRover

As outline above, three remarkable educational vehicle kits (Figures 1, 2, and 3) have been developed for this effort, all of which are built & controlled with the same Linux-based software library and electronics package, which includes the BBB paired with our robotics cape (Figure 4). Our cape, in turn, incorporates an advanced inertial measurement unit (IMU, an integrated circuit combining MEMS accelerometers, gyros, magnetometers, and signal processing), a barometer for sensitive measurement of altitude changes, H-bridges for direct connection to up to four small bi-directional DC motors, voltage regulation & battery management/charging circuitry, a real-time clock with battery backup, and several standard connectors for easy connection to COTS I2C, SPI, R/C, and GPS peripherals as well as up to three quadrature encoders and up to eight servomechanisms (servos) or electronic speed control (ESC) circuits. These kits are designed to reach retail SKUs of less than $200 each, readily facilitating *individual ownership* by students.

BeagleMiP (Figure 1) is the centerpiece of this effort. It is an unstable, nonminimum-phase “mobile inverted pendulum”, with dynamics similar to the standard test problem (a.k.a. “plant”) of a pendulum swinging freely from a cart as it moves along a track, but is much more compact, economical, and fun. We have found BeagleMiP to be quite useful and versatile for teaching feedback control theory at the professional level, and have implemented several different types of controllers to stabilize it, including classical (SISO) control strategies in the successive loop closure (SLC) framework, state-space control strategies, and adaptive control strategies. BeagleMiP is about 6” tall and 4” wide, with 3D printed and laser-cut structural members. Our latest prototype incorporates the robust wheel assembly used in the consumer “MiP” toy (see Figure 5), which was developed and manufactured in partnership between our lab and WowWee Robotics. BeagleMiP is rather simple to assemble and extend, with small wire bundles to connect the cape to the two motors, to the two encoders, and to the LiPo battery in its minimal configuration. At the maker level, BeagleMiP is useful to motivate more advanced (college-level) investigations in dynamic modeling and feedback control; a reference control solution, which makes the vehicle self-upright from horizontal and balance in the upright configuration, is provided for a rewarding out-of-the-box experience.

BeagleRover (Figure 2) is a four-wheel-drive rover characterized by “extreme four-wheel steering”, with each wheel capable of being turned independently (by servos) over 90 degrees. This facilitates a number of unique maneuvers, including as turning in place and sideways translation for parallel parking in unimaginably-small parking places. Four-wheel steering is a kinematically fascinating problem that is straightforward for students to visualize, but challenging to coordinate well in software.

BeagleQuad (Figure 2) is a quadrotor in a fairly standard configuration, which is a popular amongst hobbyists, as well as a versatile platform for airborne photography (e.g., disaster response, corporate espionage, …) and scientific measurement (e.g., historical sites, archaeological digs). The BeagleQuad exercises focus on stability augmentation systems, human factors like handling qualities when flying with remote control, and automatic pilots leveraging advanced optical flow algorithms.

A **warm-up kit** will also be made available, introducing how to use the BBB and cape to:
(a) drive individual LEDs as well as seven-segment displays,
(b) communicate over the I2C & SPI busses to offload simple repetitive tasks to auxiliary ICs,
(c) communicate with advanced ICs like the IMU, Orange & Spektrum DSM remote control units, and the EM-406 and EM-506 GPS units,
(d) communicate over standard (USB-based) bluetooth and WiFi modules to smartphones and tablets,
(e) output PWM signals to resistive or inductive loads (with flyback protection), DC motors, and servos/ESCs,
(f) read/count quadrature encoders to accurately monitor shaft rotations, and
(g) perform system identification to develop accurate dynamic models of the motors + H-bridges on the vehicle.

Since BeagleRover is dynamically stable, it provides the logical starting point for Volume 1 of this program. Well-documented educational “reference solutions” are provided for the dynamic stabilization of BeagleMiP and BeagleQuad to ease the out-of-the-box experience; the feedback controllers implemented to stabilize these vehicles are introduced in Volume 1, but are not explained fully until Volume 2. In the experience of the authors, individual student ownership and hackability/extensibility of prototype systems is essential to foster experimentation and creativity, and results in an unmitigated intellectual investment by students that is simply not achievable via group projects or institution-owned educational systems. In short, BeagleRover, BeagleQuad, and BeagleMiP form an extensible family of inexpensive test vehicles that physically embody a rich variety of advanced topics in robotics, kinematics, dynamics, control, and multi-vehicle coordination that a student may grow with through high school, college, and beyond.

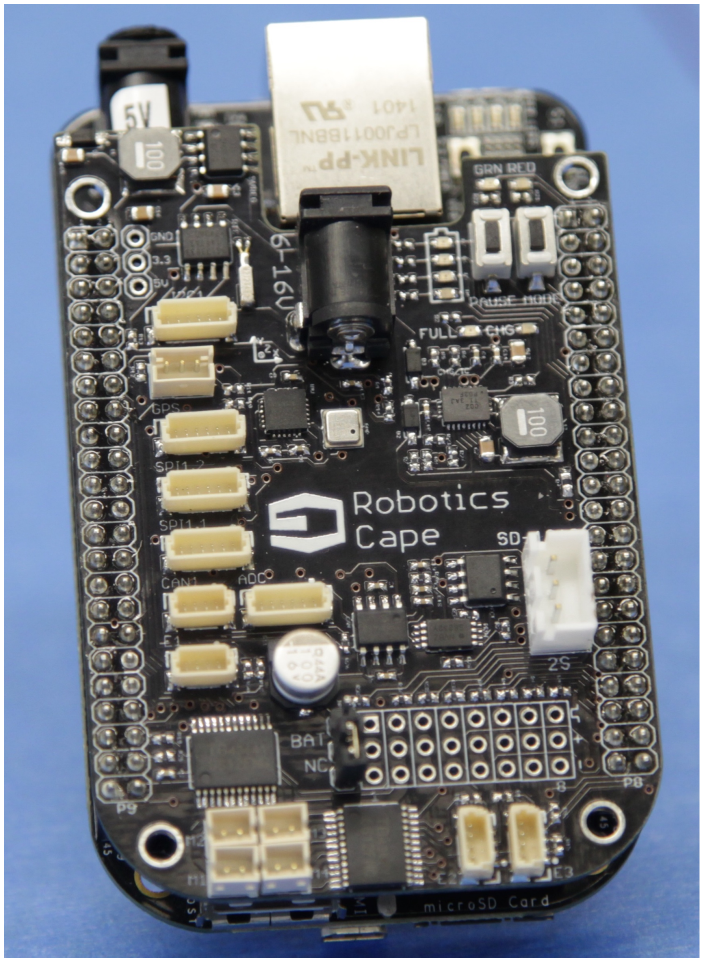

Figure 4. The feature-rich Robotics Cape developed by our team over 2013-2014 in support of MAE143c and related research projects based on the BeagleBone Black. The cape incorporates the commonly-needed additional electronics for many robotics applications, including an IMU, barometer, battery management, H-bridges, real-time clock with battery backup, and over a dozen convenient connectors, into a very small footprint.

III. A feature-rich robotics cape for the BBB

Our robotics cape (Figure 4) is a dense, feature-rich custom PCB implementing an Invensense MPU-9150 IMU, a Bosch BMP180 barometer, small H-bridges for direct connection to up to four DC motors at up to 1.2A each, voltage regulation from an 8 to 16V DC power source, battery management circuitry facilitating 1000ma charging of 2-cell LiPos (incorporating cut-off protection preventing both over-charging and over-discharging, with 5V/2A provided to the BBB itself to keep it running), a real-time clock with battery backup, 2-4 cell battery monitoring with an LED indicator, and over a dozen standard connectors for easy connection to COTS I2C, SPI, R/C, and GPS peripherals.

IV. An extensive multithreaded software library for the BBB

The well-documented software library we have published for this project, written in C, is completely open source, enabling new users to use it reliably out of the box based on the examples provided, to peek in and understand and modify/extend these library functions as they develop a deeper understanding of the project (and thereby encounter more complex needs), and to add to the software codebase itself.

Figure 5. MiP, by WowWee Robotics in collaboration with the UCSD Coordinated Robotics Lab.

A central focus of the software library is that it demonstrates clearly how to set up different threads running at different loop rates and priorities within Debian Linux, in order to keep time-critical (e.g., balancing) functions running at near-real time, with auxiliary functions fit in on an as-needed basis. This best-practices programming style is in sharp contrast with the programming experience that makers gain in the popular Arduino "sketch" environment, which unfortunately perpetuates outdated single-threaded programming approaches. In contrast, our team advocates a multithreaded, low-level, linux-based approach to robotics; when presented correctly, this approach is fully within the reach of both makers and professionals alike, and the construction of complex multifunction robotic systems necessitate such a multithreaded mindset in order to advance.

As illustrated in Figure 6, we have also developed a new, general-purpose, flexible Android app for both remote control and telemetry over wifi. This highly extensible app is suitable for linux-based robotic systems such as BeagleMiP, BeagleQuad, and BeagleRover, and many others of the users choosing. Use and extension of this app will also be described in the educational materials provided in Volume 1.

V. Tie to Commercial Products

The UCSD Coordinated Robotics Lab worked closely with WowWee Robotics over two years to develop a high volume, low-cost Mobile Inverted Pendulum toy dubbed MiP (Figure 5), which promises to be a banner tech toy in 2014 and 2015. Many exciting new consumer products and toys are in the pipeline as a result of this productive partnership. The educational materials we are developing reflect the design-for-low-cost-manufacturing techniques our team has learned while working closely with WowWee Robotics on MiP and related problems, and how these techniques might be extended to vehicles designed to be deployed in "squads" or "swarms" for more serious applications, such as firefighting, search & rescue, and sentry applications. We are also planning the development of follow-on educational resources detailing how multithreaded linux codes can be ported to low-cost ARM chips, such as the ARM Cortex M0 implemented in MiP, using a lightweight software library built on top of the Keil RTX RTOS to provide efficient multithreading. This software library will closely model that discussed here for the BBB.

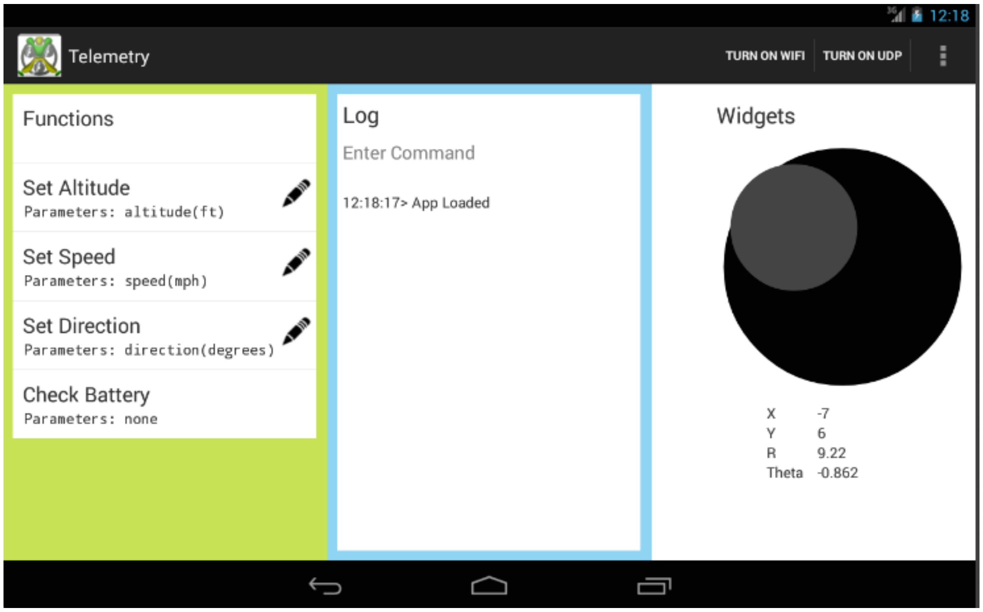

Figure 6. A new, general-purpose, flexible Android app for control and telemetry over wifi, suitable for linux-based robotic systems such as BeagleMiP, BeagleQuad, and BeagleRover, and many others.

VI. Conclusions

There are a number of essential topics in robotics that can be taught with an exceptionally effective, hands-on, exploratory approach using the family of vehicles and accompanying hardware/software described here, including:

1. Discrete-time control and stability augmentation of continuous-time stable and unstable systems.
2. Multivehicle coordination algorithms.
3. Multithreaded software architectures leveraging open-source standards on powerful low-cost Linux computers.
4. PCB design software (e.g. Eagle) coupled with mail-order PCB fabrication and low-volume PCB assembly.
5. 3D CAD software (e.g. SketchUp or Solidworks) coupled with 3D printing and laser cutting for rapid prototyping.
6. Smartphone and tablet SDKs, and their use in the development of a general app for control of robots.

What began as the development of a simple controls lab for a one-quarter senior-level course at UCSD has grown into a meticulous representative embodiment of a wide range of key topics in robotics, including multithreaded software architectures, embedded controls, rapid prototyping, custom PCB fabrication, integration with smartphones & tablets, and design-for-large-scale-manufacturing. The resulting educational program provides a unique backdrop for motivating makers with many aspects of STEM that they might otherwise be unaware of, and provides students in college and industry with compelling capstone projects in robotics that tie together key component technologies and concepts, and provide clear best-practice example realizations for the solution of many common problems in robotics.

Acknowledgements

The authors gratefully acknowledge Jason Kridner, Gerald Coley, and Cathy Wicks at TI, and Peter Yanofsky, Davin Sufer, and Adam Fairless at WowWee Robotics, for their generous support and involvement in this project, Saam Ostovari for his tireless efforts in the Arduino-based (2012) offering of MAE143c, David Schwartz for his active involvement in the 2013 offering of MAE143c, Robert Bitmead for his creative lectures in the 2014 offering of MAE143c, Zachary Nagengast and Steven Hypnarowski for their dedicated work on the Android Telemetry App and BeagleRover, and the many dedicated students at UCSD who helped our team refine these teaching materials during the 2012, 2013, and 2014 offerings of MAE143c, as well as the several undergraduate research projects that grew out of these classes.